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	Surface Mount
Package / Case	16-SOIC (0.295", 7.50mm Width)
Supplier Device Package	16-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc68hc908kx8cdw

How to Reach Us:

Home Page:
freescale.com

Web Support:
freescale.com/support

Document Number: MC68HC908KX8
Addendum to rev. 2.1
10/2014

- Serial communications interface (SCI) module
- 5-bit keyboard interrupt (KBI) with wakeup feature
- 13 general-purpose input/output (I/O) ports:
  - Five shared with KBI and TIM, with 15-mA source/15-mA sink capabilities and with programmable pullups on general- purpose input ports
  - Four shared with ADC
  - Two shared with SCI
- Low-voltage inhibit (LVI) module with software selectable trip points, 2.6-V or 4.3-V trip point
- Timebase module (TBM) with
  - Clock prescaler for eight user-selectable, periodic real-time interrupts
  - Active clock source in stop mode for periodic wakeup from stop using external crystal or internal oscillator
- External asynchronous interrupt pin with internal pullup ($\overline{\text{IRQ1}}$)
- System protection features:
  - Computer operating properly (COP) reset
  - Low-voltage detection with reset
  - Illegal opcode detection with reset
  - Illegal address detection with reset
- 16-pin plastic dual in-line (PDIP) or small outline (SOIC) package
- Low-power design fully static with stop and wait modes
- Internal power-up reset circuit requiring no external pins
- –40°C to +125°C operation

Features of the CPU08 include:

- Enhanced HC05 programming model
- Extensive loop control functions
- 16 addressing modes, eight more than the M68HC05
- 16-bit index register and stack pointer
- Memory-to-memory data transfers
- Fast 8 × 8 multiply instruction
- Fast 16/8 divide instruction
- Binary-coded decimal (BCD) instructions
- Optimization for controller applications
- Third party C language support

1.3 MCU Block Diagram

Figure 1-1 shows the structure of the MC68HC908KX8.

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$FE05	Interrupt Status Register 2 (INT2) See page 150.	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
$FE06	Interrupt Status Register 3 (INT3) See page 150.	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
$FE07	FLASH Test Control Register (FLTCR)	Read:								
		Write:	R	R	R	R	R	R	R	R
		Reset:								
$FE08	FLASH Control Register (FLCR) See page 31.	Read:	0	0	0	0	HVEN	MARGIN	ERASE	PGM
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$FE09	Break Address Register High (BRKH) See page 172.	Read:	Bit 15	14	13	12	11	10	9	Bit 8
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$FE0A	Break Address Register Low (BRKL) See page 172.	Read:	Bit 7	6	5	4	3	2	1	Bit 0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$FE0B	Break Status and Control Register (BRKSCR) See page 171.	Read:	BRKE	BRKA	0	0	0	0	0	0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$FE0C	LVI Status Register (LVISR) See page 103.	Read:	LVIOUT	0	0	0	0	0	0	R
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$FF7E	FLASH Block Protect Register (FLBPR)(1) See page 36.	Read:	BPR7	BPR6	BPR5	BPR4	BPR3	BPR2	BPR1	BPR0
		Write:								
		Reset:	Unaffected by reset							

1. Non-volatile FLASH register

$FFFF	COP Control Register (COPCTL) See page 53.	Read:	Low byte of reset vector
		Write:	Writing clears COP counter (any value)
		Reset:	Unaffected by reset

= Unimplemented R = Reserved U = Unaffected

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

2.9 FLASH Program/Read Operation

Programming of the FLASH memory is done on a row basis. A row consists of 32 consecutive bytes starting from addresses $XX00, $XX20, $XX40, $XX60, $XX80, $XXA0, $XXC0, and $XXE0. Use this step-by-step procedure to program a row of FLASH memory (Figure 2-4 is a flowchart representation).

NOTE

Only bytes which are currently $FF may be programmed.

1. Set the PGM bit. This configures the memory for program operation and enables the latching of address and data for programming.
2. Read the FLASH block protect register.
3. Write any data to any FLASH location within the address range desired.
4. Wait for a time, t_{NVS} (minimum 10 µs).
5. Set the HVEN bit.
6. Wait for a time, t_{PGS} (minimum 5 µs).
7. Write data to the FLASH address being programmed(1).
8. Wait for time, t_{PROG} (minimum 30 µs).
9. Repeat step 7 and 8 until all desired bytes within the row are programmed.
10. Clear the PGM bit(1).
11. Wait for time, t_{NVH} (minimum 5 µs).
12. Clear the HVEN bit.
13. After time, t_{RCV} (typical 1 µs), the memory can be accessed in read mode again.

