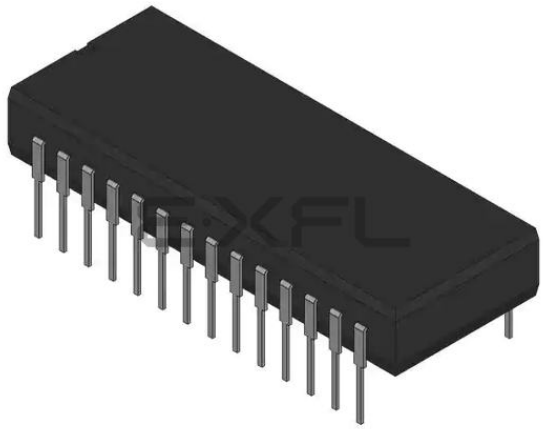


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	HC08
Core Size	8-Bit
Speed	6MHz
Connectivity	SCI, USB
Peripherals	LED, LVD, POR, PWM
Number of I/O	21
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	384 x 8
Voltage - Supply (Vcc/Vdd)	4V ~ 5.5V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	28-SOIC (0.295", 7.50mm Width)
Supplier Device Package	28-SOIC
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mc908jb16dwe

Figure	Title	Page

- Serial communications interface module (SCI)
- Dual clock generator modules (CGM) (32-pin package)
- In-circuit programming capability using USB communication or standard serial link on PTA0 pin
- System protection features:
  - Optional computer operating properly (COP) reset
  - Optional Low-voltage detection with reset
  - Illegal opcode detection with reset
  - Illegal address detection with reset
- Master reset pin with internal pull-up and power-on reset
- $\overline{\text{IRQ}}$ interrupt pin with internal pull-up and schmitt-trigger input
- 32-pin low-profile quad flat pack (LQFP) and 28-pin small outline integrated circuit package (SOIC)

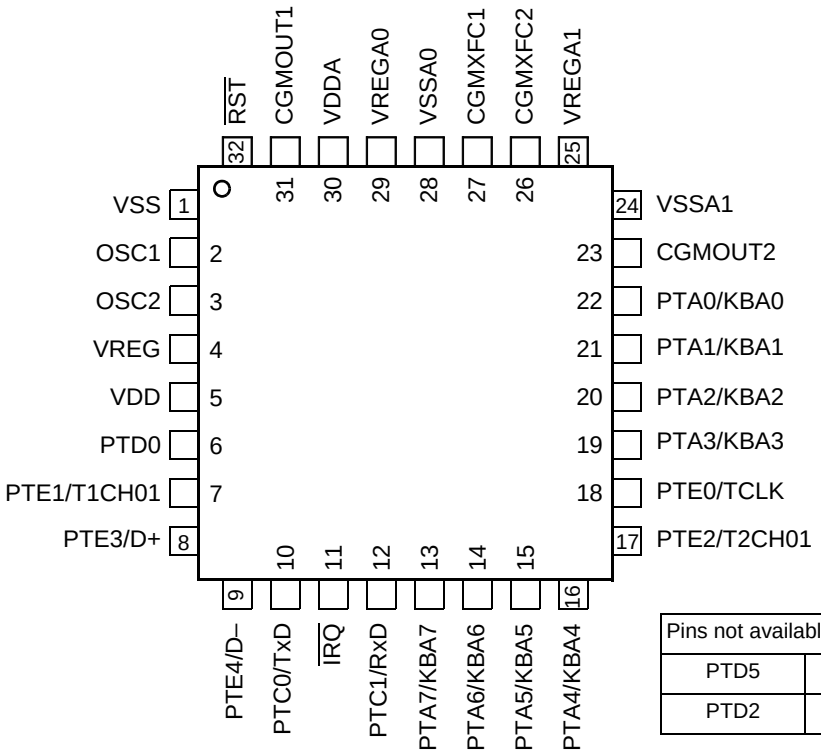
Features of the CPU08 include the following:

