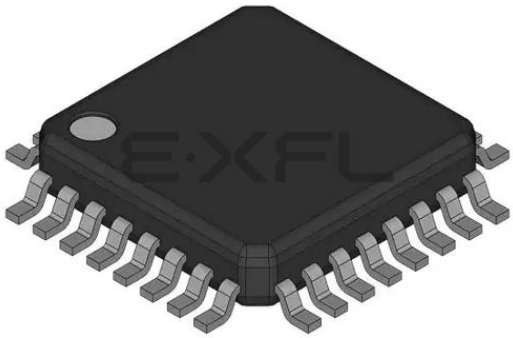


Welcome to E-XFL.COM

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	Active
Core Processor	HC08
Core Size	8-Bit
Speed	4MHz
Connectivity	-
Peripherals	LVD, POR, PWM, RF Mod
Number of I/O	12
Program Memory Size	2KB (2K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	128 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	-
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/pro/item?MUrl=&PartUrl=mc68hc908rf2mfa

Revision History

Revision History

Date	Revision Level	Description	Page Number(s)
June, 2001	1.0	First bulleted paragraph under the subsection **11.5 Interrupts** reworded for clarity	137
		Revision to the description of the CHxMAX bit and the note that follows that description	145
		14.2 Absolute Maximum Ratings — ESD HBM and ESD MM entries added	171
		14.8 UHF Transmitter Module — Table entries and figures revised throughout	176
		14.11 LVI Characteristics — V_{LVS} and V_{LVR} specifications updated — low voltage reset and detection entries deleted	183
		14.12 Memory Characteristics — Maximum value for FLASH page program pulses updated	184
December, 2001	2.0	**Section 11. Timer Interface Module (TIM)** — Timer discrepancies corrected throughout this section.	129
March, 2004	3.0	Document reformatted to meet current publications standards	Throughout
May, 2004	4.0	**1.2 Features** — Added –40°C to 125°C operation.	16
		14.3 Functional Operating Range — Corrected oprating temperature range	172
		14.5 1.8-Volt to 3.3-Volt DC Electrical Characteristics Excluding UHF Module — Added values for Stop I_{DD} at –40°C to 125°C and corrected Note 1	173
		14.6 3.0-Volt DC Electrical Characteristics Excluding UHF Module — Added values for Stop I_{DD} at –40°C to 125°C and corrected Note 1	174
		14.7 2.0-Volt DC Electrical Characteristics Excluding UHF Module — Added values for Stop I_{DD} at –40°C to 125°C and corrected Note 1	175
		Table 15-1. MC Order Numbers — Added ordering information for –40°C to 125°C temperature range	185

Section 6. Internal Clock Generator Module (ICG)

Table of Contents

1.4 Pin Assignments

Figure 1-2 shows the pin assignments.

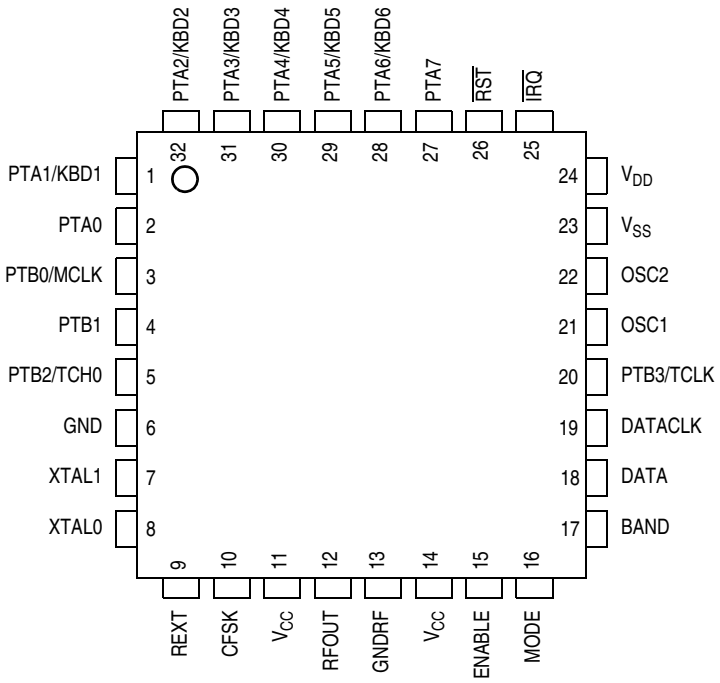


Figure 1-2. LQFP Pin Assignments

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

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

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

2.4 Random-Access Memory (RAM)

Addresses $0080–$00FF are RAM locations. The location of the stack RAM is programmable.

