

Welcome to [E-XFL.COM](https://www.e-xfl.com)

### What is "[Embedded - Microcontrollers](#)"?

"[Embedded - Microcontrollers](#)" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

### Applications of "[Embedded - Microcontrollers](#)"

#### Details

Product Status	Obsolete
Core Processor	HC08
Core Size	8-Bit
Speed	8MHz
Connectivity	CANbus, SCI, SPI
Peripherals	LVD, POR, PWM
Number of I/O	21
Program Memory Size	60KB (60K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	A/D 24x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	32-LQFP
Supplier Device Package	32-LQFP (7x7)
Purchase URL	<a href="https://www.e-xfl.com/product-detail/nxp-semiconductors/mc908gz60vfje">https://www.e-xfl.com/product-detail/nxp-semiconductors/mc908gz60vfje</a>

## List of Chapters

Chapter 1 General Description . . . . .	23
Chapter 2 Memory . . . . .	33
Chapter 3 Analog-to-Digital Converter (ADC) . . . . .	61
Chapter 4 Clock Generator Module (CGM) . . . . .	73
Chapter 5 Configuration Register (CONFIG) . . . . .	91
Chapter 6 Computer Operating Properly (COP) Module . . . . .	95
Chapter 7 Central Processor Unit (CPU) . . . . .	99
Chapter 8 External Interrupt (IRQ) . . . . .	111
Chapter 9 Keyboard Interrupt Module (KBI) . . . . .	115
Chapter 10 Low-Power Modes . . . . .	123
Chapter 11 Low-Voltage Inhibit (LVI) . . . . .	131
Chapter 12 MSCAN08 Controller (MSCAN08) . . . . .	135
Chapter 13 Input/Output (I/O) Ports . . . . .	169
Chapter 14 Enhanced Serial Communications Interface (ESCI) Module . . . . .	189
Chapter 15 System Integration Module (SIM) . . . . .	221
Chapter 16 Serial Peripheral Interface (SPI) Module . . . . .	239
Chapter 17 Timebase Module (TBM) . . . . .	259
Chapter 18 Timer Interface Module (TIM1) . . . . .	263
Chapter 19 Timer Interface Module (TIM2) . . . . .	279
Chapter 20 Development Support . . . . .	299
Chapter 21 Electrical Specifications . . . . .	315
Chapter 22 Ordering Information and Mechanical Specifications . . . . .	333
Appendix A MC68HC908GZ48 . . . . .	343
Appendix B MC68HC908GZ32 . . . . .	347

# Table of Contents

## Chapter 1 General Description

1.1	Introduction	23
1.2	Features	23
1.2.1	Standard Features	23
1.2.2	Features of the CPU08	25
1.3	MCU Block Diagram	25
1.4	Pin Assignments	25
1.5	Pin Functions	28
1.5.1	Power Supply Pins ( $V_{DD}$ and $V_{SS}$ )	28
1.5.2	Oscillator Pins (OSC1 and OSC2)	29
1.5.3	External Reset Pin ( $\overline{RST}$ )	29
1.5.4	External Interrupt Pin ( $\overline{IRQ}$ )	29
1.5.5	CGM Power Supply Pins ( $V_{DDA}$ and $V_{SSA}$ )	29
1.5.6	External Filter Capacitor Pin (CGMXFC)	29
1.5.7	ADC Power Supply/Reference Pins ( $V_{DDAD}/V_{REFH}$ and $V_{SSAD}/V_{REFL}$ )	29
1.5.8	Port A Input/Output (I/O) Pins (PTA7/KBD7/AD15–PTA0/KBD0/AD8)	30
1.5.9	Port B I/O Pins (PTB7/AD7–PTB0/AD0)	30
1.5.10	Port C I/O Pins (PTC6–PTC0/CANTX)	30
1.5.11	Port D I/O Pins (PTD7/T2CH1–PTD0/SS)	30
1.5.12	Port E I/O Pins (PTE5–PTE2, PTE1/RxD, and PTE0/TxD)	30
1.5.13	Port F I/O Pins (PTF7/T2CH5–PTF0)	31
1.5.14	Port G I/O Pins (PTG7/AD23–PTBG0/AD16)	31
1.5.15	Unused Pin Termination	31