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

1.4 MCU Block Diagram

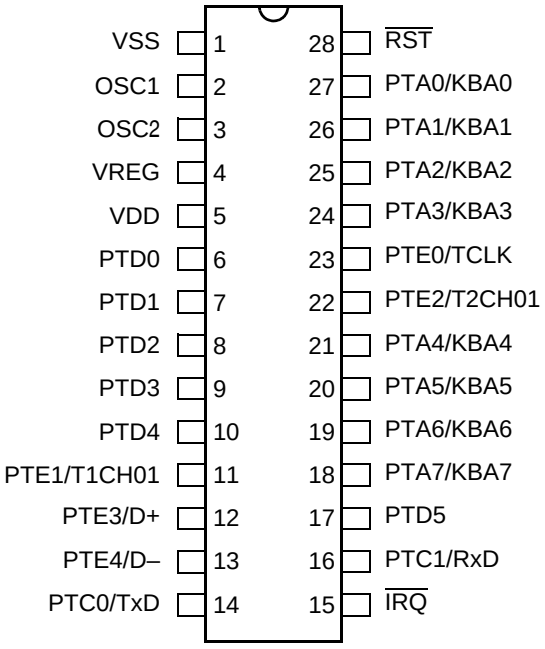
Figure 1-1 shows the structure of the MC68HC908JB16.

1.5 Pin Assignments



Pins not available on 32-pin package:		
PTD5	PTD4	PTD3
PTD2	PTD1	

Figure 1-2. 32-Pin LQFP Pin Assignment



Pins not available on 28-pin package:		
CGMXFC1	CGMXFC2	
CGMOUT1	CGMOUT2	
VREGA0	VREGA1	
VSSA0	VSSA1	VDDA

Figure 1-3. 28-Pin SOIC Pin Assignment

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$FE08	FLASH Control Register (FLCR)	Read:	0	0	0	0	HVEN	MASS	ERASE	PGM
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$FE09	FLASH Block Protect Register (FLBPR)	Read:	BPR7	BPR6	BPR5	BPR4	BPR3	BPR2	BPR1	BPR0
		Write:								
		Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 4-1. FLASH I/O Register Summary

4.3 Functional Description

The FLASH memory consists of an array of 16,384 bytes for user memory plus a block of 48 bytes for user interrupt vectors. *An erased bit reads as logic 1 and a programmed bit reads as a logic 0.* The FLASH memory is block erasable. The minimum erase block size is 512 bytes. Program and erase operation operations are facilitated through control bits in FLASH Control Register (FLCR).The address ranges for the FLASH memory are shown as follows:

- $BA00–$F9FF (user memory, 16,384 bytes)
- $FFD0–$FFFF (user interrupt vectors, 48 bytes)

Programming tools are available from Freescale. Contact your local Freescale representative for more information.

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

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

BPR0 is used only for BPR[7:0] = $FF, for no block protection.

The resultant 16-bit address is used for specifying the start address of the FLASH memory for block protection. The FLASH is protected from this start address to the end of FLASH memory, at $FFFF. With this mechanism, the protect start address can be X000, X200, X400, X600, X800, XA00, XC00, or XE00 within the FLASH memory.

Examples of protect start address:

BPR[7:0]	Start of Address of Protect Range
$00 to $BA	The entire FLASH memory is protected.
$BC (**1011 1100**)	$BC00 (**1011 1100** 0000 0000)
$BE (**1011 1110**)	$BE00 (**1011 1110** 0000 0000)
$C0 (**1100 0000**)	$C000 (**1100 0000** 0000 0000)
$C2 (**1100 0010**)	$C200 (**1100 0010** 0000 0000)
and so on...	
$FE	$FFD0–$FFFF (User vectors)
$FF	The entire FLASH memory is not protected.

Note:
The end address of the protected range is always $FFFF.

4.9 ROM-Resident Routines

ROM-resident routines can be called by a program running in user mode or in monitor mode (see **Section 9. Monitor ROM (MON)**) for FLASH programming, erasing, and verifying. The range of the FLASH memory must be unprotected (see **4.8 FLASH Protection**) before calling the erase or programming routine.

Table 4-1. ROM-Resident Routines

Routine Name	Call Address	Description
VERIFY	$FC03	FLASH verify routine
ERASE	$FC06	FLASH mass or block erase routine
PROGRAM	$FC09	FLASH program routine

The SIM disables the oscillator signals (OSCOUT and OSCDCLK) in stop mode, stopping the CPU and peripherals. Stop recovery time is selectable using the SSREC bit in the configuration register (CONFIG). If SSREC is set, stop recovery is reduced from the normal delay of 4096 OSCDCLK cycles down to 2048. This is ideal for applications using canned oscillators that do not require long startup times from stop mode.