NOTE: *For correct operation, the stack pointer must point only to RAM locations.*

Before processing an interrupt, the CPU uses five bytes of the stack to save the contents of the CPU registers.

NOTE: *For M68HC05, M6805, and M146805 compatibility, the H register is not stacked.*

During a subroutine call, the CPU uses two bytes of the stack to store the return address. The stack pointer decrements during pushes and increments during pulls.

NOTE: *Be careful when using nested subroutines. The CPU could overwrite data in the RAM during a subroutine or during the interrupt stacking operation.*

2.5 FLASH 2TS Memory

This section describes the operation of the embedded FLASH 2TS memory. This memory can be read, programmed, and erased from a single external supply. The program and erase operations are enabled through the use of an internal charge pump.

The FLASH 2TS memory is appropriately named to describe its 2-transistor source-select bit cell. The FLASH 2TS memory is an array of 2031 bytes with an additional 14 bytes of user vectors and one byte for block protection. An erased bit reads as a 0 and a programmed bit reads as a 1.

The address ranges for the user memory, control register, and vectors are:

- $7800–$7FEE, user space
- $7FEF, reserved — optional ICG trim value, see **6.7.3 ICG Trim Register**
- $FFF0, block protect register
- $FE08, FLASH 2TS control register
- $FFF2–$FFFF, these locations are reserved for user-defined interrupt and reset vectors

This list is the row architecture for the user space array:

$7800–$7807 (Row 0)

$7808–$780F (Row 1)

$7810–$7817 (Row 2)

$7818–$781F (Row 3)

$7820–$7827 (Row 4)

$7FE8–$7FEF (Row 253)

The functions shown in **Table 2-4** accept data through the CPU registers and global variables in RAM. **Table 2-5** shows the RAM locations that are used for passing parameters.

Table 2-5. Embedded FLASH Routine Global Variables

Variable Location	Variable Address(1)
CTRLBYT	RAMSTART + 8
CPUSPD	RAMSTART + 9
LADDR	RAMSTART + 10
BUMPS	RAMSTART + 12
DERASE	RAMSTART + 13
DATA	RAMSTART + 15

1. RAMSTART is defined as the starting address of the RAM in the memory map. This is address $0080.

2.5.8 Embedded Function Descriptions

This subsection describes the embedded functions.

2.5.8.1 RDVRRNG Routine

Name: RDVRRNG

Purpose: Read and/or verify a range of FLASH memory

Entry conditions:

H:X	Contains the first address of the range
LADDR	Contains the last address of the range
DATA	Contains the data to compare the read data against for read/verify to RAM only (length is user determined)
ACC	Non-zero for read/verify to RAM, 0 for output to PA0

Exit conditions:

C bit	Set if good compare for read/verify to RAM only
ACC	Contains checksum
DATA	Contains read FLASH data for read/verify to RAM only

Ready/verify RAM option: This subroutine both compares data passed in the DATA array to the FLASH data and reads the data from FLASH into the DATA array. It also calculates the checksum of the data.

Output to PA0 option: This subroutine dumps the data from the range to PA0 in the same format as monitor data. It also calculates the checksum of the data.

NOTE: *This serial dump does not circumvent security because the security vectors must still be passed to make FLASH readable in monitor mode.*

The row whose cycling bit is different will be erased and the entire row will be programmed with the given data, including a toggled version of the cycling bit.

2.5.8.5 Example Routine Calls

This code is for illustrative purposes only and does not represent valid syntax for any particular assembler.

```
RAM             EQU     $80
RDVRRNG         EQU     $F000
PRGRNGE         EQU     $F003
ERANRGE         EQU     $F006
REDPROG         EQU     $F009
;*************************************************************
 RAM Definitions for Subroutines
;*************************************************************

ORG     RAM+8

CTRLBYT         RMB     1
CPUSPD          RMB     1
LADDR           RMB     2
BUMPS           RMB     1
DERASE          RMB     2

;Allocation of "DATA" space is dependent on the device and
;application

DATA    RMB     8
;*************************************************************
; CALLING EXAMPLE FOR READ/VERIFY A RANGE (RDVRRNG)
;*************************************************************
LDA     #$FF            ;TARGET IS RAM
LDHX    #$7807          ;END AFTER FIRST ROW
STHX    LADDR
LDHX    #$7800          ;START AT FIRST ROW
JSR     RDVRRNG         ;DATA WILL CONTAIN FLASH INFO
;*************************************************************;
CALLING EXAMPLE FOR ERASE A RANGE (RNGEERA)
;*************************************************************
MOV     #$08,CPUSPD     ;Load Bus frequency in MHz * 4
MOV     #$60,CTRLBYT    ;Bits 5&6 hold the block size to erase
                        ;00  Full Array
                        ;20  One-Half Array
                        ;40  Eight Rows
                        ;60  Singe Row
                        ;Remember a Row is 1 byte

                        ;Set erase time in uS/24, number in
                        ;decimal
LDHX    #100000/24
STHX    DERASE
```

Table 5-1. Instruction Set Summary (Sheet 7 of 7)

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

5.8 Opcode Map

See **Table 5-2**.