This program sequence is repeated throughout the memory until all data is programmed.

NOTE

Programming and erasing of FLASH locations cannot be performed by code being executed from the FLASH memory. While these operations must be performed in the order shown, other unrelated operations may occur between the steps. Do not exceed t_{PROG} maximum. See 17.11 Memory Characteristics.

1. The time between each FLASH address change, or the time between the last FLASH address programmed to clearing PGM bit, must not exceed the maximum programming time, t_{PROG} maximum.

2.10 FLASH Block Protection

Due to the ability of the on-board charge pump to erase and program the FLASH memory in the target application, provision is made for protecting a block of memory from unintentional erase or program operations due to system malfunction. This protection is done by using the FLASH block protect register (FLBPR). The FLBPR determines the range of the FLASH memory which is to be protected. The range of the protected area starts from a location defined by FLBPR and ends at the bottom of the FLASH memory ($FFFF). When the memory is protected, the HVEN bit cannot be set in either erase or program operations.

NOTE

In performing a program or erase operation, the FLASH block protect register must be read after setting the PGM or ERASE bit and before asserting the HVEN bit.

When FLBPR is programmed with all 0s, the entire memory is protected from being programmed and erased. When all the bits are erased (all 1s), the entire memory is accessible for program and erase.

When bits within the FLBPR are programmed, they lock a block of memory address ranges as shown in 2.11 FLASH Block Protect Register. Once the FLBPR is programmed with a value other than $FF, any erase or program of the FLBPR or the protected block of FLASH memory is prohibited. The FLBPR itself can be erased or programmed only with an external voltage, V_{TST}, present on the $\overline{IRQ}$ pin. This voltage also allows entry from reset into the monitor mode.

2.11 FLASH Block Protect Register

The FLASH block protect register (FLBPR) is implemented as a byte within the FLASH memory, and therefore can be written only during a programming sequence of the FLASH memory. The value in this register determines the starting location of the protected range within the FLASH memory.

Address: $FF7E

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	BPR7	BPR6	BPR5	BPR4	BPR3	BPR2	BPR1	BPR0
Reset:	U	U	U	U	U	U	U	U

U = Unaffected by reset. Initial value from factory is 1.
Write to this register is by a programming sequence to the FLASH memory.

Figure 2-5. FLASH Block Protect Register (FLBPR)

BPR7–BPR0 — FLASH Block Protect Bits

These eight bits represent bits 13–6 of a 16-bit memory address. Bits 15 and 14 are 1s and bits 5–0 are 0s.

The resultant 16-bit address is used for specifying the start address of the FLASH memory for block protection. The FLASH is protected from this start address to the end of FLASH memory, at $FFFF. With this mechanism, the protect start address can be $XX00, $XX40, etc., (64 bytes page boundaries) within the FLASH memory. See Figure 2-6 and Table 2-2.

5.4.6 Reset Vector Fetch

A reset vector fetch occurs when the vector address appears on the data bus. A reset vector fetch clears the COP prescaler.

5.4.7 COPD (COP Disable)

The COPD signal reflects the state of the COP disable bit (COPD) in the configuration register. See Chapter 4 Configuration Register (CONFIG).

5.4.8 COPRS (COP Rate Select)

The COPRS signal reflects the state of the COP rate select bit (COPRS) in the configuration register. See Chapter 4 Configuration Register (CONFIG).

5.5 COP Control Register

The COP control register (COPCTL) is located at address $FFFF and overlaps the reset vector. Writing any value to $FFFF clears the COP counter and stages 12–5 of the COP prescaler and starts a new timeout period. Reading location $FFFF returns the low byte of the reset vector.

Address: $FFFF

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	Low byte of reset vector							
Write:	Clear COP counter							
Reset:	Unaffected by reset							

Figure 5-2. COP Control Register (COPCTL)

5.6 Interrupts

The COP does not generate CPU interrupt requests.

5.7 Monitor Mode

The COP is disabled in monitor mode when V_{TST} is present on the $\overline{IRQ1}$ pin.

5.8 Low-Power Modes

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

5.8.1 Wait Mode

The COP remains active in wait mode. To prevent a COP reset during wait mode, periodically clear the COP counter in a CPU interrupt routine.