NOTE: *External crystal applications should use the full stop recovery time by clearing the SSREC bit.*

A break interrupt during stop mode sets the SIM break stop/wait bit (SBSW) in the SIM break status register (SBSR).

The SIM counter is held in reset from the execution of the STOP instruction until the beginning of stop recovery. It is then used to time the recovery period. **Figure 8-17** shows stop mode entry timing.

NOTE: *To minimize stop current, all pins configured as inputs should be driven to a logic 1 or logic 0.*

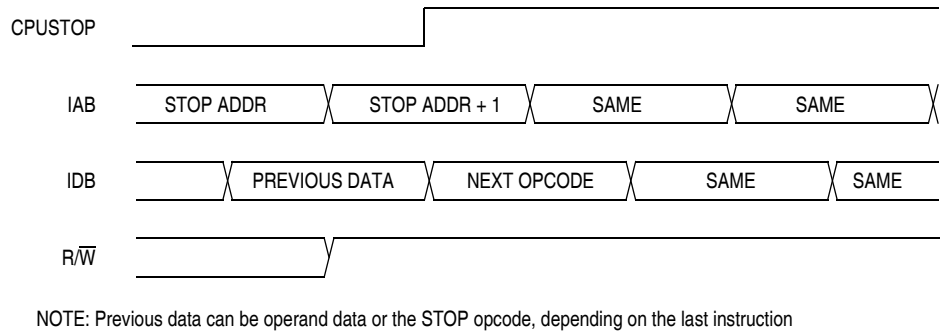


Figure 8-17. Stop Mode Entry Timing

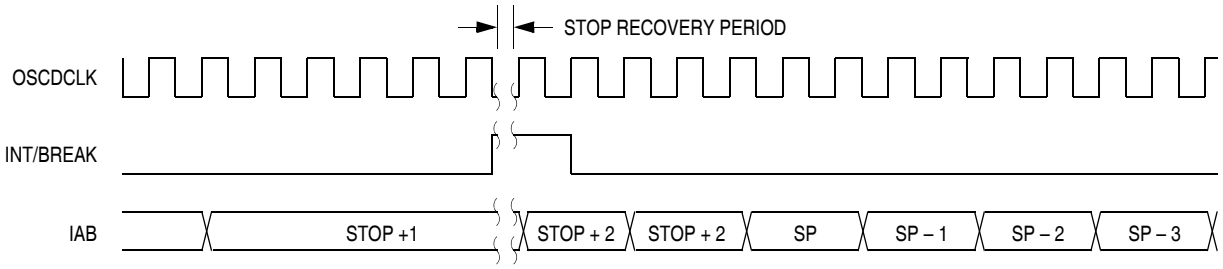


Figure 8-18. Stop Mode Recovery from Interrupt or Break

10.4 Pin Name Conventions

The text that follows describes both timers, TIM1 and TIM2. The TIM input/output (I/O) pin names are T[1,2]CH01 (timer channel 01), where “1” is used to indicate TIM1 and “2” is used to indicate TIM2. The two TIMs share two I/O pins with two I/O port pins. The full names of the TIM I/O pins are listed in **Table 10-1**. The generic pin names appear in the text that follows.

Table 10-1. Pin Name Conventions

TIM Generic Pin Names:		T[1,2]CH01	TCLK
Full TIM Pin Names:	TIM1	PTE1/T1CH01	PTE0/TCLK
	TIM2	PTE2/T2CH01	

NOTE: *References to either timer 1 or timer 2 may be made in the following text by omitting the timer number. For example, TCH01 may refer generically to T1CH01 and T2CH01.*

10.5 Functional Description

Figure 10-1 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:TMODL, control the modulo value of the TIM counter. Software can read the TIM counter value at any time without affecting the counting sequence.

Channel 0 and channel 1 I/Os are connected together, forming a common I/O. Although the two TIM channels are programmable independently as input capture channels, the input capture signal will be the same for both channels. Output compare functions should only be enabled for one channel to avoid I/O contention.

11.7.3 USB Control Logic

The USB control logic manages data movement between the CPU and the transceiver. The control logic handles both transmit and receive operations on the USB. It contains the logic used to manipulate the transceiver and the endpoint registers.

The byte count buffer is loaded with the active transmit endpoints byte count value during transmit operations. This same buffer is used for receive transactions to count the number of bytes received and, upon the end of the transaction, transfer that number to the receive endpoints byte count register.