## Chapter 2 Memory

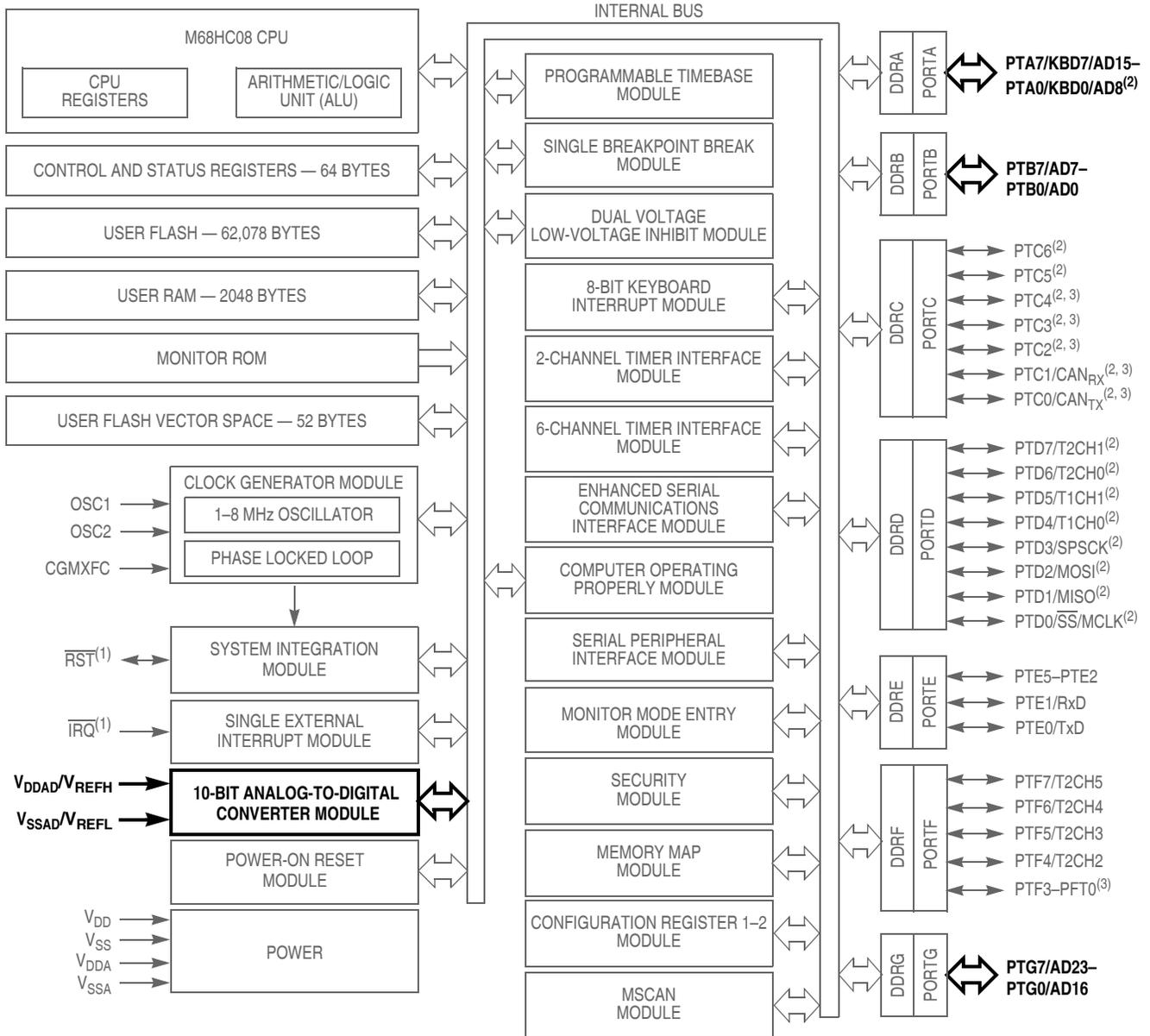
2.1	Introduction	33
2.2	Unimplemented Memory Locations	33
2.3	Reserved Memory Locations	33
2.4	Input/Output (I/O) Section	33
2.5	Random-Access Memory (RAM)	45
2.6	FLASH-1 Memory (FLASH-1)	45
2.6.1	Functional Description	45
2.6.2	FLASH-1 Control and Block Protect Registers	46
2.6.2.1	FLASH-1 Control Register	46
2.6.2.2	FLASH-1 Block Protect Register	47
2.6.3	FLASH-1 Block Protection	48
2.6.4	FLASH-1 Mass Erase Operation	48
2.6.5	FLASH-1 Page Erase Operation	49

Addr.	Register Name	Bit 7	6	5	4	3	2	1	Bit 0	
\$0000	Port A Data Register (PTA) See page 173.	Read:	PTA7	PTA6	PTA5	PTA4	PTA3	PTA2	PTA1	PTA0
		Write:								
		Reset:	Unaffected by reset							
\$0001	Port B Data Register (PTB) See page 176.	Read:	PTB7	PTB6	PTB5	PTB4	PTB3	PTB2	PTB1	PTB0
		Write:								
		Reset:	Unaffected by reset							
\$0002	Port C Data Register (PTC) See page 178.	Read:	1	PTC6	PTC5	PTC4	PTC3	PTC2	PTC1	PTC0
		Write:								
		Reset:	Unaffected by reset							
\$0003	Port D Data Register (PTD) See page 180.	Read:	PTD7	PTD6	PTD5	PTD4	PTD3	PTD2	PTD1	PTD0
		Write:								
		Reset:	Unaffected by reset							
\$0004	Data Direction Register A (DDRA) See page 174.	Read:	DDRA7	DDRA6	DDRA5	DDRA4	DDRA3	DDRA2	DDRA1	DDRA0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0005	Data Direction Register B (DDRB) See page 176.	Read:	DDRB7	DDRB6	DDRB5	DDRB4	DDRB3	DDRB2	DDRB1	DDRB0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0006	Data Direction Register C (DDRC) See page 178.	Read:	0	DDRC6	DDRC5	DDRC4	DDRC3	DDRC2	DDRC1	DDRC0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0007	Data Direction Register D (DDRD) See page 181.	Read:	DDRD7	DDRD6	DDRD5	DDRD4	DDRD3	DDRD2	DDRD1	DDRD0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0008	Port E Data Register (PTE) See page 183.	Read:	0	0	PTE5	PTE4	PTE3	PTE2	PTE1	PTE0
		Write:								
		Reset:	Unaffected by reset							
\$0009	ESCI Prescaler Register (SCPSC) See page 214.	Read:	PDS2	PDS1	PDS0	PSSB4	PSSB3	PSSB2	PSSB1	PSSB0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$000A	ESCI Arbiter Control Register (SCIACCTL) See page 217.	Read:	AM1	ALOST	AM0	ACLK	AFIN	ARUN	AROVFL	ARD8
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$000B	ESCI Arbiter Data Register (SCIADAT) See page 218.	Read:	ARD7	ARD6	ARD5	ARD4	ARD3	ARD2	ARD1	ARD0
		Write:								
		Reset:	0	0	0	0	0	0	0	0

= Unimplemented      R = Reserved      U = Unaffected

**Figure 2-2. Control, Status, and Data Registers (Sheet 1 of 9)**

## Analog-to-Digital Converter (ADC)



1. Pin contains integrated pullup device.

2. Ports are software configurable with pullup device if input port or pullup/pulldown device for keyboard input.

3. Higher current drive port pins

**Figure 3-1. Block Diagram Highlighting ADC Block and Pins**

The following conditions apply when in manual mode:

- $\overline{ACQ}$  is a writable control bit that controls the mode of the filter. Before turning on the PLL in manual mode, the  $\overline{ACQ}$  bit must be clear.
- Before entering tracking mode ( $\overline{ACQ} = 1$ ), software must wait a given time,  $t_{ACQ}$  (See 4.8 Acquisition/Lock Time Specifications.), after turning on the PLL by setting PLLON in the PLL control register (PCTL).
- Software must wait a given time,  $t_{AL}$ , after entering tracking mode before selecting the PLL as the clock source to CGMOUT (BCS = 1).
- The LOCK bit is disabled.
- CPU interrupts from the CGM are disabled.

### 4.3.6 Programming the PLL

Use the following procedure to program the PLL. For reference, the variables used and their meaning are shown in Table 4-1.