6.3.1 Clock Enable Circuit

The clock enable circuit is used to enable the internal clock (ICLK) or external clock (ECLK). The clock enable circuit generates an ICG stop (ICGSTOP) signal which stops all clocks (ICLK, ECLK, and the low-frequency base clock, IBASE) low. ICGSTOP is set and the ICG is disabled in stop mode.

The internal clock enable signal (ICGEN) turns on the internal clock generator which generates ICLK. ICGEN is set (active) whenever the ICGON bit is set and the ICGSTOP signal is clear. When ICGEN is clear, ICLK and IBASE are both low.

The external clock enable signal (ECGEN) turns on the external clock generator which generates ECLK. ECGEN is set (active) whenever the ECGON bit is set and the ICGSTOP signal is clear. When ECGEN is clear, ECLK is low.

6.3.2 Internal Clock Generator

The internal clock generator, shown in **Figure 6-2**, creates a low-frequency base clock (IBASE), which operates at a nominal frequency (f_{NOM}) of 307.2 kHz ±25 percent, and an internal clock (ICLK) which is an integer multiple of IBASE. This multiple is the ICG multiplier factor (N), which is programmed in the ICG multiplier register (ICGMR). The internal clock generator is turned off and the output clocks (IBASE and ICLK) are held low when the internal clock generator enable signal (ICGEN) is clear.

