



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 "<u>Embedded -</u> <u>Microcontrollers</u>"

Details

Product Status	Obsolete
Core Processor	HC08
Core Size	8-Bit
Speed	8MHz
Connectivity	CANbus, LINbus, SCI, SPI
Peripherals	LVD, POR, PWM
Number of I/O	21
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	32-LQFP
Supplier Device Package	32-LQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc68hc908gz8mfj

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong



General Description

These port pins also have selectable pullups when configured for input mode. The pullups are disengaged when configured for output mode. The pullups are selectable on an individual port bit basis.

1.5.12 Port E I/O Pins (PTE5-PTE2, PTE1/RxD, and PTE0/TxD)

PTE5–PTE0 are general-purpose, bidirectional I/O port pins. PTE1 and PTE0 can also be programmed to be enhanced serial communications interface (ESCI) pins. PTE5–PTE2 are only available on the 48-pin LQFP package. See Chapter 15 Enhanced Serial Communications Interface (ESCI) Module and Chapter 13 Input/Output (I/O) Ports.

1.6 Unused Pin Termination

Input pins and I/O port pins that are not used in the application must be terminated. This prevents excess current caused by floating inputs, and enhances immunity during noise or transient events. Termination methods include:

- 1. Configuring unused pins as outputs and driving high or low;
- 2. Configuring unused pins as inputs and enabling internal pull-ups;
- 3. Configuring unused pins as inputs and using external pull-up or pull-down resistors.

Never connect unused pins directly to V_{DD} or V_{SS} .

Since some general-purpose I/O pins are not available on all packages, these pins must be terminated as well. Either method 1 or 2 above are appropriate.

Input/Output (I/O) Section



Vector Priority	Vector	Address	Vector
	VCCIO	\$FFD4	MSCAN08 Transmit Vector (High)
2011001	IF20	\$FED5	MSCAN08 Transmit Vector (Low)
1 Î		\$FFD6	MSCAN08 Beceive Vector (High)
	IF19	\$FFD7	MSCAN08 Receive Vector (Low)
		\$FFD8	MSCAN08 Error Vector (High)
	IF18	\$FFD9	MSCAN08 Error Vector (Low)
		\$FFDA	MSCAN08 Wakeup Vector (High)
	IF17	\$FFDB	MSCAN08 Wakeup Vector (Low)
	1540	\$FFDC	Timebase Vector (High)
	IF16	\$FFDD	Timebase Vector (Low)
	1545	\$FFDE	ADC Conversion Complete Vector (High)
	IF 15	\$FFDF	ADC Conversion Complete Vector (Low)
	1514	\$FFE0	Keyboard Vector (High)
	1614	\$FFE1	Keyboard Vector (Low)
	1510	\$FFE2	ESCI Transmit Vector (High)
	1113	\$FFE3	ESCI Transmit Vector (Low)
	IE10	\$FFE4	ESCI Receive Vector (High)
	1612	\$FFE5	ESCI Receive Vector (Low)
	IF11	\$FFE6	ESCI Error Vector (High)
		\$FFE7	ESCI Error Vector (Low)
	IF10 IF9	\$FFE8	SPI Transmit Vector (High)
		\$FFE9	SPI Transmit Vector (Low)
		\$FFEA	SPI Receive Vector (High)
		\$FFEB	SPI Receive Vector (Low)
	IF8	\$FFEC	TIM2 Overflow Vector (High)
		\$FFED	TIM2 Overflow Vector (Low)
	IF7	\$FFEE	TIM2 Channel 1 Vector (High)
		\$FFEF	TIM2 Channel 1 Vector (Low)
	IF6	\$FFF0	TIM2 Channel 0 Vector (High)
		\$FFF1	TIM2 Channel 0 Vector (Low)
	IE5	\$FFF2	TIM1 Overflow Vector (High)
	11.0	\$FFF3	TIM1 Overflow Vector (Low)
	IF4	\$FFF4	TIM1 Channel 1 Vector (High)
		\$FFF5	TIM1 Channel 1 Vector (Low)
	IE3	\$FFF6	TIM1 Channel 0 Vector (High)
	110	\$FFF7	TIM1 Channel 0 Vector (Low)
	IF2	\$FFF8	PLL Vector (High)
		\$FFF9	PLL Vector (Low)
	IF1	\$FFFA	IRQ Vector (High)
		\$FFFB	IRQ Vector (Low)
		\$FFFC	SWI Vector (High)
		\$FFFD	SWI Vector (Low)
♥		\$FFFE	Reset Vector (High)
Highest		\$FFFF	Reset Vector (Low)