When transmitting, the control logic handles parallel-to-serial conversion, CRC generation, NRZI encoding, and bit stuffing.

When receiving, the control logic handles sync detection, packet identification, end-of-packet detection, bit (un)stuffing, NRZI decoding, CRC validation, and serial-to-parallel conversion. Errors detected by the control logic include bad CRC, timeout while waiting for EOP, and bit stuffing violations.

11.8 I/O Registers

These I/O registers control and monitor USB operation:

- USB address register (UADDR)
- USB control registers 0–4 (UCR0–UCR4)
- USB status registers 0–1 (USR0–USR1)
- USB interrupt registers 0–2 (UIR0–UIR2)
- USB endpoint 0 data registers 0–7 (UE0D0–UE0D7)
- USB endpoint 1 data registers 0–7 (UE1D0–UE1D7)
- USB endpoint 2 data registers 0–7 (UE2D0–UE2D7)

TXD0F — Endpoint 0 Data Transmit Flag

This read-only bit is set after the data stored in endpoint 0 transmit buffers has been sent and an ACK handshake packet from the host is received. Once the next set of data is ready in the transmit buffers, software must clear this flag by writing a logic 1 to the TXD0FR bit. To enable the next data packet transmission, TX0E also must be set. If the TXD0F bit is not cleared, a NAK handshake will be returned in the next IN transaction.

Reset clears this bit. Writing to TXD0F has no effect.

1 = Transmit on endpoint 0 has occurred
0 = Transmit on endpoint 0 has not occurred

RXD0F — Endpoint 0 Data Receive Flag

This read-only bit is set after the USB module has received a data packet and responded with an ACK handshake packet. Software must clear this flag by writing a logic 1 to the RXD0FR bit after all of the received data has been read. Software also must set the RX0E bit to 1 to enable the next data packet reception. If the RXD0F bit is not cleared, the USB will respond with a NAK handshake to any endpoint 0 OUT tokens; but does not respond to a SETUP token.

Reset clears this bit. Writing to RXD0F has no effect.

1 = Receive on endpoint 0 has occurred
0 = Receive on endpoint 0 has not occurred

ORIE — Receiver Overrun Interrupt Enable Bit

This read/write bit enables SCI error CPU interrupt requests generated by the receiver overrun bit, OR.

1 = SCI error CPU interrupt requests from OR bit enabled
0 = SCI error CPU interrupt requests from OR bit disabled

NEIE — Receiver Noise Error Interrupt Enable Bit

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

1 = SCI error CPU interrupt requests from NE bit enabled
0 = SCI error CPU interrupt requests from NE bit disabled

FEIE — Receiver Framing Error Interrupt Enable Bit

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

1 = SCI error CPU interrupt requests from FE bit enabled
0 = SCI error CPU interrupt requests from FE bit disabled

PEIE — Receiver Parity Error Interrupt Enable Bit

This read/write bit enables SCI receiver CPU interrupt requests generated by the parity error bit, PE. (See **12.9.4 SCI Status Register 1**.) Reset clears PEIE.

1 = SCI error CPU interrupt requests from PE bit enabled
0 = SCI error CPU interrupt requests from PE bit disabled

12.9.5 SCI Status Register 2

SCI status register 2 contains flags to signal the following conditions:

- Break character detected
- Incoming data

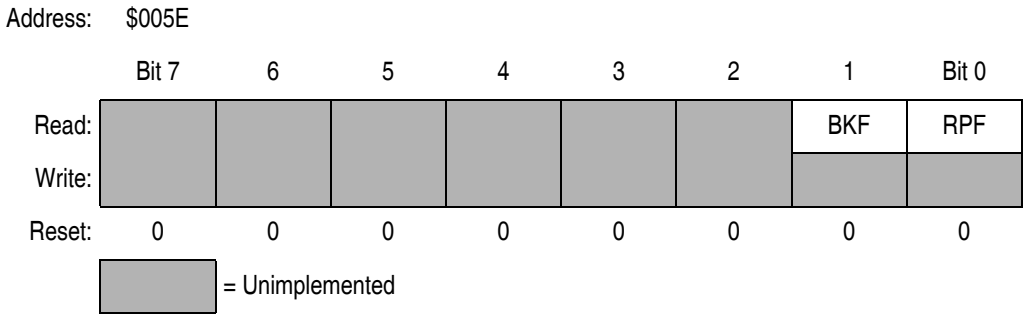


Figure 12-14. SCI Status Register 2 (SCS2)

BKF — Break Flag Bit

This clearable, read-only bit is set when the SCI detects a break character on the RxD pin. In SCS1, the FE and SCRF bits are also set. In 9-bit character transmissions, the R8 bit in SCC3 is cleared. BKF does not generate a CPU interrupt request. Clear BKF by reading SCS2 with BKF set and then reading the SCDR. Once cleared, BKF can become set again only after logic 1s again appear on the RxD pin followed by another break character. Reset clears the BKF bit.