Table 6-1. Instruction Set Summary (Sheet 6 of 6)

Source Form	Operation	Description	Effect on CCR						Address Mode	Opcode	Operand	Cycles
			V	H	I	N	Z	C				
SWI	Software Interrupt	PC ← (PC) + 1; Push (PCL) SP ← (SP) – 1; Push (PCH) SP ← (SP) – 1; Push (X) SP ← (SP) – 1; Push (A) SP ← (SP) – 1; Push (CCR) SP ← (SP) – 1; I ← 1 PCH ← Interrupt Vector High Byte PCL ← Interrupt Vector Low Byte	–	–	1	–	–	–	INH	83		9
TAP	Transfer A to CCR	CCR ← (A)	↕	↕	↕	↕	↕	↕	INH	84		2
TAX	Transfer A to X	X ← (A)	–	–	–	–	–	–	INH	97		1
TPA	Transfer CCR to A	A ← (CCR)	–	–	–	–	–	–	INH	85		1
TST *opr* TSTA TSTX TST *opr*,X TST ,X TST *opr*,SP	Test for Negative or Zero	(A) – $00 or (X) – $00 or (M) – $00	0	–	–	↕	↕	–	DIR INH INH IX1 IX SP1	3D 4D 5D 6D 7D 9E6D	dd ff ff	3 1 1 3 2 4
TSX	Transfer SP to H:X	H:X ← (SP) + 1	–	–	–	–	–	–	INH	95		2
TXA	Transfer X to A	A ← (X)	–	–	–	–	–	–	INH	9F		1
TXS	Transfer H:X to SP	(SP) ← (H:X) – 1	–	–	–	–	–	–	INH	94		2
WAIT	Enable Interrupts; Wait for Interrupt	I bit ← 0; Inhibit CPU clocking until interrupted	–	–	0	–	–	–	INH	8F		1

A	Accumulator	*n*	Any bit
C	Carry/borrow bit	*opr*	Operand (one or two bytes)
CCR	Condition code register	PC	Program counter
dd	Direct address of operand	PCH	Program counter high byte
dd rr	Direct address of operand and relative offset of branch instruction	PCL	Program counter low byte
DD	Direct to direct addressing mode	REL	Relative addressing mode
DIR	Direct addressing mode	*rel*	Relative program counter offset byte
DIX+	Direct to indexed with post increment addressing mode	rr	Relative program counter offset byte
ee ff	High and low bytes of offset in indexed, 16-bit offset addressing	SP1	Stack pointer, 8-bit offset addressing mode
EXT	Extended addressing mode	SP2	Stack pointer 16-bit offset addressing mode
ff	Offset byte in indexed, 8-bit offset addressing	SP	Stack pointer
H	Half-carry bit	U	Undefined
H	Index register high byte	V	Overflow bit
hh ll	High and low bytes of operand address in extended addressing	X	Index register low byte
I	Interrupt mask	Z	Zero bit
ii	Immediate operand byte	&	Logical AND
IMD	Immediate source to direct destination addressing mode	|	Logical OR
IMM	Immediate addressing mode	⊕	Logical EXCLUSIVE OR
INH	Inherent addressing mode	( )	Contents of
IX	Indexed, no offset addressing mode	–( )	Negation (two's complement)
IX+	Indexed, no offset, post increment addressing mode	#	Immediate value
IX+D	Indexed with post increment to direct addressing mode	«	Sign extend
IX1	Indexed, 8-bit offset addressing mode	←	Loaded with
IX1+	Indexed, 8-bit offset, post increment addressing mode	?	If
IX2	Indexed, 16-bit offset addressing mode	:	Concatenated with
M	Memory location	↕	Set or cleared
N	Negative bit	—	Not affected

6.8 Opcode Map

See Table 6-2.

7.6.3 Slow External Clock (EXTSLOW)

Slow external clock (EXTSLOW), when set, will decrease the drive strength of the oscillator amplifier, enabling low-frequency crystal operation (30 kHz–100 kHz) if properly enabled with the external clock enable (EXTCLKEN) and external crystal enable (EXTXTALEN) bits. When clear, EXTSLOW enables high-frequency crystal operation (1 MHz to 8 MHz).

EXTSLOW, when set, also configures the clock monitor to expect an external clock source that is slower than the low-frequency base clock (60 Hz–307.2 kHz). When EXTSLOW is clear, the clock monitor will expect an external clock faster than the low-frequency base clock (307.2 kHz–32 MHz).

The default state for this option is clear.

7.6.4 Oscillator Enable In Stop (OSCENINSTOP)

Oscillator enable in stop (OSCENINSTOP), when set, will enable the ICG to continue to generate clocks (either CGMXCLK, CGMOUT, or TBMCLK) in stop mode. This function is used to keep the timebase running while the rest of the microcontroller stops. When OSCENINSTOP is clear, all clock generation will cease and CGMXCLK, CGMOUT, and TBMCLK will be forced low during stop.