Table 2-1. Vector Addresses



Chapter 4 Clock Generator Module (CGM)

4.1 Introduction

This section describes the clock generator module. The CGM generates the crystal clock signal, CGMXCLK, which operates at the frequency of the crystal. The CGM also generates the base clock signal, CGMOUT, which is based on either the crystal clock divided by two or the phase-locked loop (PLL) clock, CGMVCLK, divided by two. In user mode, CGMOUT is the clock from which the SIM derives the system clocks, including the bus clock, which is at a frequency of CGMOUT/2. The PLL is a fully functional frequency generator designed for use with crystals or ceramic resonators. The PLL can generate a maximum bus frequency of 8 MHz using a 1-8MHz crystal or external clock source.

4.2 Features

Features of the CGM include:

- Phase-locked loop with output frequency in integer multiples of an integer dividend of the crystal reference
- High-frequency crystal operation with low-power operation and high-output frequency resolution
- Programmable hardware voltage-controlled oscillator (VCO) for low-jitter operation
- Automatic bandwidth control mode for low-jitter operation
- Automatic frequency lock detector
- CPU interrupt on entry or exit from locked condition
- Configuration register bit to allow oscillator operation during stop mode

4.3 Functional Description

The CGM consists of three major submodules:

- Crystal oscillator circuit The crystal oscillator circuit generates the constant crystal frequency clock, CGMXCLK.
- Phase-locked loop (PLL) The PLL generates the programmable VCO frequency clock, CGMVCLK.
- Base clock selector circuit This software-controlled circuit selects either CGMXCLK divided by two or the VCO clock, CGMVCLK, divided by two as the base clock, CGMOUT. The SIM derives the system clocks from either CGMOUT or CGMXCLK.

Figure 4-1 shows the structure of the CGM.



4.3.1 Crystal Oscillator Circuit

The crystal oscillator circuit consists of an inverting amplifier and an external crystal. The OSC1 pin is the input to the amplifier and the OSC2 pin is the output. The SIMOSCEN signal from the system integration module (SIM) or the OSCSTOPENB bit in the CONFIG register enable the crystal oscillator circuit.

The CGMXCLK signal is the output of the crystal oscillator circuit and runs at a rate equal to the crystal frequency. CGMXCLK is then buffered to produce CGMRCLK, the PLL reference clock.

CGMXCLK can be used by other modules which require precise timing for operation. The duty cycle of CGMXCLK is not guaranteed to be 50% and depends on external factors, including the crystal and related external components. An externally generated clock also can feed the OSC1 pin of the crystal oscillator circuit. Connect the external clock to the OSC1 pin and let the OSC2 pin float.

4.3.2 Phase-Locked Loop Circuit (PLL)

The PLL is a frequency generator that can operate in either acquisition mode or tracking mode, depending on the accuracy of the output frequency. The PLL can change between acquisition and tracking modes either automatically or manually.

4.3.3 PLL Circuits

The PLL consists of these circuits:

- Voltage-controlled oscillator (VCO)
- Modulo VCO frequency divider
- Phase detector
- Loop filter
- Lock detector

The operating range of the VCO is programmable for a wide range of frequencies and for maximum immunity to external noise, including supply and CGMXFC noise. The VCO frequency is bound to a range from roughly one-half to twice the center-of-range frequency, f_{VRS} . Modulating the voltage on the CGMXFC pin changes the frequency within this range. By design, f_{VRS} is equal to the nominal center-of-range frequency, f_{NOM} , (71.4 kHz) times a linear factor, L, and a power-of-two factor, E, or $(L \times 2^E) f_{NOM}$.

