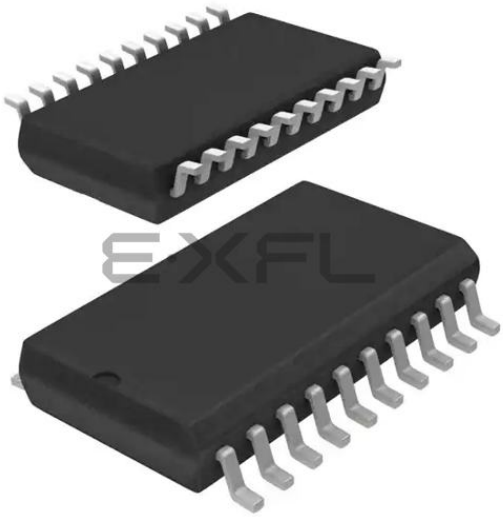


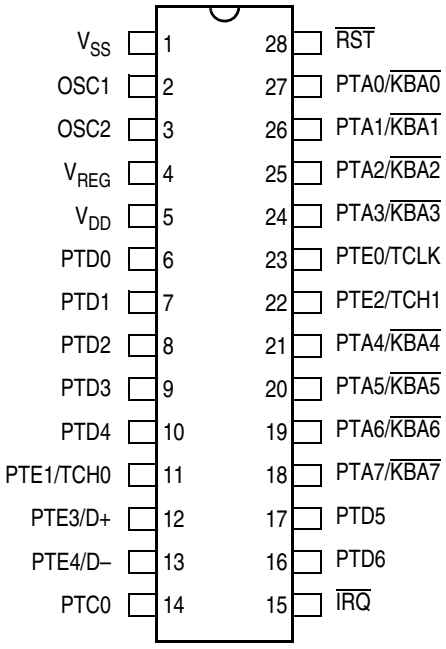
Welcome to E-XFL.COM

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.

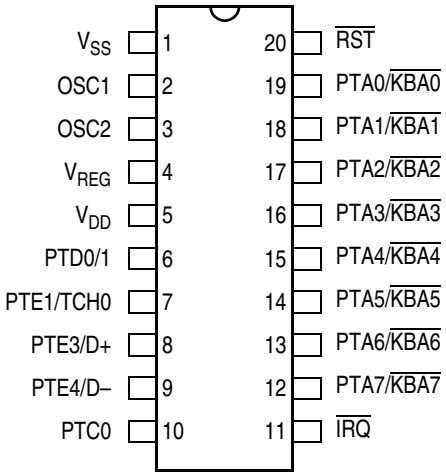
Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	HC08
Core Size	8-Bit
Speed	3MHz
Connectivity	USB
Peripherals	LVD, POR, PWM
Number of I/O	13
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 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	20-SOIC (0.295", 7.50mm Width)
Supplier Device Package	20-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc908jb8jdwe



Pins not available on 28-pin package:		
PTB0		
PTB1	PTC1	
PTB2	PTC2	
PTB3	PTC3	
PTB4	PTC4	
PTB5	PTC5	
PTB6	PTC6	
PTB7	PTC7	PTD7
Internal pads are unconnected.		

Figure 1-3. 28-Pin SOIC Pin Assignments



PTD0/1 pin: PTD0 and PTD1 internal pads are bonded together to PTD0/1 pin.

Pins not available on 20-pin package:			
PTB0			PTE0/TCLK
PTB1	PTC1		
PTB2	PTC2	PTD2	PTE2/TCH1
PTB3	PTC3	PTD3	
PTB4	PTC4	PTD4	
PTB5	PTC5	PTD5	
PTB6	PTC6	PTD6	
PTB7	PTC7	PTD7	
Internal pads are unconnected.			

Figure 1-4. 20-Pin PDIP and SOIC Pin Assignments

NOTE: *In 20-pin package, the PTD0 and PTD1 internal pads are bonded together to PTD0/1 pin.*

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

Figure 4-1. FLASH Memory Register Summary

4.3 Functional Description

The FLASH memory consists of an array of 8,192 bytes for user memory plus a small block of 16 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:

- $DC00–$FBFF (user memory; 8,192 bytes)
- $FFF0–$FFFF (user interrupt vectors; 16 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.