The default state for this option is clear.

7.7 I/O Registers

The ICG contains five registers, summarized in Figure 7-10. These registers are:

- ICG control register
- ICG multiplier register
- ICG trim register
- ICG DCO divider control register
- ICG DCO stage control register

Several of the bits in these registers have interaction where the state of one bit may force another bit to a particular state or prevent another bit from being set or cleared. A summary of this interaction is shown in Table 7-5.

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$0035	ICG Control Register (ICGCR) See page 87.	Read:	CMIE	CMF	CMON	CS	ICGON	ICGS	ECGON	ECGS
		Write:		0(1)						
		Reset:	0	0	0	0	1	0	0	0
	1. See CMF bit description for method of clearing.									
$0037	ICG Multiplier Register (ICGMR) See page 88.	Read:		N6	N5	N4	N3	N2	N1	N0
		Write:								
		Reset:	0	0	0	1	0	1	0	1
$0038	ICG Trim Register (ICGTR) See page 89.	Read:	TRIM7	TRIM6	TRIM5	TRIM4	TRIM3	TRIM2	TRIM1	TRIM0
		Write:								
		Reset:	1	0	0	0	0	0	0	0
			= Unimplemented			R	= Reserved	U = Unaffected		

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

Chapter 8
External Interrupt (IRQ)

8.1 Introduction

The external interrupt (IRQ) module provides a maskable interrupt input.

8.2 Features

Features of the IRQ module include:

- A dedicated external interrupt pin ($\overline{\text{IRQ1}}$)
- IRQ1 interrupt control bits
- Internal pullup resistor
- Hysteresis buffer
- Programmable edge-only or edge- and level-interrupt sensitivity
- Automatic interrupt acknowledge

8.3 Functional Description

A logic 0 applied to the external interrupt pin can latch a central processor unit (CPU) interrupt request. Figure 8-2 shows the structure of the IRQ module.

Interrupt signals on the $\overline{\text{IRQ1}}$ pin are latched into the IRQ1 latch. An interrupt latch remains set until one of these actions occurs:

- Vector fetch — A vector fetch automatically generates an interrupt acknowledge signal that clears the latch that caused the vector fetch.
- Software clear — Software can clear an interrupt latch by writing to the appropriate acknowledge bit in the interrupt status and control register (ISCR). Writing a 1 to the ACK1 bit clears the IRQ1 latch.
- Reset — A reset automatically clears the interrupt latch.

The external interrupt pin is falling-edge triggered and is software- configurable to be both falling-edge and low-level triggered. The MODE1 bit in the ISCR controls the triggering sensitivity of the $\overline{\text{IRQ1}}$ pin.

When an interrupt pin is edge-triggered only, the interrupt latch remains set until a vector fetch, software clear, or reset occurs.

When an interrupt pin is both falling-edge and low-level triggered, the interrupt latch remains set until both of these occur:

- Vector fetch or software clear
- Return of the interrupt pin to logic 1

The keyboard flag bit (KEYF) in the keyboard status and control register can be used to see if a pending interrupt exists. The KEYF bit is not affected by the keyboard interrupt mask bit (IMASKK) which makes it useful in applications where polling is preferred.

To determine the logic level on a keyboard interrupt pin, use the data direction register to configure the pin as an input and read the data register.

NOTE

Setting a keyboard interrupt enable bit (KBIEx) forces the corresponding keyboard interrupt pin to be an input, overriding the data direction register. However, the data direction register bit must be a 0 for software to read the pin.

9.4 Keyboard Initialization

When a keyboard interrupt pin is enabled, the pin may initially be low and cause a false interrupt to occur. A false interrupt on an edge-triggered pin can be acknowledged immediately after enabling the pin. A false interrupt on an edge- and level-triggered interrupt pin must be acknowledged after the pin has been pulled high.

The internal pullup device, the pin capacitance, as well as the external load will factor into the actual amount of time it takes for the pin to pull high. Considering only an internal pullup of 48 kΩ and pin capacitance of 8 pF, the pullup time will be on the order of 1 μs.

To prevent a false interrupt on keyboard initialization:

1. Mask keyboard interrupts by setting the IMASKK bit in the keyboard status and control register.
2. Enable the KBI pins by setting the appropriate KBIEx bits in the keyboard interrupt enable register.
3. Write to the ACKK bit in the keyboard status and control register to clear any false interrupts.
4. Clear the IMASKK bit.