**Table 4-1. Variable Definitions**

Variable	Definition
$f_{BUSDES}$	Desired bus clock frequency
$f_{VCLKDES}$	Desired VCO clock frequency
$f_{RCLK}$	Chosen reference crystal frequency
$f_{VCLK}$	Calculated VCO clock frequency
$f_{BUS}$	Calculated bus clock frequency
$f_{NOM}$	Nominal VCO center frequency
$f_{VRS}$	Programmed VCO center frequency

**NOTE**

*The round function in the following equations means that the real number should be rounded to the nearest integer number.*

1. Choose the desired bus frequency,  $f_{BUSDES}$ .
2. Calculate the desired VCO frequency (four times the desired bus frequency).

$$f_{VCLKDES} = 4 \times f_{BUSDES}$$

3. Choose a practical PLL (crystal) reference frequency,  $f_{RCLK}$ . Typically, the reference crystal is 1–8 MHz.

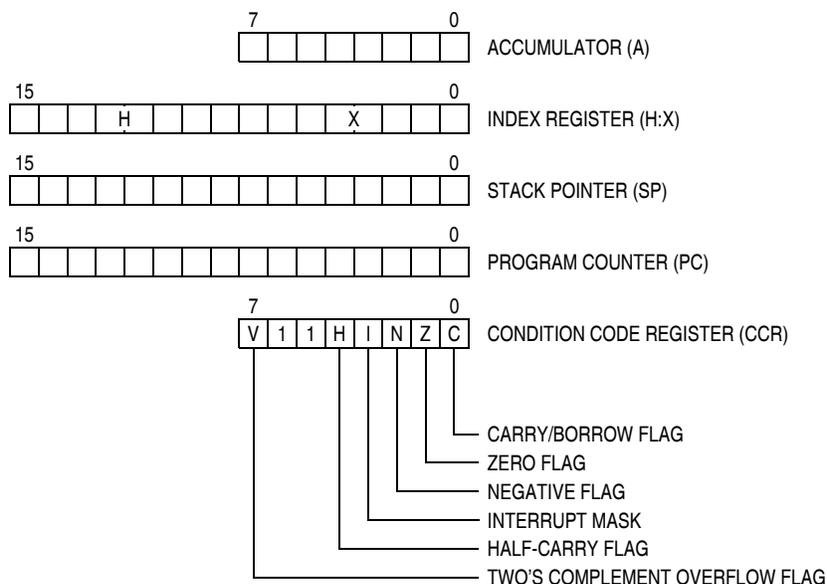
Frequency errors to the PLL are corrected at a rate of  $f_{RCLK}$ .

For stability and lock time reduction, this rate must be as fast as possible. The VCO frequency must be an integer multiple of this rate. The relationship between the VCO frequency,  $f_{VCLK}$ , and the reference frequency,  $f_{RCLK}$ , is:

$$f_{VCLK} = (N) (f_{RCLK})$$

N, the range multiplier, must be an integer.

## Central Processor Unit (CPU)



**Figure 7-1. CPU Registers**

### 7.3.1 Accumulator

The accumulator is a general-purpose 8-bit register. The CPU uses the accumulator to hold operands and the results of arithmetic/logic operations.



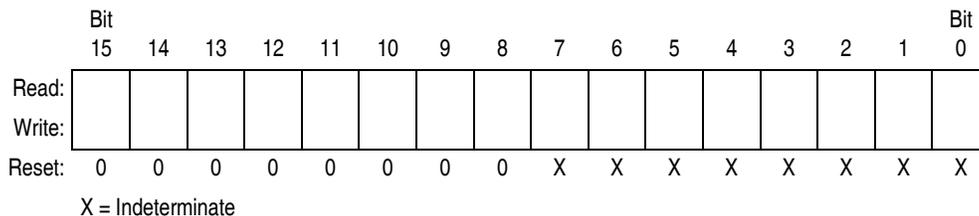
**Figure 7-2. Accumulator (A)**

### 7.3.2 Index Register

The 16-bit index register allows indexed addressing of a 64-Kbyte memory space. H is the upper byte of the index register, and X is the lower byte. H:X is the concatenated 16-bit index register.

In the indexed addressing modes, the CPU uses the contents of the index register to determine the conditional address of the operand.

The index register can serve also as a temporary data storage location.



**Figure 7-3. Index Register (H:X)**

### SFTRES — Soft Reset

When this bit is set by the CPU, the MSCAN08 immediately enters the soft reset state. Any ongoing transmission or reception is aborted and synchronization to the bus is lost.

The following registers enter and stay in their hard reset state:

CMCR0, CRFLG, CRIER, CTFLG, and CTCR.

The registers CMCR1, CBTR0, CBTR1, CIDAC, CIDAR0–CIDAR3, and CIDMR0–CIDMR3 can only be written by the CPU when the MSCAN08 is in soft reset state. The values of the error counters are not affected by soft reset.

When this bit is cleared by the CPU, the MSCAN08 tries to synchronize to the CAN bus. If the MSCAN08 is not in bus-off state, it will be synchronized after 11 recessive bits on the bus; if the MSCAN08 is in bus-off state, it continues to wait for 128 occurrences of 11 recessive bits.

Clearing SFTRES and writing to other bits in CMCR0 must be in separate instructions.

1 = MSCAN08 in soft reset state

0 = Normal operation

### 12.13.2 MSCAN08 Module Control Register 1

Address: \$0501

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	0	0	0	0	LOOPB	WUPM	CLKSRC
Write:								
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 12-17. Module Control Register (CMCR1)

### LOOPB — Loop Back Self-Test Mode

When this bit is set, the MSCAN08 performs an internal loop back which can be used for self-test operation: the bit stream output of the transmitter is fed back to the receiver internally. The CAN<sub>RX</sub> input pin is ignored and the CAN<sub>TX</sub> output goes to the recessive state (1). The MSCAN08 behaves as it does normally when transmitting and treats its own transmitted message as a message received from a remote node. In this state the MSCAN08 ignores the bit sent during the ACK slot of the CAN frame Acknowledge field to insure proper reception of its own message. Both transmit and receive interrupts are generated.

1 = Activate loop back self-test mode

0 = Normal operation

### WUPM — Wakeup Mode

This flag defines whether the integrated low-pass filter is applied to protect the MSCAN08 from spurious wakeups (see 12.8.5 Programmable Wakeup Function).