4.6 FLASH Mass Erase Operation

Use the following procedure to erase the entire FLASH memory:

1. Set both the ERASE bit and the MASS bit in the FLASH control register.
2. Write any data to any FLASH address within the address range $FFE0–$FFFF.
3. Wait for a time, t_{nvs} (5 μs).
4. Set the HVEN bit.
5. Wait for a time t_{me} (2 ms).
6. Clear the ERASE bit.
7. Wait for a time, t_{nvh1} (100 μs).
8. Clear the HVEN bit.
9. After time, t_{rcv} (1 μs), the memory can be accessed in read mode again.

NOTE: *Programming and erasing of FLASH locations cannot be performed by code being executed from the FLASH memory. While these operations must be performed in the order as shown, but other unrelated operations may occur between the steps.*

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 $DC	The entire FLASH memory is protected.
$DE (**1101 1110**)	$DE00 (**1101 1110** 0000 0000)
$E0 (**1110 0000**)	$E000 (**1110 0000** 0000 0000)
$E2 (**1110 0010**)	$E200 (**1110 0010** 0000 0000)
$E4 (**1110 0100**)	$E400 (**1110 0100** 0000 0000)
and so on...	
$FE	$FFE0–$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 10. 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	Routine Function
VERIFY	$FC03	FLASH verify routine
ERASE	$FC06	FLASH mass erase routine
PROGRAM	$FC09	FLASH program routine

4.9.1 Variables

The ROM-resident routines use three variables: CTRLBYT, CPUSPD and LADDR; and one data buffer. The minimum size of the data buffer is one byte and the maximum size is 64 bytes.

CPUSPD must be set before calling the ERASE or PROGRAM routine, and should be set to four times the value of the CPU internal bus speed in MHz. For example: for CPU speed of 3MHz, CPUSPD should be set to 12.

Table 4-2. ROM-Resident Routine Variables

Variable	Address	Description
CTRLBYT	$0048	Control byte for setting mass erase.
CPUSPD	$0049	Timing adjustment for different CPU speeds.
LADDR	$004A–$004B	Last FLASH address to be programmed.
DATABUF	$004C–$008B	Data buffer for programming and verifying.

4.9.2 ERASE Routine

The ERASE routine erases the entire FLASH memory. The routine does not check for a blank range before or after erase.

Table 4-3. ERASE Routine

Routine	ERASE
Calling Address	$FC06
Stack Use	5 Bytes
Input	CPUSPD — CPU speed HX — Contains any address in the range to be erased CTRLBYT — Mass erase Mass erase if bit 6 = 1

8.6.2.1 Interrupt Status Register 1

Address: $FE04

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	IF6	IF5	IF4	IF3	IF2	IF1	0	0
Write:	R	R	R	R	R	R	R	R
Reset:	0	0	0	0	0	0	0	0

R = Reserved

Figure 8-12. Interrupt Status Register 1 (INT1)

IF6–IF1 — Interrupt Flags 1–6

These flags indicate the presence of interrupt requests from the sources shown in **Table 8-4**.

1 = Interrupt request present
0 = No interrupt request present

Bit 0 and Bit 1 — Always read 0

8.6.3 Reset

All reset sources always have equal and highest priority and cannot be arbitrated.

8.6.4 Break Interrupts

The break module can stop normal program flow at a software-programmable break point by asserting its break interrupt output. (See **Section 17. Break Module (BREAK)**.) The SIM puts the CPU into the break state by forcing it to the SWI vector location. Refer to the break interrupt subsection of each module to see how each module is affected by the break state.