Another way to avoid a false interrupt:

1. Configure the keyboard pins as outputs by setting the appropriate DDRA bits in data direction register A.
2. Write 1s to the appropriate port A data register bits.
3. Enable the KBI pins by setting the appropriate KBIEx bits in the keyboard interrupt enable register.

9.5 Low-Power Modes

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

9.5.1 Wait Mode

The keyboard module remains active in wait mode. Clearing the IMASKK bit in the keyboard status and control register enables keyboard interrupt requests to bring the MCU out of wait mode.

9.5.2 Stop Mode

The keyboard module remains active in stop mode. Clearing the IMASKK bit in the keyboard status and control register enables keyboard interrupt requests to bring the MCU out of stop mode.

9.6 I/O Registers

Two registers control and monitor operation of the keyboard module:

- Keyboard status and control register, KBSCR
- Keyboard interrupt enable register, KBIER

9.6.1 Keyboard Status and Control Register

The keyboard status and control register (KBSCR):

- Flags keyboard interrupt requests
- Acknowledges keyboard interrupt requests
- Masks keyboard interrupt requests
- Controls keyboard interrupt triggering sensitivity

Address: $001A

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	0	0	0	KEYF	0	IMASKK	MODEK
Write:						ACKK		
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 9-4. Keyboard Status and Control Register (KBSCR)

Bits 7–4 — Not used

These read-only bits always read as 0s.

KEYF — Keyboard Flag Bit

This read-only bit is set when a keyboard interrupt is pending. Reset clears the KEYF bit.

1 = Keyboard interrupt pending
0 = No keyboard interrupt pending

ACKK — Keyboard Acknowledge Bit

Writing a 1 to this write-only bit clears the keyboard interrupt request. ACKK always reads as 0. Reset clears ACKK.

IMASKK — Keyboard Interrupt Mask Bit

Writing a 1 to this read/write bit prevents the output of the keyboard interrupt mask from generating interrupt requests. Reset clears the IMASKK bit.

1 = Keyboard interrupt requests masked
0 = Keyboard interrupt requests not masked

MODEK — Keyboard Triggering Sensitivity Bit

This read/write bit controls the triggering sensitivity of the keyboard interrupt pins. Reset clears MODEK.

1 = Keyboard interrupt requests on falling edges and low levels
0 = Keyboard interrupt requests on falling edges only

Setting the LVI 5-V or 3-V trip point bit, LVI5OR3, enables the trip point voltage, V_{TRIPF}, to be configured for 5-V operation. Clearing the LVI5OR3 bit enables the trip point voltage, V_{TRIPF}, to be configured for 3-V operation. The actual trip thresholds are specified in 17.5 5.0-Vdc DC Electrical Characteristics and .

NOTE

After a power-on reset, the LVI's default mode of operation is 3 volts. If a 5-V system is used, the user must set the LVI5OR3 bit to raise the trip point to 5-V operation.

If the user requires 5-V mode and sets the LVI5OR3 bit after power-on reset while the V_{DD} supply is not above the V_{TRIPR} for 5-V mode, the MCU will immediately go into reset. The next time the LVI releases the reset, the supply will be above the V_{TRIPR} for 5-V mode.

Once an LVI reset occurs, the MCU remains in reset until V_{DD} rises above a voltage, V_{TRIPR}, which causes the MCU to exit reset. See Chapter 13 System Integration Module (SIM) for the reset recovery sequence.

The output of the comparator controls the state of the LVIOUT flag in the LVI status register (LVISR) and can be used for polling LVI operation when the LVI reset is disabled.

10.3.1 Polled LVI Operation

In applications that can operate at V_{DD} levels below the V_{TRIPF} level, software can monitor V_{DD} by polling the LVIOUT bit. In the configuration register, the LVIPWRD bit must be at 0 to enable the LVI module, and the LVIRSTD bit must be at 1 to disable LVI resets.

10.3.2 Forced Reset Operation

In applications that require V_{DD} to remain above the V_{TRIPF} level, enabling LVI resets allows the LVI module to reset the MCU when V_{DD} falls below the V_{TRIPF} level. In the configuration register, the LVIPWRD and LVIRSTD bits must be at 0 to enable the LVI module and to enable LVI resets.

10.3.3 Voltage Hysteresis Protection

Once the LVI has triggered (by having V_{DD} fall below V_{TRIPF}), the LVI will maintain a reset condition until V_{DD} rises above the rising trip point voltage, V_{TRIPR}. This prevents a condition in which the MCU is continually entering and exiting reset if V_{DD} is approximately equal to V_{TRIPF}. V_{TRIPR} is greater than V_{TRIPF} by the hysteresis voltage, V_{HYS}.