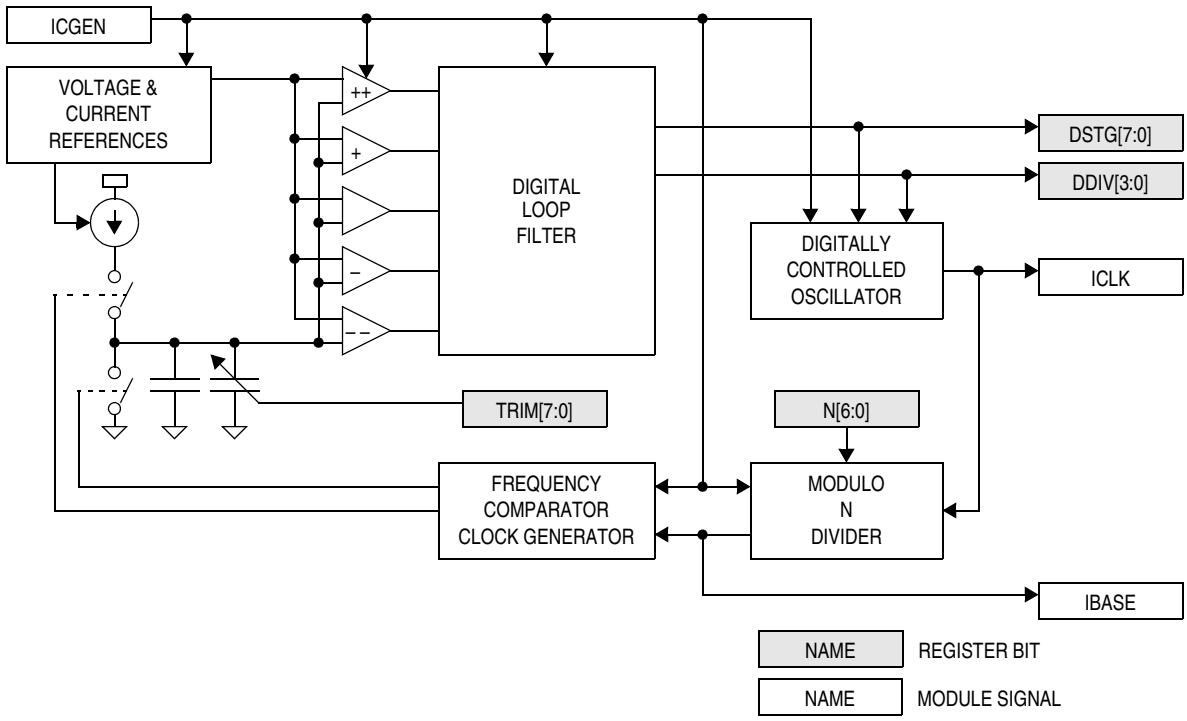


Figure 6-2. Internal Clock Generator Block Diagram

6.3.4.2 Internal Clock Activity Detector

The internal clock activity detector looks for at least one falling edge on the low-frequency base clock (IBASE) every time the external reference (EREF) is low. Since EREF is less than half the frequency of IBASE, this should occur every time. If it does not occur two consecutive times, the internal clock inactivity indicator (IOFF) is set. IOFF will be cleared the next time there is a falling edge of IBASE while EREF is low.

The internal clock stable bit (ICGS) is set when IBASE is within approximately 5 percent of the target 307.2 kHz ±25 percent for two consecutive measurements. ICGS is cleared when IBASE is outside the 5 percent of the target 307.2 kHz ±25 percent, the internal clock generator is disabled (ICGEN is clear), or when IOFF is set.

6.3.4.3 External Clock Activity Detector

The external clock activity detector looks for at least one falling edge on the external clock (ECLK) every time the internal reference (IREF) is low. Since IREF is less than half the frequency of ECLK, this should occur every time. If it does not occur two consecutive times, the external clock inactivity indicator (EOFF) is set. EOFF will be cleared the next time there is a falling edge of ECLK while IREF is low.

The external clock stable bit (ECGS) is also generated in the external clock activity detector. ECGS is set on a falling edge of the external stabilization clock (ESTBCLK). This will be 4096 ECLK cycles after the external clock generator on bit (ECGON) is set. ECGS is cleared when the external clock generator is disabled (ECGON is clear) or when EOFF is set.

6.3.5 Clock Selection Circuit

The clock selection circuit, shown in **Figure 6-5**, contains two clock switches which generate the oscillator output clock (CGMXCLK) from either the internal clock (ICLK) or the external clock (ECLK). The clock selection circuit also contains a divide-by-two circuit which creates the clock generator output clock (CGMOUT), which generates the bus clocks.

6.3.5.1 Clock Selection Switch

The clock select switch creates the oscillator output clock (CGMXCLK) from either the internal clock (ICLK) or the external clock (ECLK), based on the clock select bit (CS; set selects ECLK, clear selects ICLK). When switching the CS bit, both ICLK and ECLK must be on (ICGON and ECGON set). The clock being switched to must also be stable (ICGS or ECGS set).

6.6 Configuration Register Option

One configuration register option affects the functionality of the ICG: EXTSLOW (slow external clock).

All configuration register options will have a default setting. Refer to **Section 3. Configuration Register (CONFIG)** on how the configuration register is used.

6.6.1 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). 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.

6.7 I/O Registers

The ICG contains five registers, summarized in **Figure 6-10**. These registers are:

- ICG control register, ICGCR
- ICG multiplier register, ICGMR
- ICG trim register, ICGTR
- ICG DCO divider control register, ICGDVR
- ICG DCO stage control register, ICGDSR

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 6-5**.

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$0036	Internal Clock Generator Control Register (ICGCR) See page 88.	Read:	CMIE	CMF	CMON	CS	ICGON	ICGS	ECGON	ECGS
		Write:								
		Reset:	0	0	0	0	1	0	0	0
$0037	Internal Clock Generator Multiplier Register (ICGMR) See page 90.	Read:	R	N6	N5	N4	N3	N2	N1	N0
		Write:								
		Reset:	0	0	0	1	0	1	0	1

Figure 6-10. ICG I/O Register Summary

9.2 Port A

Port A is an 8-bit special function port that shares six of its pins with the keyboard interrupt module (KBD). PTA6–PTA1 contain pullup resistors enabled when the port pin is enabled as a keyboard interrupt. Port A pins are also high-current port pins with 3-mA sink capabilities.