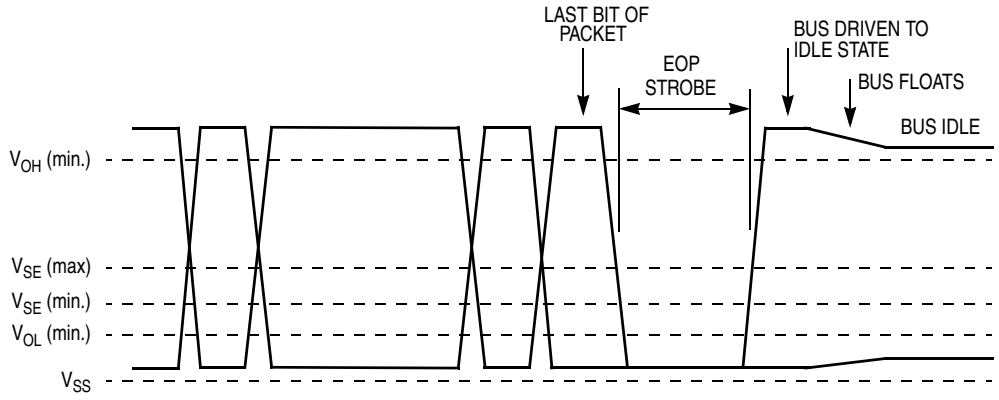


Figure 9-7. EOP Transaction Voltage Levels

The width of the SE0 in the EOP is about two bit times. The EOP width is measured with the same capacitive load used for maximum rise and fall times and is measured at the same level as the differential signal crossover points of the data lines.

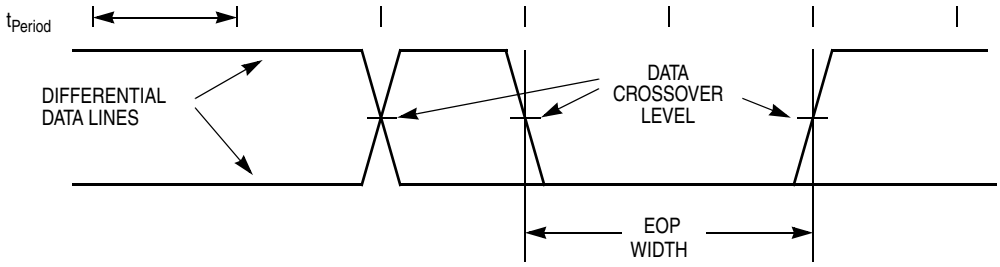


Figure 9-8. EOP Width Timing

9.5.2 Reset Signaling

The USB module will detect a reset signaled on the bus by the presence of an extended SE0 at the USB data pins of a device. The MCU seeing a single-ended 0 on its USB data inputs for more than 8 µs treats that signal as a reset.

A USB sourced reset will hold the MCU in reset for the duration of the reset on the USB bus. The USB bit in the reset status register (RSR) will be set after the internal reset is removed. Refer to **8.8.2 Reset Status Register** for more detail. The MCU's reset recovery sequence is detailed in **Section 8. System Integration Module (SIM)**.

9.5.5 Low-Speed Device

Low-speed devices are configured by the position of a pull-up resistor on the USB D– pin of the MCU. Low-speed devices are terminated as shown in **Figure 9-9** with the pull-up on the D– line.

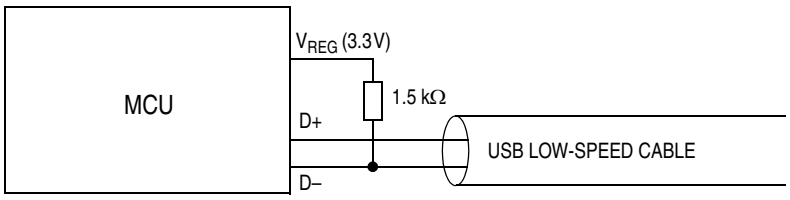


Figure 9-9. External Low-Speed Device Configuration

For low-speed transmissions, the transmitter's EOP width must be between 1.25 µs and 1.50 µs. These ranges include timing variations due to differential buffer delay and rise/fall time mismatches and to noise and other random effects. A low-speed receiver must accept a 670 ns SE0 followed by a J transition as a valid EOP. An SE0 shorter than 330 ns or an SE0 not followed by a J transition are rejected as an EOP. Any SE0 that is 8 µs or longer is automatically a reset.