10.3.4 LVI Trip Selection

The LVI5OR3 bit in the configuration register selects whether the LVI is configured for 5-V or 3-V protection.

NOTE

The microcontroller is guaranteed to operate at a minimum supply voltage. The trip point (V_{TRIPF} [5 V] or V_{TRIPF} [3 V]) may be lower than this. See 17.5 5.0-Vdc DC Electrical Characteristics and for the actual trip point voltages.

Chapter 12
Serial Communications Interface Module (SCI)

12.1 Introduction

The serial communications interface (SCI) allows asynchronous communications with peripheral devices and other microcontroller unit (MCU).

12.2 Features

The SCI module's features include:

- Full-duplex operation
- Standard mark/space non-return-to-zero (NRZ) format
- Choice of baud rate clock source:
  - Internal bus clock
  - CGMXCLK
- 32 programmable baud rates
- Programmable 8-bit or 9-bit character length
- Separately enabled transmitter and receiver
- Separate receiver and transmitter central processor unit (CPU) interrupt requests
- Programmable transmitter output polarity
- Two receiver wakeup methods:
  - Idle line wakeup
  - Address mark wakeup
- Interrupt-driven operation with eight interrupt flags:
  - Transmitter empty
  - Transmission complete
  - Receiver full
  - Idle receiver input
  - Receiver overrun
  - Noise error
  - Framing error
  - Parity error
- Receiver framing error detection
- Hardware parity checking
- 1/16 bit-time noise detection

SCRF — SCI Receiver Full Bit

This clearable, read-only bit is set when the data in the receive shift register transfers to the SCI data register. SCRF can generate an SCI receiver CPU interrupt request. When the SCRIE bit in SCC2 is set the SCRF generates a CPU interrupt request. In normal operation, clear the SCRF bit by reading SCS1 with SCRF set and then reading the SCDR. Reset clears SCRF.

1 = Received data available in SCDR
0 = Data not available in SCDR

IDLE — Receiver Idle Bit

This clearable, read-only bit is set when 10 or 11 consecutive 1s appear on the receiver input. IDLE generates an SCI error CPU interrupt request if the ILIE bit in SCC2 is also set. Clear the IDLE bit by reading SCS1 with IDLE set and then reading the SCDR. After the receiver is enabled, it must receive a valid character that sets the SCRF bit before an idle condition can set the IDLE bit. Also, after the IDLE bit has been cleared, a valid character must again set the SCRF bit before an idle condition can set the IDLE bit. Reset clears the IDLE bit.

1 = Receiver input idle
0 = Receiver input active (or idle since the IDLE bit was cleared)

OR — Receiver Overrun Bit

This clearable, read-only bit is set when software fails to read the SCDR before the receive shift register receives the next character. The OR bit generates an SCI error CPU interrupt request if the ORIE bit in SCC3 is also set. The data in the shift register is lost, but the data already in the SCDR is not affected. Clear the OR bit by reading SCS1 with OR set and then reading the SCDR. Reset clears the OR bit.

1 = Receive shift register full and SCRF = 1
0 = No receiver overrun

Software latency may allow an overrun to occur between reads of SCS1 and SCDR in the flag-clearing sequence. Figure 12-14 shows the normal flag-clearing sequence and an example of an overrun caused by a delayed flag-clearing sequence. The delayed read of SCDR does not clear the OR bit because OR was not set when SCS1 was read. Byte 2 caused the overrun and is lost. The next flag-clearing sequence reads byte 3 in the SCDR instead of byte 2.

In applications that are subject to software latency or in which it is important to know which byte is lost due to an overrun, the flag-clearing routine can check the OR bit in a second read of SCS1 after reading the data register.

NF — Receiver Noise Flag Bit

This clearable, read-only bit is set when the SCI detects noise on the RxD pin. NF generates an NF CPU interrupt request if the NEIE bit in SCC3 is also set. Clear the NF bit by reading SCS1 and then reading the SCDR. Reset clears the NF bit.

1 = Noise detected
0 = No noise detected

FE — Receiver Framing Error Bit

This clearable, read-only bit is set when a 0 is accepted as the stop bit. FE generates an SCI error CPU interrupt request if the FEIE bit in SCC3 also is set. Clear the FE bit by reading SCS1 with FE set and then reading the SCDR. Reset clears the FE bit.

1 = Framing error detected
0 = No framing error detected

13.7 SIM Registers

The SIM has four memory mapped registers described here.