1 = Break character detected
0 = No break character detected

RPF — Reception in Progress Flag Bit

This read-only bit is set when the receiver detects a logic 0 during the RT1 time period of the start bit search. RPF does not generate an interrupt request. RPF is reset after the receiver detects false start bits (usually from noise or a baud rate mismatch) or when the receiver detects an idle character. Polling RPF before disabling the SCI module or entering stop mode can show whether a reception is in progress.

1 = Reception in progress
0 = No reception in progress

Table 12-8. SCI Baud Rate Selection Examples

SCP1 and SCP0	Prescaler Divisor (PD)	SCR2, SCR1, and SCR0	Baud Rate Divisor (BD)	Baud Rate (OSCDCLK=24MHz)
00	1	000	1	Baud rate settings not recommended
00	1	001	2	
00	1	010	4	
00	1	011	8	
00	1	100	16	
00	1	101	32	
00	1	110	64	
00	1	111	128	
01	3	000	1	
01	3	001	2	
01	3	010	4	
01	3	011	8	
01	3	100	16	
01	3	101	32	
01	3	110	64	
01	3	111	128	
10	4	000	1	
10	4	001	2	
10	4	010	4	
10	4	011	8	
10	4	100	16	
10	4	101	32	
10	4	110	64	
10	4	111	128	
11	13	000	1	38461.54
11	13	001	2	19230.77
11	13	010	4	9615.38
11	13	011	8	4807.69
11	13	100	16	2403.85
11	13	101	32	1201.92
11	13	110	64	600.96
11	13	111	128	300.48

13.4.4 External Filter Capacitor Pins (CGMXFC1, CGMXFC2)

The CGMXFC1 and CGMXFC2 pins are required by the loop filter to filter out phase corrections for each PLL. An external filter network is connected to each pin. (See **13.5 CGMXFC External Connections**.)

13.4.5 CGM Clock Output Pins (CGMOUT1, CGMOUT2)

CGMOUT1 and CGMOUT2 are VCO output signals. The output signals are buffered through logic stages to output pins without degrading the loop performance.

13.5 CGMXFC External Connections

The external filter network is critical to the stability and reaction time of the PLL. The configurations shown in **Figure 13-4** (a) and (b) are recommended for connection to CGMXFC1 and CGMXFC2.

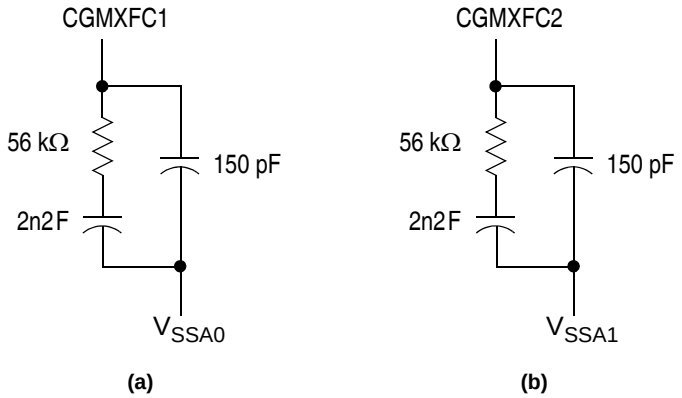


Figure 13-4. CGMXFC External Connections

13.9.5 Reference Divider Select Register Low

The divider select registers low (PRSL1 and PRLS2) contain the programming information for the low byte of reference divider, R.