9.2.1 Port A Data Register

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

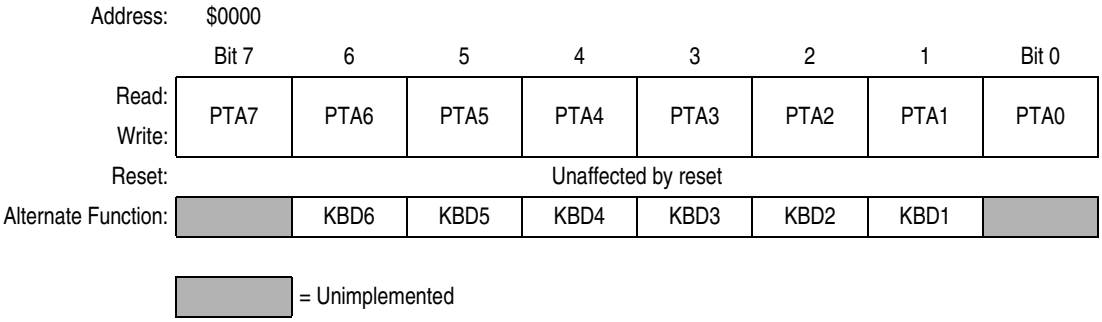


Figure 9-2. Port A Data Register (PTA)

PTA[7:0] — Port A Data Bits

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

KBD[6:1] — Keyboard Wakeup Pins

The keyboard interrupt enable bits, KBIE[6:1], in the keyboard interrupt control register enable the port A pin as external interrupt pins and related internal pullup resistor. See **Section 7. Keyboard/External Interrupt Module (KBI)**.

NOTE: *The enabling of a keyboard interrupt pin will override the corresponding definition of the pin in the data direction register. However, the data direction register bit must be a 0 for software to read the pin.*

Table 9-1. Port A Pin Functions

KBIE[2] Bit	DDRA Bit	PTA Bit	I/O Pin Mode	Accesses to DDRA	Accesses to PTA	
				Read/Write	Read	Write
1	X	X[1]	Input, V_{DD}[4]	DDRA[7:0]	Pin	PTA[7:0][3]
0	0	X	Input, Hi-Z[5]	DDRA[7:0]	Pin	PTA[7:0][3]
0	1	X	Output	DDRA[7:0]	PTA[7:0]	PTA[7:0]

Notes:
1. X = Don't care
2. Keyboard interrupt enable bit (see **7.5.2 Keyboard Interrupt Enable Register**)
3. Writing affects data register, but does not affect input.
4. I/O pin pulled up to V_{DD} by internal pullup device
5. Hi-Z = High impedance

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.*

9.3 Port B

Port B is a 4-bit special function port that shares two of its pins with the timer (TIM) module and one with the buffered internal bus clock MCLK.

9.3.1 Port B Data Register

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

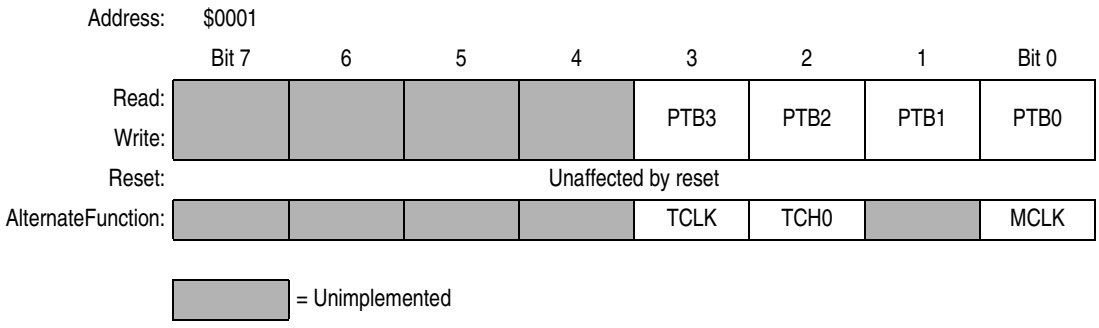


Figure 9-5. Port B Data Register (PTB)

PTB[3:0] — 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.