CGMRCLK is the PLL reference clock, a buffered version of CGMXCLK. CGMRCLK runs at a frequency, f_{RCLK} . The VCO's output clock, CGMVCLK, running at a frequency, f_{VCLK} , is fed back through a programmable modulo divider. The modulo divider reduces the VCO clock by a factor, N. The dividers output is the VCO feedback clock, CGMVDV, running at a frequency, $f_{VDV} = f_{VCLK}/(N)$. For more information, see 4.3.6 Programming the PLL.

The phase detector then compares the VCO feedback clock, CGMVDV, with the final reference clock, CGMRDV. A correction pulse is generated based on the phase difference between the two signals. The loop filter then slightly alters the DC voltage on the external capacitor connected to CGMXFC based on the width and direction of the correction pulse. The filter can make fast or slow corrections depending on its mode, described in 4.3.4 Acquisition and Tracking Modes. The value of the external capacitor and the reference frequency determines the speed of the corrections and the stability of the PLL.

The lock detector compares the frequencies of the VCO feedback clock, CGMVDV, and the reference clock, CGMRCLK. Therefore, the speed of the lock detector is directly proportional to the reference



Computer Operating Properly (COP) Module

6.7.2 Stop Mode

Stop mode turns off the CGMXCLK input to the COP and clears the COP prescaler. Service the COP immediately before entering or after exiting stop mode to ensure a full COP timeout period after entering or exiting stop mode.

To prevent inadvertently turning off the COP with a STOP instruction, a configuration option is available that disables the STOP instruction. When the STOP bit in the configuration register has the STOP instruction disabled, execution of a STOP instruction results in an illegal opcode reset.

6.8 COP Module During Break Mode

The COP is disabled during a break interrupt when V_{TST} is present on the \overline{RST} pin.



Central Processor Unit (CPU)

7.3.5 Condition Code Register

The 8-bit condition code register contains the interrupt mask and five flags that indicate the results of the instruction just executed. Bits 6 and 5 are set permanently to 1. The following paragraphs describe the functions of the condition code register.



Figure 7-6. Condition Code Register (CCR)

V — Overflow Flag

The CPU sets the overflow flag when a two's complement overflow occurs. The signed branch instructions BGT, BGE, BLE, and BLT use the overflow flag.

- 1 = Overflow
- 0 = No overflow

H — Half-Carry Flag

The CPU sets the half-carry flag when a carry occurs between accumulator bits 3 and 4 during an add-without-carry (ADD) or add-with-carry (ADC) operation. The half-carry flag is required for binary-coded decimal (BCD) arithmetic operations. The DAA instruction uses the states of the H and C flags to determine the appropriate correction factor.

1 = Carry between bits 3 and 4

0 = No carry between bits 3 and 4

I — Interrupt Mask

When the interrupt mask is set, all maskable CPU interrupts are disabled. CPU interrupts are enabled when the interrupt mask is cleared. When a CPU interrupt occurs, the interrupt mask is set automatically after the CPU registers are saved on the stack, but before the interrupt vector is fetched.

- 1 = Interrupts disabled
- 0 = Interrupts enabled

NOTE

To maintain M6805 Family compatibility, the upper byte of the index register (H) is not stacked automatically. If the interrupt service routine modifies H, then the user must stack and unstack H using the PSHH and PULH instructions.

After the I bit is cleared, the highest-priority interrupt request is serviced first. A return-from-interrupt (RTI) instruction pulls the CPU registers from the stack and restores the interrupt mask from the stack. After any reset, the interrupt mask is set and can be cleared only by the clear interrupt mask software instruction (CLI).

N — Negative Flag

The CPU sets the negative flag when an arithmetic operation, logic operation, or data manipulation produces a negative result, setting bit 7 of the result.

1 = Negative result

0 = Non-negative result



MSCAN08 Controller (MSCAN08)







MSCAN08 Controller (MSCAN08)

12.12.3 Data Length Register (DLR)

This register keeps the data length field of the CAN frame.