1 = MSCAN08 will wakeup the CPU only in cases of a dominant pulse on the bus which has a length of at least  $t_{wup}$ .

0 = MSCAN08 will wakeup the CPU after any recessive-to-dominant edge on the CAN bus.

**NOTE**

The CBTR0 register can be written only if the SFTRES bit in the MSCAN08 module control register is set.

**12.13.4 MSCAN08 Bus Timing Register 1**

Address: \$0503

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	SAMP	TSEG22	TSEG21	TSEG20	TSEG13	TSEG12	TSEG11	TSEG10
Write:								
Reset:	0	0	0	0	0	0	0	0

**Figure 12-19. Bus Timing Register 1 (CBTR1)**

**SAMP — Sampling**

This bit determines the number of serial bus samples to be taken per bit time. If set, three samples per bit are taken, the regular one (sample point) and two preceding samples, using a majority rule. For higher bit rates, SAMP should be cleared, which means that only one sample will be taken per bit.

- 1 = Three samples per bit<sup>(1)</sup>
- 0 = One sample per bit

**TSEG22–TSEG10 — Time Segment**

Time segments within the bit time fix the number of clock cycles per bit time and the location of the sample point. Time segment 1 (TSEG1) and time segment 2 (TSEG2) are programmable as shown in Table 12-8. The bit time is determined by the oscillator frequency, the baud rate prescaler, and the number of time quanta (T<sub>q</sub>) clock cycles per bit as shown in Table 12-4).

$$\text{Bit time} = \frac{\text{Pres value}}{f_{\text{MSCANCLK}}} \cdot \text{number of time quanta}$$

**NOTE**

The CBTR1 register can only be written if the SFTRES bit in the MSCAN08 module control register is set.

**Table 12-8. Time Segment Values**

TSEG13	TSEG12	TSEG11	TSEG10	Time Segment 1	TSEG22	TSEG21	TSEG20	Time Segment 2
0	0	0	0	1 T <sub>q</sub> Cycle <sup>(1)</sup>	0	0	0	1 T <sub>q</sub> Cycle <sup>(1)</sup>
0	0	0	1	2 T <sub>q</sub> Cycles <sup>(1)</sup>	0	0	1	2 T <sub>q</sub> Cycles
0	0	1	0	3T <sub>q</sub> Cycles <sup>(1)</sup>	.	.	.	.
0	0	1	1	4 T <sub>q</sub> Cycles	.	.	.	.
.	.	.	.	.	1	1	1	8T <sub>q</sub> Cycles
.	.	.	.	.				
1	1	1	1	16 T <sub>q</sub> Cycles				

1. This setting is not valid. Please refer to Table 12-4 for valid settings.

1. In this case PHASE\_SEG1 must be at least 2 time quanta.

### PTDPUE7–PTDPUE0 — Port D Input Pullup Enable Bits

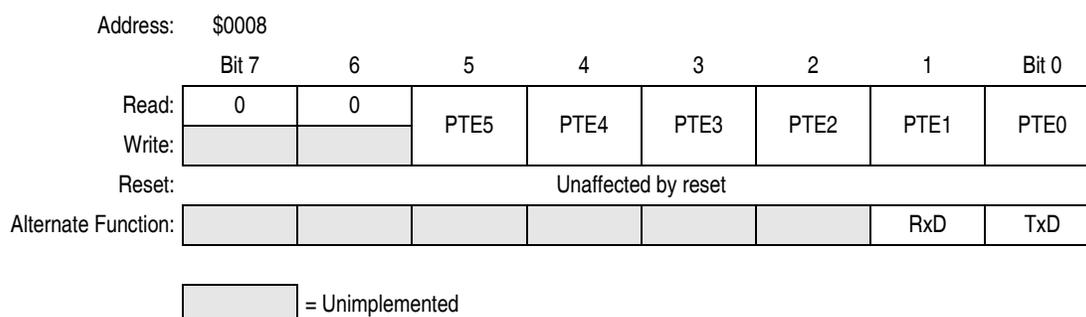
These writable bits are software programmable to enable pullup devices on an input port bit.  
 1 = Corresponding port D pin configured to have internal pullup  
 0 = Corresponding port D pin has internal pullup disconnected

## 13.7 Port E

Port E is a 6-bit special-function port that shares two of its pins with the enhanced serial communications interface (ESCI) module.

### 13.7.1 Port E Data Register

The port E data register contains a data latch for each of the six port E pins.



**Figure 13-17. Port E Data Register (PTE)**

### PTE5–PTE0 — Port E Data Bits

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

**NOTE**

*Data direction register E (DDRE) does not affect the data direction of port E pins that are being used by the ESCI module. However, the DDRE bits always determine whether reading port E returns the states of the latches or the states of the pins. See Table 13-6.*

### RxD — SCI Receive Data Input

The PTE1/RxD pin is the receive data input for the ESCI module.

When the enable SCI bit, ENSCI, is clear, the ESCI module is disabled, and the PTE1/RxD pin is available for general-purpose I/O. See Chapter 14 Enhanced Serial Communications Interface (ESCI) Module.

### TxD — SCI Transmit Data Output

The PTE0/TxD pin is the transmit data output for the ESCI module. When the enable SCI bit, ENSCI, is clear, the ESCI module is disabled, and the PTE0/TxD pin is available for general-purpose I/O. See Chapter 14 Enhanced Serial Communications Interface (ESCI) Module.

To initiate an ESCI transmission:

1. Enable the ESCI by writing a 1 to the enable ESCI bit (ENSCI) in ESCI control register 1 (SCC1).
2. Enable the transmitter by writing a 1 to the transmitter enable bit (TE) in ESCI control register 2 (SCC2).
3. Clear the ESCI transmitter empty bit (SCTE) by first reading ESCI status register 1 (SCS1) and then writing to the SCDR. For 9-bit data, also write the T8 bit in SCC3.
4. Repeat step 3 for each subsequent transmission.

At the start of a transmission, transmitter control logic automatically loads the transmit shift register with a preamble of 1s. After the preamble shifts out, control logic transfers the SCDR data into the transmit shift register. A 0 start bit automatically goes into the least significant bit (LSB) position of the transmit shift register. A 1 stop bit goes into the most significant bit (MSB) position.