1. SIM reset status register (SRSR)
2. Interrupt status register 1 (INT1)
3. Interrupt status register 2 (INT2)
4. Interrupt status register 2 (INT3)

13.7.1 SIM Reset Status Register

This register contains five bits that show the source of the last reset. The status register will clear automatically after reading it. A power-on reset sets the POR bit and clears all other bits in the register.

Address: $FE01

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	POR	0	COP	ILOP	ILAD	MENRST	LVI	0
Write:								
POR:	1	0	0	0	0	0	0	0

= Unimplemented

Figure 13-16. SIM Reset Status Register (SRSR)

POR — Power-On Reset Bit

1 = Last reset caused by POR circuit
0 = Read of SRSR

COP — Computer Operating Properly Reset Bit

1 = Last reset caused by COP counter
0 = POR or read of SRSR

ILOP — Illegal Opcode Reset Bit

1 = Last reset caused by an illegal opcode
0 = POR or read of SRSR

ILAD — Illegal Address Reset Bit (opcode fetches only)

1 = Last reset caused by an opcode fetch from an illegal address
0 = POR or read of SRSR

MENRST — Forced Monitor Mode Entry Reset Bit

1 = Last reset was caused by the MENRST circuit
0 = POR or read of SRSR

LVI — Low-Voltage Inhibit Reset Bit

1 = Last reset was caused by the LVI circuit
0 = POR or read of SRSR

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

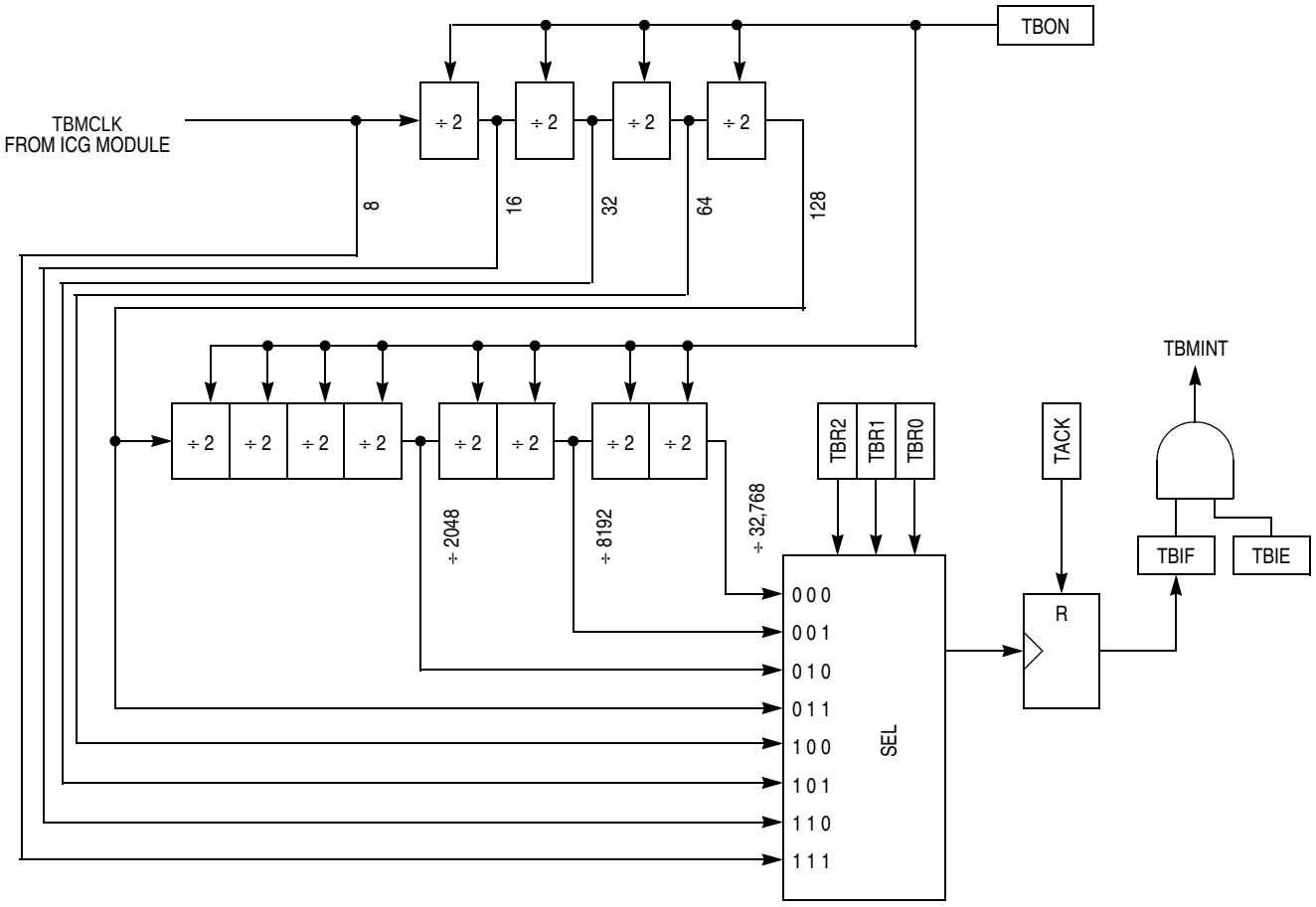

Figure 14-1. Timebase Block Diagram

The MCU executes the SWI and PSHH instructions when it enters monitor mode. The RUN command tells the MCU to execute the PULH and RTI instructions. Before sending the RUN command, the host can modify the stacked CPU registers to prepare to run the host program. The READSP command returns the incremented stack pointer value, SP + 1. The high and low bytes of the program counter are at addresses SP + 5 and SP + 6.

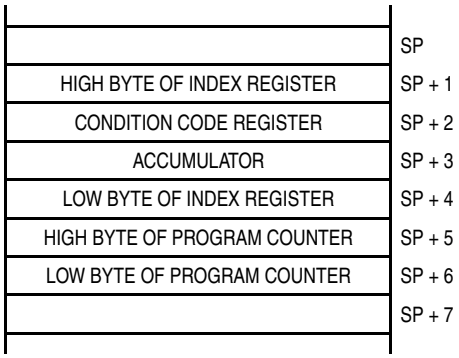

Figure 16-14. Stack Pointer at Monitor Mode Entry

16.3.2 Security

A security feature discourages unauthorized reading of FLASH locations while in monitor mode. The host can bypass the security feature at monitor mode entry by sending eight security bytes that match the bytes at locations $FFF6–$FFFD. Locations $FFF6–$FFFD contain user-defined data.

NOTE