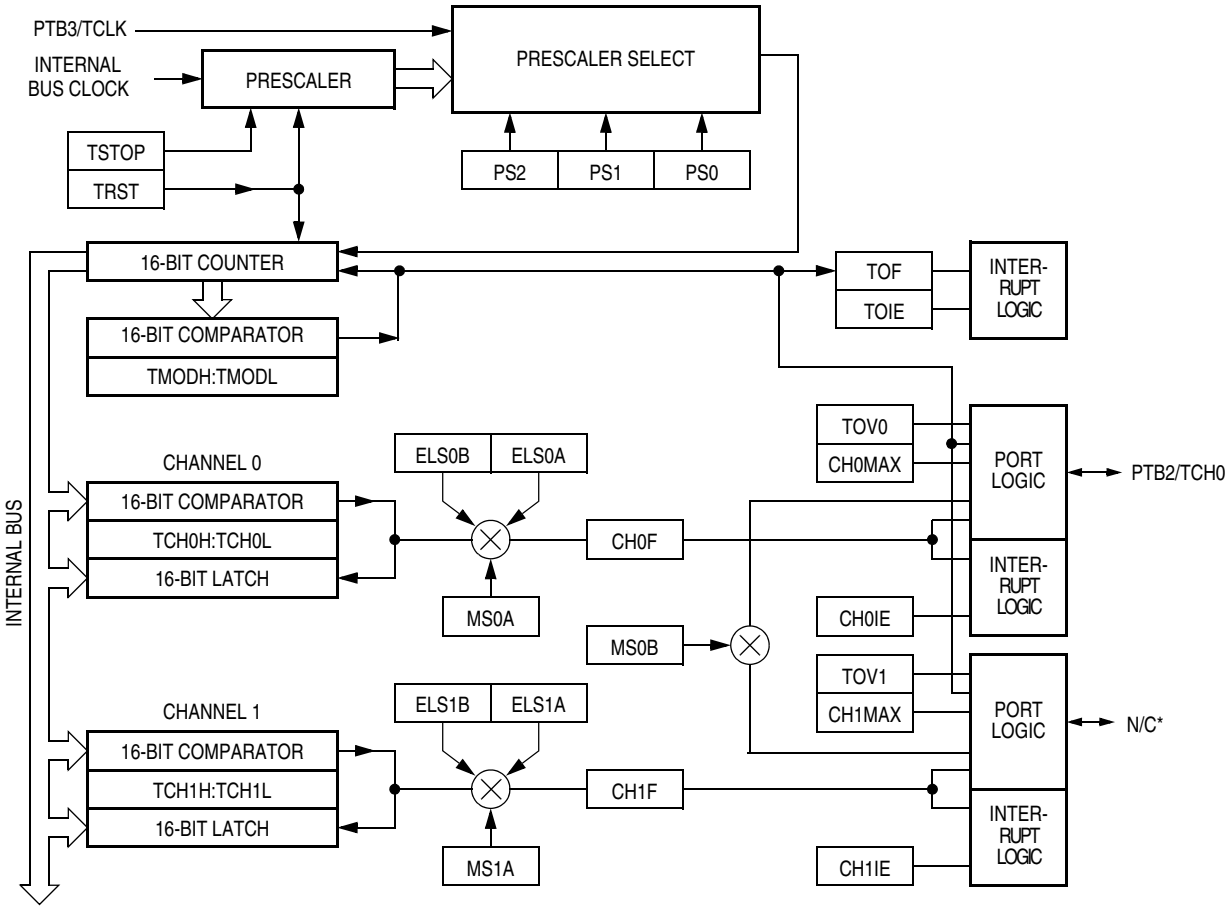
11.4 Functional Description

Figure 11-2 shows the structure of the TIM. The central component of the TIM is the 16-bit TIM counter that can operate as a free-running counter or a modulo up-counter. The TIM counter provides the timing reference for the input capture and output compare functions. The TIM counter modulo registers, TMODH and TMODL, control the modulo value of the TIM counter. Software can read the TIM counter value at any time without affecting the counting sequence.

TIM channel 0 is programmable independently as input capture or output compare. TIM channel 1 is programmable only as output compare. The output compare is detected only through an interrupt.

WARNING: ***TIM channel 1 should not be configured as an input capture because this would result in a floating input that would cause a higher I_{DD}.***

Refer to Figure 11-3 for a summary of the TIM I/O registers.



*This pin is not available on the MC68HC908RF2.

Figure 11-2. TIM Block Diagram

11.4.1 TIM Counter Prescaler

The TIM clock source can be one of the seven prescaler outputs or the TIM clock pin, TCLK. The prescaler generates seven clock rates from the internal bus clock. The prescaler select bits, PS[2:0], in the TIM status and control register select the TIM clock source.

11.4.2 Input Capture

With the input capture function, the TIM can capture the time at which an external event occurs. When an active edge occurs on the pin of an input capture channel, the TIM latches the contents of the TIM counter into the TIM channel registers, TCHxH and TCHxL. The polarity of the active edge is programmable. Input captures can generate TIM CPU interrupt requests.

NOTE: *TIM channel 1 should not be configured in this mode.*

11.4.3 Output Compare

With the output compare function, the TIM 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 TIM channel 0 can set, clear, or toggle the channel pin. Output compares can generate TIM CPU interrupt requests for both TIM channel 0 and TIM channel 1.