The ESCI transmitter empty bit, SCTE, in SCS1 becomes set when the SCDR transfers a byte to the transmit shift register. The SCTE bit indicates that the SCDR can accept new data from the internal data bus. If the ESCI transmit interrupt enable bit, SCTIE, in SCC2 is also set, the SCTE bit generates a transmitter CPU interrupt request.

When the transmit shift register is not transmitting a character, the TxD pin goes to the idle condition, high. If at any time software clears the ENSCI bit in ESCI control register 1 (SCC1), the transmitter and receiver relinquish control of the port E pins.

#### 14.4.2.3 Break Characters

Writing a 1 to the send break bit, SBK, in SCC2 loads the transmit shift register with a break character. For TXINV = 0 (output not inverted), a transmitted break character contains all 0s and has no start, stop, or parity bit. Break character length depends on the M bit in SCC1 and the LINR bits in SCBR. As long as SBK is at 1, transmitter logic continuously loads break characters into the transmit shift register. After software clears the SBK bit, the shift register finishes transmitting the last break character and then transmits at least one 1. The automatic 1 at the end of a break character guarantees the recognition of the start bit of the next character.

When LINR is cleared in SCBR, the ESCI recognizes a break character when a start bit is followed by eight or nine 0 data bits and a 0 where the stop bit should be, resulting in a total of 10 or 11 consecutive 0 data bits. When LINR is set in SCBR, the ESCI recognizes a break character when a start bit is followed by 9 or 10 0 data bits and a 0 where the stop bit should be, resulting in a total of 11 or 12 consecutive 0 data bits.

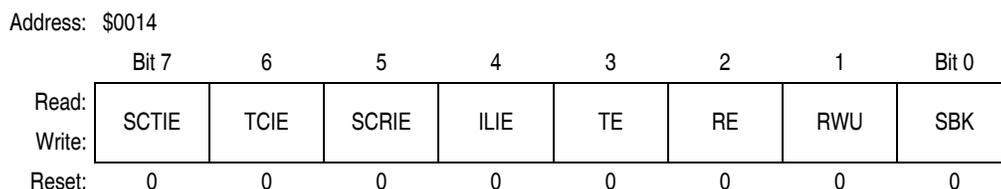
Receiving a break character has these effects on ESCI registers:

- Sets the framing error bit (FE) in SCS1
- Sets the ESCI receiver full bit (SCRF) in SCS1
- Clears the ESCI data register (SCDR)
- Clears the R8 bit in SCC3
- Sets the break flag bit (BKF) in SCS2
- May set the overrun (OR), noise flag (NF), parity error (PE), or reception in progress flag (RPF) bits

## 14.8.2 ESCI Control Register 2

ESCI control register 2 (SCC2):

- Enables these CPU interrupt requests:
  - SCTE bit to generate transmitter CPU interrupt requests
  - TC bit to generate transmitter CPU interrupt requests
  - SCRF bit to generate receiver CPU interrupt requests
  - IDLE bit to generate receiver CPU interrupt requests
- Enables the transmitter
- Enables the receiver
- Enables ESCI wakeup
- Transmits ESCI break characters



**Figure 14-11. ESCI Control Register 2 (SCC2)**

### SCTIE — ESCI Transmit Interrupt Enable Bit

This read/write bit enables the SCTE bit to generate ESCI transmitter CPU interrupt requests. Setting the SCTIE bit in SCC2 enables the SCTE bit to generate CPU interrupt requests. Reset clears the SCTIE bit.

- 1 = SCTE enabled to generate CPU interrupt
- 0 = SCTE not enabled to generate CPU interrupt

### TCIE — Transmission Complete Interrupt Enable Bit

This read/write bit enables the TC bit to generate ESCI transmitter CPU interrupt requests. Reset clears the TCIE bit.

- 1 = TC enabled to generate CPU interrupt requests
- 0 = TC not enabled to generate CPU interrupt requests

### SCRIE — ESCI Receive Interrupt Enable Bit

This read/write bit enables the SCRF bit to generate ESCI receiver CPU interrupt requests. Setting the SCRIE bit in SCC2 enables the SCRF bit to generate CPU interrupt requests. Reset clears the SCRIE bit.

- 1 = SCRF enabled to generate CPU interrupt
- 0 = SCRF not enabled to generate CPU interrupt

### ILIE — Idle Line Interrupt Enable Bit

This read/write bit enables the IDLE bit to generate ESCI receiver CPU interrupt requests. Reset clears the ILIE bit.

- 1 = IDLE enabled to generate CPU interrupt requests
- 0 = IDLE not enabled to generate CPU interrupt requests

### 14.8.4 ESCI Status Register 1

ESCI status register 1 (SCS1) contains flags to signal these conditions:

- Transfer of SCDR data to transmit shift register complete
- Transmission complete
- Transfer of receive shift register data to SCDR complete
- Receiver input idle
- Receiver overrun
- Noisy data
- Framing error
- Parity error

Address: \$0016

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	SCTE	TC	SCRF	IDLE	OR	NF	FE	PE
Write:								
Reset:	1	1	0	0	0	0	0	0

= Unimplemented

**Figure 14-13. ESCI Status Register 1 (SCS1)**

#### SCTE — ESCI Transmitter Empty Bit

This clearable, read-only bit is set when the SCDR transfers a character to the transmit shift register. SCTE can generate an ESCI transmitter CPU interrupt request. When the SCTIE bit in SCC2 is set, SCTE generates an ESCI transmitter CPU interrupt request. In normal operation, clear the SCTE bit by reading SCS1 with SCTE set and then writing to SCDR. Reset sets the SCTE bit.

- 1 = SCDR data transferred to transmit shift register
- 0 = SCDR data not transferred to transmit shift register

#### TC — Transmission Complete Bit

This read-only bit is set when the SCTE bit is set, and no data, preamble, or break character is being transmitted. TC generates an ESCI transmitter CPU interrupt request if the TCIE bit in SCC2 is also set. TC is cleared automatically when data, preamble, or break is queued and ready to be sent. There may be up to 1.5 transmitter clocks of latency between queueing data, preamble, and break and the transmission actually starting. Reset sets the TC bit.