Address: $0055

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	RDS1_7	RDS1_6	RDS1_5	RDS1_4	RDS1_3	RDS1_2	RDS1_1	RDS1_0
Reset:	1	0	0	1	0	0	0	0

Figure 13-12. PLL1 R Divider Select Register Low (PRSL1)

Address: $0058

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	RDS2_7	RDS2_6	RDS2_5	RDS2_4	RDS2_3	RDS2_2	RDS2_1	RDS2_0
Reset:	1	0	0	1	0	0	0	0

Figure 13-13. PLL2 R Divider Select Register Low (PRSL2)

RDSx_[7:0] — Reference Divider Select Bits

These read/write bits control the high byte of the reference divider, R.

NOTE: *Writing to PRSL also latches the respective high bits, RDSx_[9:8].*

14.4.2 Data Direction Register C

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

Address: $0006

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	0	0	0	0	0	DDRC1	DDRC0
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 14-6. Data Direction Register C (DDRC)

DDRC[1:0] — Data Direction Register C Bits

These read/write bits control port C data direction. Reset clears DDRC[1:0], configuring all port C pins as inputs.

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

NOTE: *Avoid glitches on port C pins by writing to the port C data register before changing data direction register C bits from 0 to 1.*

Figure 14-7 shows the port C I/O logic.

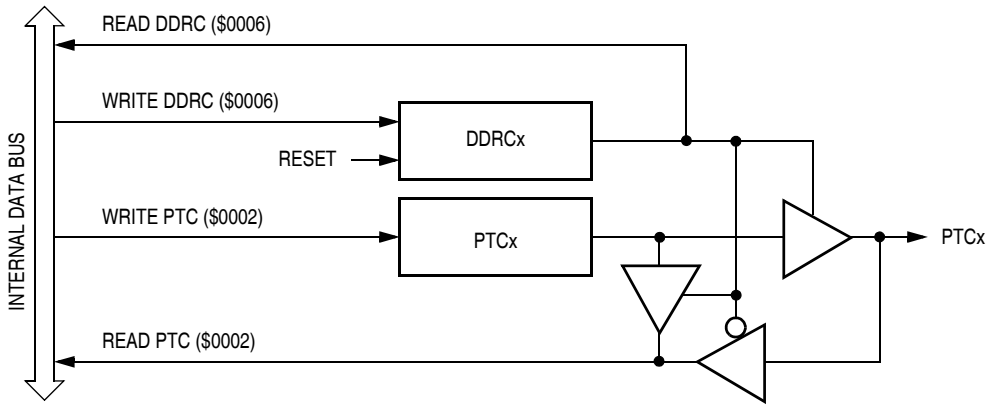


Figure 14-7. Port C I/O Circuit

14.5.2 Data Direction Register D

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

Address: $0007

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	0	DDRD5	DDRD4	DDRD3	DDRD2	DDRD1	DDRD0
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 14-9. Data Direction Register D (DDRD)

DDRD[5:0] — Data Direction Register D Bits

These read/write bits control port D data direction. Reset clears DDRD[5:0], configuring all port D pins as inputs.

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

Port D pins are open-drain when configured as output.

NOTE: *Avoid glitches on port D pins by writing to the port D data register before changing data direction register D bits from 0 to 1.*

NOTE: *For those devices packaged in a 32-pin low-profile quad flat pack, PTD5–1 are not connected. DDRD5–1 should be set to a 1 to configure PTD5–1 as outputs.*

Figure 14-10 shows the port D I/O circuit logic.

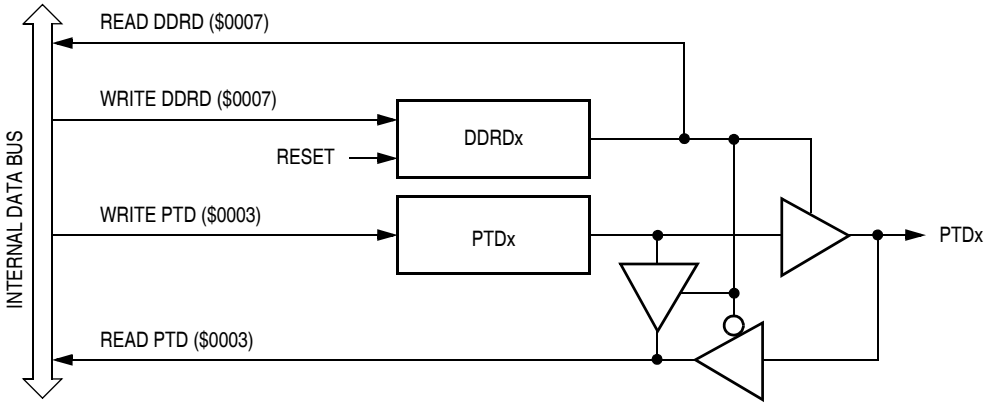


Figure 14-10. Port D I/O Circuit

When bit DDRDx is a logic 1, reading address $0003 reads the PTDx data latch. When bit DDRDx is a logic 0, reading address $0003 reads the voltage level on the pin. The data latch can always be written, regardless of the state of its data direction bit. **Table 14-4** summarizes the operation of the port D pins.