NOTE: *TIM channel 1 does not have an external pin associated with it.*

11.4.4 Unbuffered Output Compare

Any output compare channel can generate unbuffered output compare pulses as described in **11.4.3 Output Compare**. The pulses are unbuffered because changing the output compare value requires writing the new value over the old value currently in the TIM channel registers.

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

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

- When changing to a smaller value, enable channel x output compare interrupts and write the new value in the output compare interrupt routine. The output compare interrupt occurs at the end of the current output compare pulse. The interrupt routine has until the end of the counter overflow period to write the new value.

11.8.2 TIM Counter Registers

The two read-only TIM counter registers contain the high and low bytes of the value in the TIM counter. Reading the high byte (TCNTH) latches the contents of the low byte (TCNTL) into a buffer. Subsequent reads of TCNTH do not affect the latched TCNTL value until TCNTL is read. Reset clears the TIM counter registers. Setting the TIM reset bit (TRST) also clears the TIM counter registers

NOTE: *If you read TCNTH during a break interrupt, be sure to unlatch TCNTL by reading TCNTL before exiting the break interrupt. Otherwise, TCNTL retains the value latched during the break.*

Register Name and Address: TCNTH—$0021

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	Bit 15	14	13	12	11	10	9	Bit 8
Write:								
Reset:	0	0	0	0	0	0	0	0

Register Name and Address: TCNTL—$0022

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

= Unimplemented

Figure 11-6. TIM Counter Registers (TCNTH and TCNTL)