- 1 = No transmission in progress
- 0 = Transmission in progress

#### SCRF — ESCI Receiver Full Bit

This clearable, read-only bit is set when the data in the receive shift register transfers to the ESCI data register. SCRF can generate an ESCI 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

### 14.8.8 ESCI Prescaler Register

The ESCI prescaler register (SCPSC) together with the ESCI baud rate register selects the baud rate for both the receiver and the transmitter.

**NOTE**

*There are two prescalers available to adjust the baud rate. One in the ESCI baud rate register and one in the ESCI prescaler register.*

Address: \$0009

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	PDS2	PDS1	PDS0	PSSB4	PSSB3	PSSB2	PSSB1	PSSB0
Write:								
Reset:	0	0	0	0	0	0	0	0

**Figure 14-18. ESCI Prescaler Register (SCPSC)**

#### PDS2–PDS0 — Prescaler Divisor Select Bits

These read/write bits select the prescaler divisor as shown in Table 14-9. Reset clears PDS2–PDS0.

**NOTE**

*The setting of '000' will bypass not only this prescaler but also the prescaler divisor fine adjust (PDFA). It is not recommended to bypass the prescaler while ENSCI is set, because the switching is not glitch free.*

**Table 14-9. ESCI Prescaler Division Ratio**

PDS[2:1:0]	Prescaler Divisor (PD)
0 0 0	Bypass this prescaler
0 0 1	2
0 1 0	3
0 1 1	4
1 0 0	5
1 0 1	6
1 1 0	7
1 1 1	8

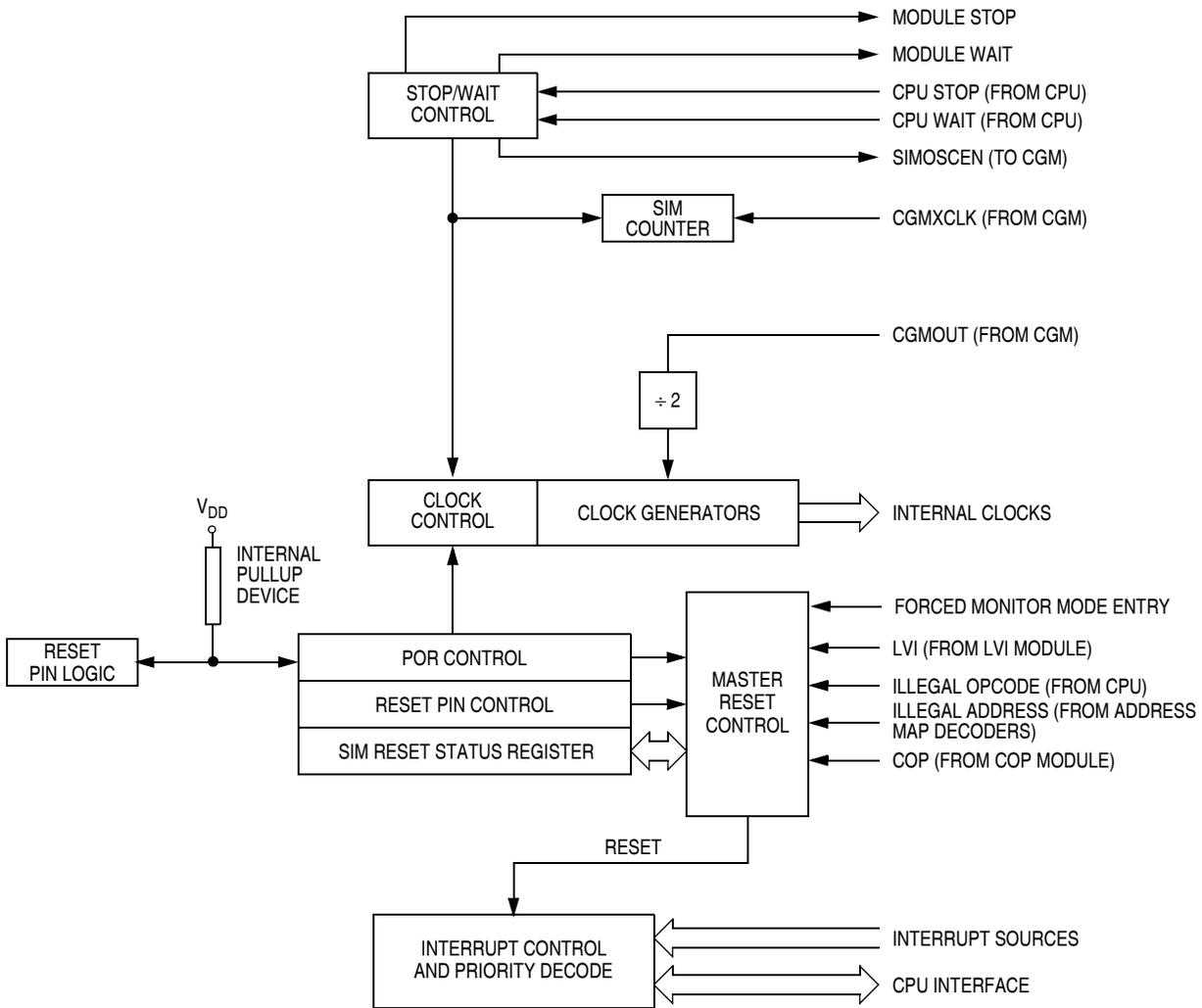
#### PSSB4–PSSB0 — Clock Insertion Select Bits

These read/write bits select the number of clocks inserted in each 32 output cycle frame to achieve more timing resolution on the **average** prescaler frequency as shown in Table 14-10. Reset clears PSSB4–PSSB0.

**Table 14-10. ESCI Prescaler Divisor Fine Adjust**

PSSB[4:3:2:1:0]	Prescaler Divisor Fine Adjust (PDFA)
0 0 0 0 0	0/32 = 0
0 0 0 0 1	1/32 = 0.03125
0 0 0 1 0	2/32 = 0.0625
0 0 0 1 1	3/32 = 0.09375
0 0 1 0 0	4/32 = 0.125

# System Integration Module (SIM)



**Figure 15-1. SIM Block Diagram**

# Chapter 18

## Timer Interface Module (TIM1)

### 18.1 Introduction

This section describes the timer interface module (TIM1). TIM1 is a two-channel timer that provides a timing reference with input capture, output compare, and pulse-width-modulation functions. Figure 18-2 is a block diagram of the TIM1.