Table 14-4. Port D Pin Functions

DDRD Bit	PTD Bit	I/O Pin Mode	Accesses to DDRD	Accesses to PTD	
			Read/Write	Read	Write
0	X(1)	Input, Hi-Z(2)	DDRD[5:0]	Pin	PTD[5:0](3)
1	X	Output	DDRD[5:0]	PTD[5:0]	PTD[5:0]

Notes:
1. X = don't care.
2. Hi-Z = high impedance.
3. Writing affects data register, but does not affect input.

14.6 Port E

Port E is a 5-bit special function port that shares three of its pins with the timer interface modules (TIMs) and two of its pins with the USB data pins D+ and D–. PTE4 and PTE3 are open-drain when configured as output.

14.6.1 Port E Data Register

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

Address: $0008

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	0	0	PTE4	PTE3	PTE2	PTE1	PTE0
Write:								
Reset:	Unaffected by reset							
Alternative Function:				D–	D+	T2CH01	T1CH01	TCLK
Additional Function:				Optional pullup	Optional pullup	Optional pullup	Optional pullup	Optional pullup
Additional Function:				External interrupt				
				Open-drain	Open-drain			

= Unimplemented

Figure 14-11. Port E Data Register (PTE)

PTE[4:0] — Port E Data Bits

PTE[4:0] are read/write, software-programmable bits. Data direction of each port E pin is under the control of the corresponding bit in data direction register E.

The PTE4 and PTE3 pullup enable bits, PTE4P and PTE3P, in the port option control register (POCR) enable 5kΩ pullups on PTE4 and PTE3 if the respective pin is configured as an input and the USB module is disabled. (See **14.7 Port Options**.)