Development Support

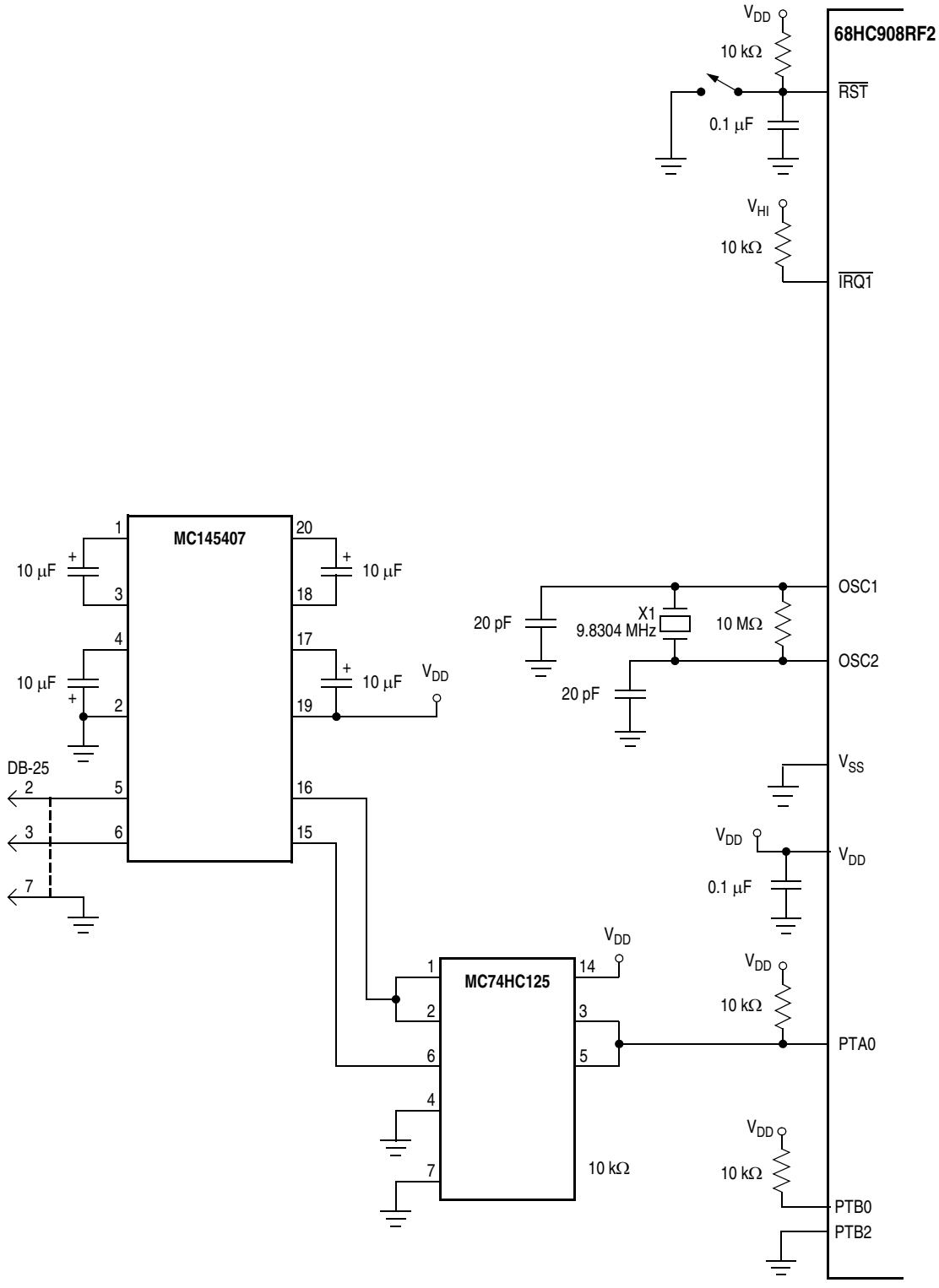


Figure 13-5. Monitor Mode Circuit

Table 13-3. READ (Read Memory) Command

Description	Read byte from memory
Operand	2-byte address in high-byte:low-byte order
Data Returned	Returns contents of specified address
Opcode	$4A

Command Sequence

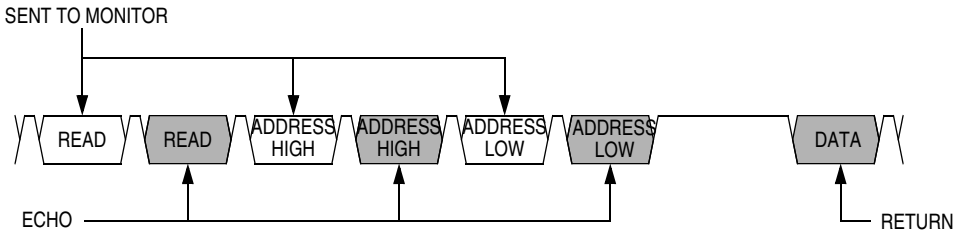


Table 13-4. WRITE (Write Memory) Command

Description	Write byte to memory
Operand	2-byte address in high-byte:low-byte order; low byte followed by data byte
Data Returned	None
Opcode	$49

Command Sequence

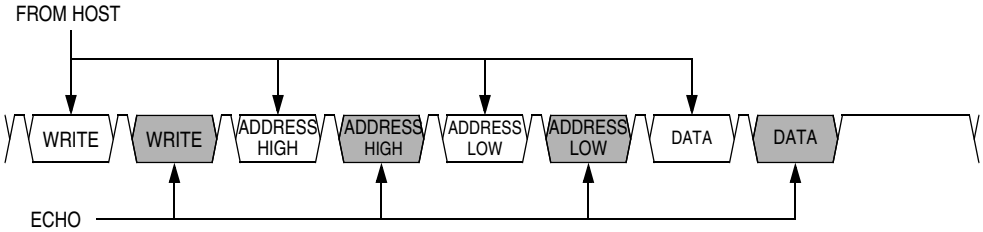


Table 13-5. IREAD (Indexed Read) Command

Description	Read next 2 bytes in memory from last address accessed
Operand	2-byte address in high byte:low byte order
Data Returned	Returns contents of next two addresses
Opcode	$1A

Command Sequence

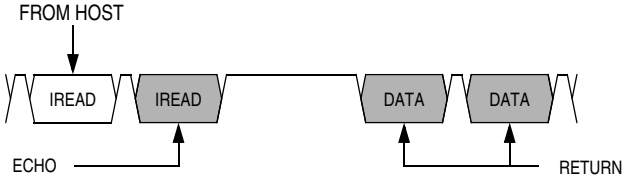
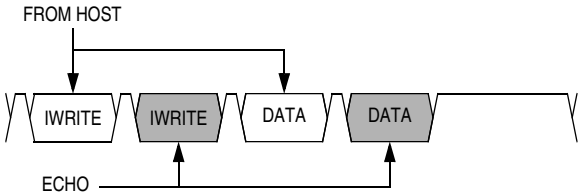


Table 13-6. IWRITE (Indexed Write) Command

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



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

Table 13-7. 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
Command Sequence	

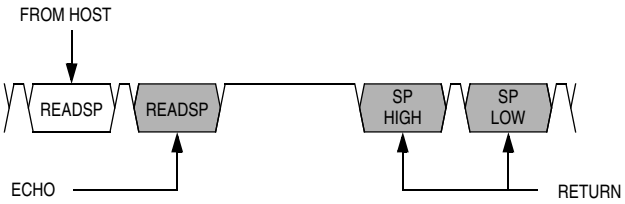


Table 13-8. RUN (Run User Program) Command

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