### 18.2 Features

Features of the TIM1 include the following:

- Two input capture/output compare channels
  - Rising-edge, falling-edge, or any-edge input capture trigger
  - Set, clear, or toggle output compare action
- Buffered and unbuffered pulse width modulation (PWM) signal generation
- Programmable TIM1 clock input with 7-frequency internal bus clock prescaler selection
- Free-running or modulo up-count operation
- Toggle any channel pin on overflow
- TIM1 counter stop and reset bits

### 18.3 Functional Description

Figure 18-2 shows the structure of the TIM1. The central component of the TIM1 is the 16-bit TIM1 counter that can operate as a free-running counter or a modulo up-counter. The TIM1 counter provides the timing reference for the input capture and output compare functions. The TIM1 counter modulo registers, T1MODH:T1MODL, control the modulo value of the TIM1 counter. Software can read the TIM1 counter value at any time without affecting the counting sequence.

The two TIM1 channels are programmable independently as input capture or output compare channels.

#### 18.3.1 TIM1 Counter Prescaler

The TIM1 clock source is one of the seven prescaler outputs. The prescaler generates seven clock rates from the internal bus clock. The prescaler select bits, PS[2:0], in the TIM1 status and control register (T1SC) select the TIM1 clock source.

#### 18.3.2 Input Capture

With the input capture function, the TIM1 can capture the time at which an external event occurs. When an active edge occurs on the pin of an input capture channel, the TIM1 latches the contents of the TIM1 counter into the TIM1 channel registers, T1CHxH:T1CHxL. The polarity of the active edge is programmable. Input captures can generate TIM1 central processor unit (CPU) interrupt requests.

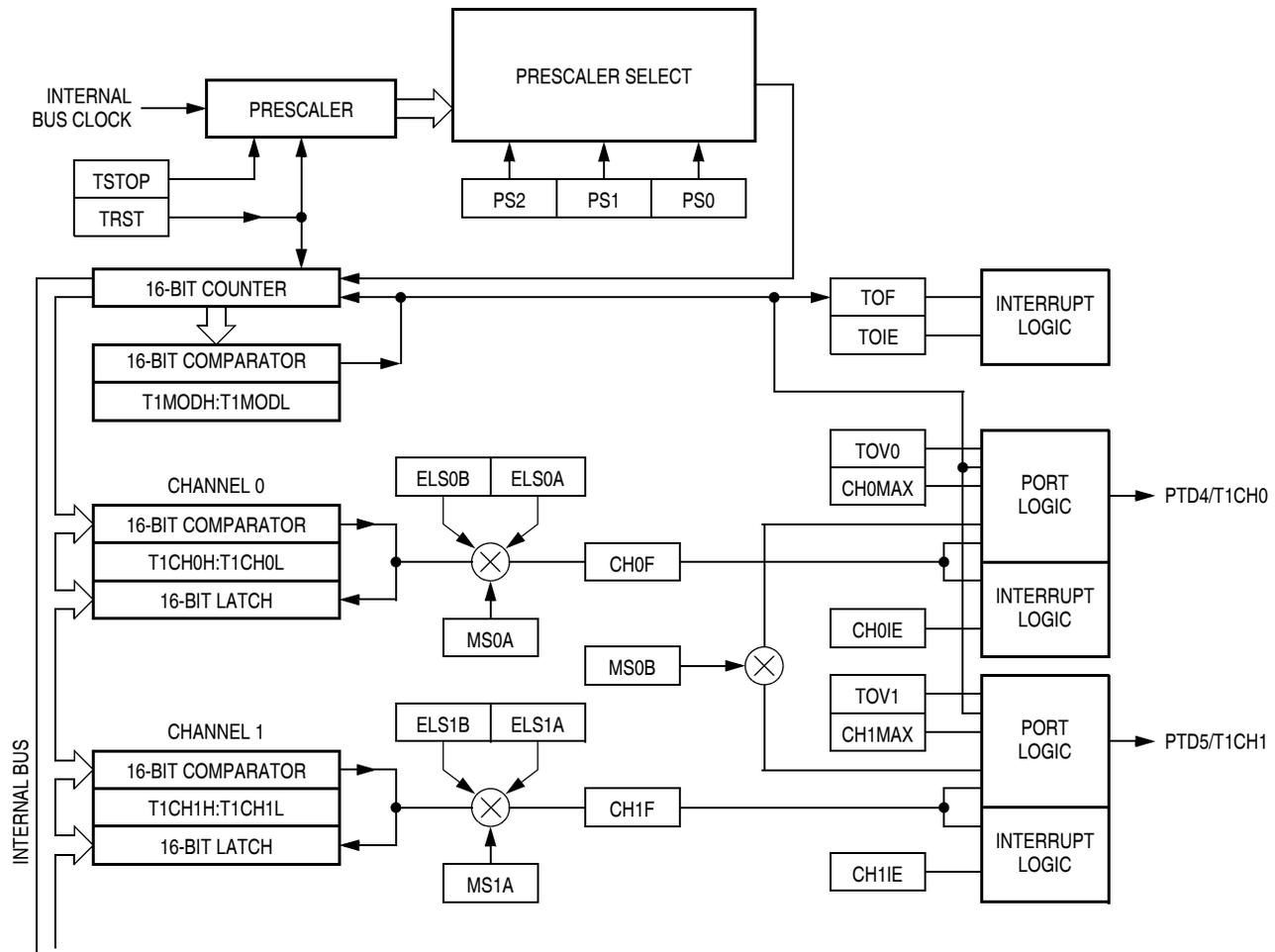


Figure 18-2. TIM1 Block Diagram

### 18.3.3 Output Compare

With the output compare function, the TIM1 can generate a periodic pulse with a programmable polarity, duration, and frequency. When the counter reaches the value in the registers of an output compare channel, the TIM1 can set, clear, or toggle the channel pin. Output compares can generate TIM1 CPU interrupt requests.

#### 18.3.3.1 Unbuffered Output Compare

Any output compare channel can generate unbuffered output compare pulses as described in 18.3.3 Output Compare. The pulses are unbuffered because changing the output compare value requires writing the new value over the old value currently in the TIM1 channel registers.

An unsynchronized write to the TIM1 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 TIM1 overflow interrupt routine to write a new, smaller output compare value may cause the compare to be missed. The TIM1 may pass the new value before it is written.