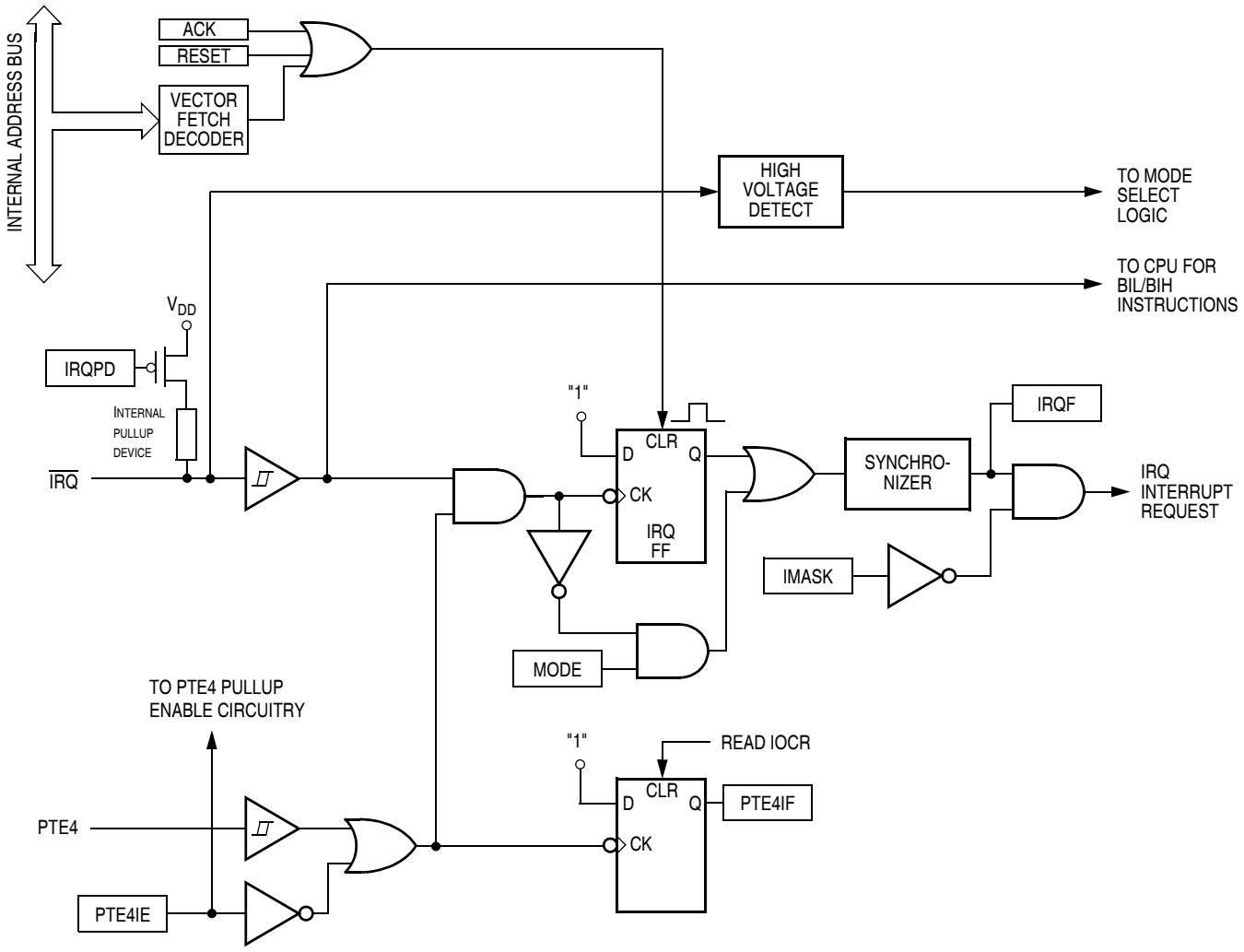


Figure 15-1. IRQ Module Block Diagram

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$001C	IRQ Option Control Register (IOCR)	Read:	0	0	0	0	0	PTE4IF	PTE4IE	IRQPD
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$001E	IRQ Status and Control Register (ISCR)	Read:	0	0	0	0	IRQF	0	IMASK	MODE
		Write:						ACK		
		Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 15-2. IRQ I/O Register Summary

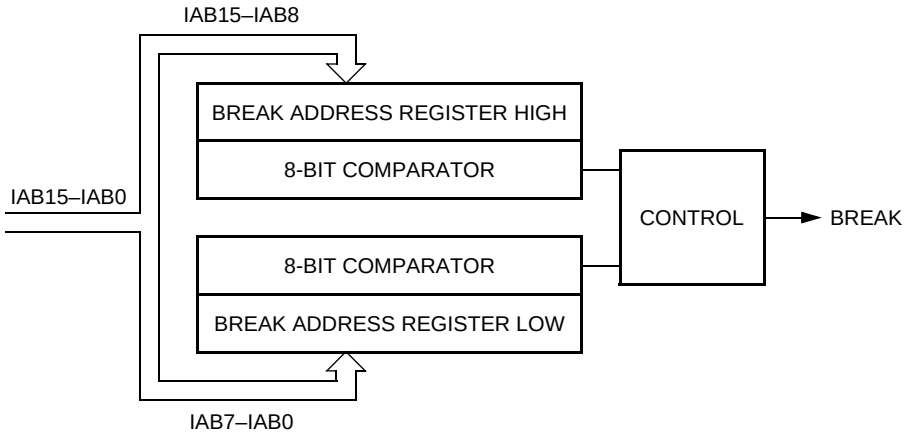


Figure 19-1. Break Module Block Diagram

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$FE00	SIM Break Status Register (SBSR)	Read:	R	R	R	R	R	R	SBSW	R
		Write:							Note	
		Reset:							0	
$FE03	SIM Break Flag Control Register (SBFCR)	Read:	BCFE	R	R	R	R	R	R	R
		Write:								
		Reset:	0							
$FE0C	Break Address Register High (BRKH)	Read:	Bit 15	14	13	12	11	10	9	Bit 8
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$FE0D	Break Address Register Low (BRKL)	Read:	Bit 7	6	5	4	3	2	1	Bit 0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$FE0E	Break Status and Control Register (BRKSCR)	Read:	BRKE	BRKA	0	0	0	0	0	0
		Write:								
		Reset:	0	0	0	0	0	0	0	0

Note: Writing a logic 0 clears SBSW.

= Unimplemented R = Reserved

Figure 19-2. Break Module I/O Register Summary