DLC3–DLC0 — Data Length Code Bits

The data length code contains the number of bytes (data byte count) of the respective message. At transmission of a remote frame, the data length code is transmitted as programmed while the number of transmitted bytes is always 0. The data byte count ranges from 0 to 8 for a data frame. Table 12-5 shows the effect of setting the DLC bits.

	Data Byte			
DLC3	DLC2	DLC1	DLC0	Count
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8

Table 12-5. Data Length Codes

12.12.4 Data Segment Registers (DSRn)

The eight data segment registers contain the data to be transmitted or received. The number of bytes to be transmitted or being received is determined by the data length code in the corresponding DLR.

12.12.5 Transmit Buffer Priority Registers



Figure 12-14	. Transmit	Buffer	Priority	Register	(TBPR)
--------------	------------	--------	-----------------	----------	--------

PRIO7–PRIO0 — Local Priority

This field defines the local priority of the associated message buffer. The local priority is used for the internal prioritization process of the MSCAN08 and is defined to be highest for the smallest binary number. The MSCAN08 implements the following internal prioritization mechanism:

- All transmission buffers with a cleared TXE flag participate in the prioritization right before the SOF is sent.
- The transmission buffer with the lowest local priority field wins the prioritization.
- In case more than one buffer has the same lowest priority, the message buffer with the lower index number wins.



12.13 Programmer's Model of Control Registers

The programmer's model has been laid out for maximum simplicity and efficiency. Figure 12-15 gives an overview on the control register block of the MSCAN08.

Addr.	Register		Bit 7	6	5	4	3	2	1	Bit 0
\$0500	CMCR0	Read: Write:	0	0	0	SYNCH	TLNKEN	SLPAK	SLPRQ	SFTRES
\$0501	CMCR1	Read:	0	0	0	0	0	LOOPB	WUPM	CLKSRC
\$0502	CBTR0	Read: Write:	SJW1	SJW0	BRP5	BRP4	BRP3	BRP2	BRP1	BRP0
\$0503	CBTR1	Read: Write:	SAMP	TSEG22	TSEG21	TSEG20	TSEG13	TSEG12	TSEG11	TSEG10
\$0504	CRFLG	Read: Write:	WUPIF	RWRNIF	TWRNIF	RERRIF	TERRIF	BOFFIF	OVRIF	RXF
\$0505	CRIER	Read: Write:	WUPIE	RWRNIE	TWRNIE	RERRIE	TERRIE	BOFFIE	OVRIE	RXFIE
\$0506	CTFLG	Read: Write:	0	ABTAK2	ABTAK1	ABTAK0	0	TXE2	TXE1	TXE0
\$0507	CTCR	Read: Write:	0	ABTRQ2	ABTRQ1	ABTRQ0	0	TXEIE2	TXEIE1	TXEIE0
\$0508	CIDAC	Read: Write:	0	0	IDAM1	IDAM0	0	0	IDHIT1	IDHIT0
\$0509	Reserved	Read: Write:	R	R	R	R	R	R	R	R
\$050E	CRXERR	Read: Write:	RXERR7	RXERR6	RXERR5	RXERR4	RXERR3	RXERR2	RXERR1	RXERR0
\$050F	CTXERR	Read: Write:	TXERR7	TXERR6	TXERR5	TXERR4	TXERR3	TXERR2	TXERR1	TXERR0
\$0510	CIDAR0	Read: Write:	AC7	AC6	AC5	AC4	AC3	AC2	AC1	AC0
\$0511	CIDAR1	Read: Write:	AC7	AC6	AC5	AC4	AC3	AC2	AC1	AC0
				= Unimpleme	ented		R	= Reserved		

Figure 12-15. MSCAN08 Control Register Structure



Input/Output (I/O) Ports

PTAPUE7–PTAPUE0 — Port A Input Pullup Enable Bits

These writable bits are software programmable to enable pullup devices on an input port bit.