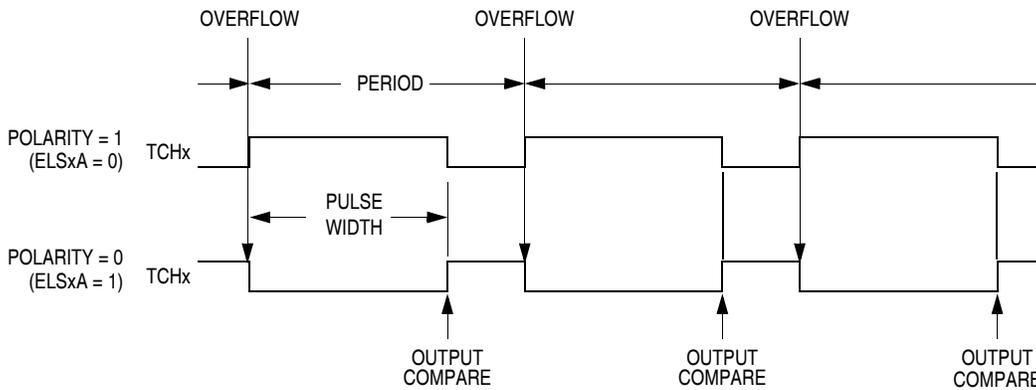


Figure 18-4. PWM Period and Pulse Width

### 18.3.4.1 Unbuffered PWM Signal Generation

Any output compare channel can generate unbuffered PWM pulses as described in 18.3.4 Pulse Width Modulation (PWM). The pulses are unbuffered because changing the pulse width requires writing the new pulse width value over the old value currently in the TIM1 channel registers.

An unsynchronized write to the TIM1 channel registers to change a pulse width value could cause incorrect operation for up to two PWM 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 PWM period. Also, using a TIM1 overflow interrupt routine to write a new, smaller pulse width value may cause the compare to be missed. The TIM1 may pass the new value before it is written to the timer channel (T1CHxH:T1CHxL) registers.

Use the following methods to synchronize unbuffered changes in the PWM pulse width on channel x:

- When changing to a shorter pulse width, 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 pulse. The interrupt routine has until the end of the PWM period to write the new value.
- When changing to a longer pulse width, enable TIM1 overflow interrupts and write the new value in the TIM1 overflow interrupt routine. The TIM1 overflow interrupt occurs at the end of the current PWM 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 PWM period.

**NOTE**

*In PWM signal generation, do not program the PWM channel to toggle on output compare. Toggling on output compare prevents reliable 0% duty cycle generation and removes the ability of the channel to self-correct in the event of software error or noise. Toggling on output compare also can cause incorrect PWM signal generation when changing the PWM pulse width to a new, much larger value.*

**Table 20-7. 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
<b>Command Sequence</b>	

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

Description	Executes PULH and RTI instructions
Operand	None
Data Returned	None
Opcode	\$28
<b>Command Sequence</b>	

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.

	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

**Figure 20-16. Stack Pointer at Monitor Mode Entry**

## 21.11 3.3-Volt ADC Characteristics

Characteristic <sup>(1)</sup>	Symbol	Min	Max	Unit	Comments
Supply voltage	$V_{DDAD}$	3.0	3.6	V	$V_{DDAD}$ should be tied to the same potential as $V_{DD}$ via separate traces.
Input voltages	$V_{ADIN}$	0	$V_{DDAD}$	V	$V_{ADIN} \leq V_{DDAD}$
Resolution	$B_{AD}$	10	10	Bits	
Absolute accuracy	$A_{AD}$	-6	+6	LSB	Includes quantization
ADC internal clock	$f_{ADIC}$	500 k	1.048 M	Hz	$t_{AIC} = 1/f_{ADIC}$
Conversion range	$R_{AD}$	$V_{SSAD}$	$V_{DDAD}$	V	
Power-up time	$t_{ADPU}$	16	—	$t_{AIC}$ cycles	
Conversion time	$t_{ADC}$	16	17	$t_{AIC}$ cycles	
Sample time	$t_{ADS}$	5	—	$t_{AIC}$ cycles	
Monotonicity	$M_{AD}$	Guaranteed			
Zero input reading	$Z_{ADI}$	000	005	Hex	$V_{ADIN} = V_{SSA}$
Full-scale reading	$F_{ADI}$	3FA	3FF	Hex	$V_{ADIN} = V_{DDA}$
Input capacitance	$C_{ADI}$	—	30	pF	Not tested
$V_{DDAD}/V_{REFH}$ current	$I_{VREF}$	—	1.2	mA	
Absolute accuracy (8-bit truncation mode)	$A_{AD}$	-1	+1	LSB	Includes quantization
Quantization error (8-bit truncation mode)	—	-1/8	+7/8	LSB	

1.  $V_{DD} = 3.3 \text{ Vdc} \pm 10\%$ ,  $V_{SS} = 0 \text{ Vdc}$ ,  $V_{DDAD}/V_{REFH} = 3.3 \text{ Vdc} \pm 10\%$ ,  $V_{SSAD}/V_{REFL} = 0 \text{ Vdc}$

## **How to Reach Us:**

### **Home Page:**

[www.freescale.com](http://www.freescale.com)

### **E-mail:**

[support@freescale.com](mailto: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](mailto: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](mailto: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](mailto: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](mailto: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](mailto:LDCForFreescaleSemiconductor@hibbertgroup.com)

RoHS-compliant and/or Pb-free versions of Freescale products have the functionality and electrical characteristics of their non-RoHS-compliant and/or non-Pb-free counterparts. For further information, see <http://www.freescale.com> or contact your Freescale sales representative.

For information on Freescale's Environmental Products program, go to <http://www.freescale.com/epp>.

Information in this document is provided solely to enable system and software implementers to use Freescale Semiconductor products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.

Freescale Semiconductor reserves the right to make changes without further notice to any products herein. Freescale Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters that may be provided in Freescale Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals", must be validated for each customer application by customer's technical experts. Freescale Semiconductor does not convey any license under its patent rights nor the rights of others. Freescale Semiconductor products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Freescale Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use Freescale Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold Freescale Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Freescale Semiconductor was negligent regarding the design or manufacture of the part.

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners.

© Freescale Semiconductor, Inc. 2005, 2006, 2007. All rights reserved.