Do not leave locations $FFF6–$FFFD blank. For security reasons, program locations $FFF6–$FFFD even if they are not used for vectors. If FLASH is erased, the eight security byte values to be sent to the MCU are $FF, the unprogrammed state of the FLASH.

During monitor mode entry, a reset must be asserted. PTA1 must be held low during the reset and 24 CGMXCLK cycles after the end of the reset. Then the MCU will wait for eight security bytes on PTA0. Each byte will be echoed back to the host. See Figure 16-15.

If the received bytes match those at locations $FFF6–$FFFD, the host bypasses the security feature and can read all FLASH locations and execute code from FLASH. Security remains bypassed until a reset occurs. After any reset, security will be locked. To bypass security again, the host must resend the eight security bytes on PTA0.

If the received bytes do not match the data at locations $FFF6–$FFFD, the host fails to bypass the security feature. The MCU remains in monitor mode, but reading FLASH locations returns undefined data, and trying to execute code from FLASH causes an illegal address reset.

B.4 Electrical Specifications

This subsection contains electrical and timing specifications for the MC68HC08KX8.

B.4.1 Absolute Maximum Ratings

Maximum ratings are the extreme limits to which the microcontroller unit (MCU) can be exposed without permanently damaging it.

NOTE

This device is not guaranteed to operate properly at the maximum ratings. Refer to B.4.4 5.0-Vdc DC Electrical Characteristics, and for guaranteed operating conditions.

Characteristic[1]	Symbol	Value	Unit
Supply voltage	V_{DD}	–0.3 to +6.0	V
Input voltage	V_{In}	V_{SS} –0.3 to V_{DD} +0.3	V
Maximum current per pin Excluding V_{DD}, V_{SS}, and PTA0–PTA4	I	±15	mA
Maximum current for pins PTA0–PTA4	I_{PTA0}–I_{PTA4}	±25	mA
Maximum current out of V_{SS}	I_{MVSS}	100	mA
Maximum current into V_{DD}	I_{MVDD}	100	mA
Storage temperature	T_{STG}	–55 to +150	°C

1. Voltages referenced to V_{SS}

NOTE

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum-rated voltages to this high-impedance circuit. For proper operation, it is recommended that V_{In} and V_{Out} be constrained to the range $V_{SS} \le (V_{In}$ or $V_{Out}) \le V_{DD}$. Reliability of operation is enhanced if unused inputs are connected to an appropriate logic voltage level (for example, either V_{SS} or V_{DD}).