- 1 = Corresponding port A pin configured to have internal pullup
- 0 = Corresponding port A pin has internal pullup disconnected

13.4 Port B

Port B is an 8-bit special-function port that shares all eight of its pins with the analog-to-digital converter (ADC) module.

13.4.1 Port B Data Register

The port B data register (PTB) contains a data latch for each of the eight port pins.



Figure 13-6. Port B Data Register (PTB)

PTB7-PTB0 — Port B Data Bits

These read/write bits are software-programmable. Data direction of each port B pin is under the control of the corresponding bit in data direction register B. Reset has no effect on port B data.

AD7-AD0 — Analog-to-Digital Input Bits

AD7–AD0 are pins used for the input channels to the analog-to-digital converter module. The channel select bits in the ADC status and control register define which port B pin will be used as an ADC input and overrides any control from the port I/O logic by forcing that pin as the input to the analog circuitry. See Chapter 3 Analog-to-Digital Converter (ADC) for more information.

NOTE

Care must be taken when reading port B while applying analog voltages to AD7–AD0 pins. If the appropriate ADC channel is not enabled, excessive current drain may occur if analog voltages are applied to the PTBx/ADx pin, while PTB is read as a digital input. Those ports not selected as analog input channels are considered digital I/O ports.

13.4.2 Data Direction Register B

Data direction register B (DDRB) determines whether each port B pin is an input or an output. Writing a logic 1 to a DDRB bit enables the output buffer for the corresponding port B pin; a logic 0 disables the output buffer.



14.2.3.4 Illegal Opcode Reset

An illegal opcode reset is an internal reset caused by an opcode that is not in the instruction set. An illegal opcode reset sets the ILOP bit in the SIM reset status register.

If the stop enable bit, STOP, in the mask option register is a logic 0, the STOP instruction causes an illegal opcode reset.

14.2.3.5 Illegal Address Reset

An illegal address reset is an internal reset caused by opcode fetch from an unmapped address. An illegal address reset sets the ILAD bit in the SIM reset status register.

A data fetch from an unmapped address does not generate a reset.

14.2.4 System Integration Module (SIM) Reset Status Register

This read-only register contains flags to show reset sources. All flag bits are automatically cleared following a read of the register. Reset service can read the SIM reset status register to clear the register after power-on reset and to determine the source of any subsequent reset.

The register is initialized on power-up as shown with the POR bit set and all other bits cleared. During a POR or any other internal reset, the \overrightarrow{RST} pin is pulled low. After the pin is released, it will be sampled 32 CGMXCLK cycles later. If the pin is not above a V_{IH} at that time, then the PIN bit in the SRSR may be set in addition to whatever other bits are set.





Figure 14-2. SIM Reset Status Register (SRSR)

POR — Power-On Reset Flag

- 1 = Power-on reset since last read of SRSR
- 0 = Read of SRSR since last power-on reset

PIN — External Reset Flag

- 1 = External reset via RST pin since last read of SRSR
- 0 = POR or read of SRSR since last external reset

COP — Computer Operating Properly Reset Bit

- 1 = Last reset caused by timeout of COP counter
- 0 = POR or read of SRSR since any reset

ILOP — Illegal Opcode Reset Bit

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



Enhanced Serial Communications Interface (ESCI) Module



Figure 15-2. ESCI Module Block Diagram



Functional Description

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
\$0009	ESCI Prescaler Register 009 (SCPSC)	Read: Write:	PDS2	PDS1	PDS0	PSSB4	PSSB3	PSSB2	PSSB1	PSSB0
	See page 206.	Reset:	0	0	0	0	0	0	0	0
	ESCI Arbiter Control	Read:	A 1 4	ALOST	4140		AFIN	ARUN	AROVFL	ARD8
\$000A	Register (SCIACTL)	Write:	AIVIT		Alvio	AULK				
	See page 209.	Reset:	0	0	0	0	0	0	0	0
	ESCI Arbiter Data	Read:	ARD7	ARD6	ARD5	ARD4	ARD3	ARD2	ARD1	ARD0
\$000B	Register (SCIADAT)	Write:								
	See page 210.	Reset:	0	0	0	0	0	0	0	0
\$0013	ESCI Control Register 1 (SCC1)	Read: Write:	LOOPS	ENSCI	TXINV	М	WAKE	ILTY	PEN	PTY
	See page 196.	Reset:	0	0	0	0	0	0	0	0
\$0014	ESCI Control Register 2 (SCC2)	Read: Write:	SCTIE	TCIE	SCRIE	ILIE	TE	RE	RWU	SBK
	See page 198.	Reset:	0	0	0	0	0	0	0	0
	ESCI Control Register 3	Read:	R8	то	Р	Р		NEIE	EEIE	DEIE
\$0015	(SCC3)	Write:		10	n	n	ONIE	INCIE	FEIE	FLIC
	See page 200.	Reset:	U	0	0	0	0	0	0	0
	ESCI Status Register 1	Read:	SCTE	TC	SCRF	IDLE	OR	NF	FE	PE
\$0016	(SCS1)	Write:								
	See page 201.	Reset:	1	1	0	0	0	0	0	0
	ESCI Status Register 2	Read:	0	0	0	0	0	0	BKF	RPF
\$0017	(SCS2) See page 203	Write:								
	See page 203.	Reset:	0	0	0	0	0	0	0	0
	ESCI Data Register	Read:	R7	R6	R5	R4	R3	R2	R1	R0
\$0018	(SCDR) See page 204	Write:	T7	Т6	T5	T4	Т3	T2	T1	Т0
	000 page 204.	Reset:		1	. <u> </u>	Unaffecte	d by reset			
\$0019	ESCI Baud Rate Register (SCBR)	Read: Write:	LINT	LINR	SCP1	SCP0	R	SCR2	SCR1	SCR0
	See page 204.	Reset:	0	0	0	0	0	0	0	0
				= Unimplemented		R	= Reserved	U = Ur	naffected	

Figure 15-3. ESCI I/O Register Summary



Enhanced Serial Communications Interface (ESCI) Module

15.4.1 Data Format

The SCI uses the standard non-return-to-zero mark/space data format illustrated in Figure 15-4.





15.4.2 Transmitter

Figure 15-5 shows the structure of the SCI transmitter and the registers are summarized in Figure 15-3. The baud rate clock source for the ESCI can be selected via the configuration bit, ESCIBDSRC.



MC68HC908GZ16 •	MC68HC908GZ8	Data Sheet, Rev. 4
-----------------	--------------	--------------------



Enhanced Serial Communications Interface (ESCI) Module

15.8.3 ESCI Control Register 3

ESCI control register 3 (SCC3):

- Stores the ninth ESCI data bit received and the ninth ESCI data bit to be transmitted.
- Enables these interrupts:
 - Receiver overrun
 - Noise error
 - Framing error
 - Parity error



Figure 15-12. ESCI Control Register 3 (SCC3)

R8 — Received Bit 8

When the ESCI is receiving 9-bit characters, R8 is the read-only ninth bit (bit 8) of the received character. R8 is received at the same time that the SCDR receives the other 8 bits.

When the ESCI is receiving 8-bit characters, R8 is a copy of the eighth bit (bit 7). Reset has no effect on the R8 bit.

T8 — Transmitted Bit 8

When the ESCI is transmitting 9-bit characters, T8 is the read/write ninth bit (bit 8) of the transmitted character. T8 is loaded into the transmit shift register at the same time that the SCDR is loaded into the transmit shift register. Reset clears the T8 bit.

ORIE — Receiver Overrun Interrupt Enable Bit

This read/write bit enables ESCI error CPU interrupt requests generated by the receiver overrun bit, OR. Reset clears ORIE.

1 = ESCI error CPU interrupt requests from OR bit enabled

0 = ESCI error CPU interrupt requests from OR bit disabled

NEIE — Receiver Noise Error Interrupt Enable Bit

This read/write bit enables ESCI error CPU interrupt requests generated by the noise error bit, NE. Reset clears NEIE.

1 = ESCI error CPU interrupt requests from NE bit enabled

0 = ESCI error CPU interrupt requests from NE bit disabled

FEIE — Receiver Framing Error Interrupt Enable Bit

This read/write bit enables ESCI error CPU interrupt requests generated by the framing error bit, FE. Reset clears FEIE.

1 = ESCI error CPU interrupt requests from FE bit enabled

0 = ESCI error CPU interrupt requests from FE bit disabled

PEIE — Receiver Parity Error Interrupt Enable Bit

This read/write bit enables ESCI receiver CPU interrupt requests generated by the parity error bit, PE. Reset clears PEIE.

1 = ESCI error CPU interrupt requests from PE bit enabled

0 = ESCI error CPU interrupt requests from PE bit disabled





16.5.1.2 SWI Instruction

The SWI instruction is a non-maskable instruction that causes an interrupt regardless of the state of the interrupt mask (I bit) in the condition code register.

NOTE

A software interrupt pushes PC onto the stack. A software interrupt does not push PC – 1, as a hardware interrupt does.

16.5.1.3 Interrupt Status Registers

The flags in the interrupt status registers identify maskable interrupt sources. Table 16-3 summarizes the interrupt sources and the interrupt status register flags that they set. The interrupt status registers can be useful for debugging.

Priority	Interrupt Source	Interrupt Status Register Flag
Highest	Reset	_
	SWI instruction	_
Î	IRQ pin	11
	CGM clock monitor	12
	TIM1 channel 0	13
	TIM1 channel 1	14
	TIM1 overflow	15
	TIM2 channel 0	16
	TIM2 channel 1	17
	TIM2 overflow	18
	SPI receiver full	19
	SPI transmitter empty	110
	SCI receive error	l11
	SCI receive	112
	SCI transmit	113
	Keyboard	114
	ADC conversion complete	115
	Timebase module	116
	MSCAN08 wakeup	117
	MSCAN08 error	118
V	MSCAN08 receive	119
Lowest	MSCAN08 transmit	120

Table 16-3. Interrupt Sources



17.6 Queuing Transmission Data

The double-buffered transmit data register allows a data byte to be queued and transmitted. For an SPI configured as a master, a queued data byte is transmitted immediately after the previous transmission has completed. The SPI transmitter empty flag (SPTE) indicates when the transmit data buffer is ready to accept new data. Write to the transmit data register only when the SPTE bit is high. Figure 17-9 shows the timing associated with doing back-to-back transmissions with the SPI (SPSCK has CPHA: CPOL = 1:0).



(6) CPU READS SPSCR WITH SPRF BIT SET.

Figure 17-9. SPRF/SPTE CPU Interrupt Timing

The transmit data buffer allows back-to-back transmissions without the slave precisely timing its writes between transmissions as in a system with a single data buffer. Also, if no new data is written to the data buffer, the last value contained in the shift register is the next data word to be transmitted.

For an idle master or idle slave that has no data loaded into its transmit buffer, the SPTE is set again no more than two bus cycles after the transmit buffer empties into the shift register. This allows the user to queue up a 16-bit value to send. For an already active slave, the load of the shift register cannot occur until the transmission is completed. This implies that a back-to-back write to the transmit data register is not possible. The SPTE indicates when the next write can occur.



Timebase Module (TBM)

18.7 Timebase Control Register

The timebase has one register, the timebase control register (TBCR), which is used to enable the timebase interrupts and set the rate.



Figure 18-2. Timebase Control Register (TBCR)

TBIF — Timebase Interrupt Flag

This read-only flag bit is set when the timebase counter has rolled over.

- 1 = Timebase interrupt pending
- 0 = Timebase interrupt not pending

TBR2–TBR0 — Timebase Divider Selection Bits

These read/write bits select the tap in the counter to be used for timebase interrupts as shown in Table 18-1.

NOTE

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

TACK— Timebase Acknowledge Bit

The TACK bit is a write-only bit and always reads as 0. Writing a logic 1 to this bit clears TBIF, the timebase interrupt flag bit. Writing a logic 0 to this bit has no effect.

1 = Clear timebase interrupt flag

0 = No effect

TBIE — Timebase Interrupt Enabled Bit

This read/write bit enables the timebase interrupt when the TBIF bit becomes set. Reset clears the TBIE bit.

1 = Timebase interrupt is enabled.

0 = Timebase interrupt is disabled.

TBON — Timebase Enabled Bit

This read/write bit enables the timebase. Timebase may be turned off to reduce power consumption when its function is not necessary. The counter can be initialized by clearing and then setting this bit. Reset clears the TBON bit.

1 = Timebase is enabled.

0 = Timebase is disabled and the counter initialized to 0s.



Development Support





Table 20-8. RUN (Run User Program) Command



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.

	l
	SP
HIGH BYTE OF INDEX REGISTER	SP + 1
CONDITION CODE REGISTER	SP + 2
ACCUMULATOR	SP + 3
LOW BYTE OF INDEX REGISTER	SP + 4
HIGH BYTE OF PROGRAM COUNTER	SP + 5
LOW BYTE OF PROGRAM COUNTER	SP + 6
	SP + 7
	1

Figure 20-15. Stack Pointer at Monitor Mode Entry



Electrical Specifications

Characteristic ⁽¹⁾	Symbol	Min	Тур ⁽²⁾	Мах	Unit
Monitor mode entry voltage	V _{TST}	V _{DD} + 2.5		V _{DD} + 4.0	V
Low-voltage inhibit, trip falling voltage	V _{TRIPF}	2.35	2.6	2.7	V
Low-voltage inhibit, trip rising voltage	V _{TRIPR}	2.4	2.66	2.8	V
Low-voltage inhibit reset/recover hysteresis (V _{TRIPF} + V _{HYS} = V _{TRIPR})	V _{HYS}	_	100	_	mV
POR rearm voltage ⁽¹²⁾	V _{POR}	0		100	mV
POR reset voltage ⁽¹³⁾	V _{PORRST}	0	700	800	mV
POR rise time ramp rate ⁽¹⁴⁾	R _{POR}	0.035	_	_	V/ms

1. V_{DD} = 3.3 Vdc ± 10%, V_{SS} = 0 Vdc, T_A = T_A (min) to T_A (max), unless otherwise noted

2. Typical values reflect average measurements at midpoint of voltage range, 25°C only.

 Run (operating) I_{DD} measured using external square wave clock source (f_{OSC} = 16 MHz). All inputs 0.2 V from rail. No dc loads. Less than 100 pF on all outputs. C_L = 20 pF on OSC2. All ports configured as inputs. OSC2 capacitance linearly affects run I_{DD}. Measured with all modules enabled.

4. Wait I_{DD} measured using external square wave clock source ($f_{OSC} = 16$ MHz). All inputs 0.2 V from rail. No dc loads. Less than 100 pF on all outputs. C_L = 20 pF on OSC2. All ports configured as inputs. OSC2 capacitance linearly affects wait I_{DD}. Measured with CGM and LVI enabled.

Stop I_{DD} is measured with OSC1 = V_{SS}. All inputs 0.2 V from rail. No dc loads. Less than 100 pF on all outputs. All ports configured as inputs. Typical values at midpoint of voltage range, 25°C only.

6. Stop I_{DD} with TBM enabled is measured using an external square wave clock source (f_{OSC} = 4 MHz). All inputs 0.2 V from rail. No dc loads. Less than 100 pF on all outputs. All inputs configured as inputs.

7. This parameter is characterized and not tested on each device.

8. All functional non-supply pins are internally clamped to V_{SS} and V_{DD} .

9. Input must be current limited to the value specified. To determine the value of the required current-limiting resistor, calculate resistance values for positive and negative clamp voltages, then use the larger of the two values.

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

11. Pullups and pulldowns are disabled.

12. Maximum is highest voltage that POR is guaranteed.

13. Maximum is highest voltage that POR is possible.

14. If minimum V_{DD} is not reached before the internal POR reset is released, RST must be driven low ext ernally until minimum V_{DD} is reached.