9.6 Clock Requirements

The low-speed data rate is nominally 1.5 Mbps. The OSCXCLK signal driven by the oscillator circuits is the clock source for the USB module and requires that a 6-MHz oscillator circuit be connected to the OSC1 and OSC2 pins. The permitted frequency tolerance for low-speed functions is approximately ±1.5% (15,000 ppm). This tolerance includes inaccuracies from all sources: initial frequency accuracy, crystal capacitive loading, supply voltage on the oscillator, temperature, and aging. The jitter in the low-speed data rate must be less than 10 ns.

9.7.3 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.

9.8 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)

RXD2IE — Endpoint 2 Receive Interrupt Enable

This read/write bit enables the receive endpoint 2 to generate CPU interrupt requests when the RXD2F bit becomes set. Reset clears the RXD2IE bit.

1 = Receive endpoint 2 can generate a CPU interrupt request
0 = Receive endpoint 2 cannot generate a CPU interrupt request

TXD1IE — Endpoint 1 Transmit Interrupt Enable

This read/write bit enables the transmit endpoint 1 to generate CPU interrupt requests when the TXD1F bit becomes set. Reset clears the TXD1IE bit.

1 = Transmit endpoints 1 can generate a CPU interrupt request
0 = Transmit endpoints 1 cannot generate a CPU interrupt request

TXD0IE — Endpoint 0 Transmit Interrupt Enable

This read/write bit enables the transmit endpoint 0 to generate CPU interrupt requests when the TXD0F bit becomes set. Reset clears the TXD0IE bit.

1 = Transmit endpoint 0 can generate a CPU interrupt request
0 = Transmit endpoint 0 cannot generate a CPU interrupt request

RXD0IE — Endpoint 0 Receive Interrupt Enable

This read/write bit enables the receive endpoint 0 to generate CPU interrupt requests when the RXD0F bit becomes set. Reset clears the RXD0IE bit.

1 = Receive endpoint 0 can generate a CPU interrupt request
0 = Receive endpoint 0 cannot generate a CPU interrupt request

TP0SIZ3–TP0SIZ0 — Endpoint 0 Transmit Data Packet Size

These read/write bits store the number of transmit data bytes for the next IN token request for endpoint 0. These bits are cleared by reset.

9.8.6 USB Control Register 1

Address: $003C

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	T1SEQ	STALL1	TX1E	FRESUM	TP1SIZ3	TP1SIZ2	TP1SIZ1	TP1SIZ0
Reset:	0	0	0	0	0	0	0	0

Figure 9-20. USB Control Register 1 (UCR1)

T1SEQ — Endpoint 1 Transmit Sequence Bit

This read/write bit determines which type of data packet (DATA0 or DATA1) will be sent during the next IN transaction directed to endpoint 1. Toggling of this bit must be controlled by software. Reset clears this bit.

1 = DATA1 token active for next endpoint 1 transmit
0 = DATA0 token active for next endpoint 1 transmit

STALL1 — Endpoint 1 Force Stall Bit

This read/write bit causes endpoint 1 to return a STALL handshake when polled by either an IN or OUT token by the USB host controller. Reset clears this bit.

1 = Send STALL handshake
0 = Default

TX1E — Endpoint 1 Transmit Enable

This read/write bit enables a transmit to occur when the USB host controller sends an IN token to endpoint 1. The appropriate endpoint enable bit, ENABLE1 bit in the UCR3 register, also should be set. Software should set the TX1E bit when data is ready to be transmitted. It must be cleared by software when no more data needs to be transmitted.

10.5 Security

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

NOTE: *Do not leave locations $FFF6–$FFFD blank. For security reasons, program locations $FFF6–$FFFD even if they are not used for vectors.*

During monitor mode entry, the MCU waits after the power-on reset for the host to send the eight security bytes on pin PTA0. If the received bytes match those at locations $FFF6–$FFFD, the host bypasses the security feature and can read all FLASH locations and execute code from FLASH. Security remains bypassed until a power-on reset occurs. If the reset was not a power-on reset, security remains bypassed and security code entry is not required. (See **Figure 10-7**.)

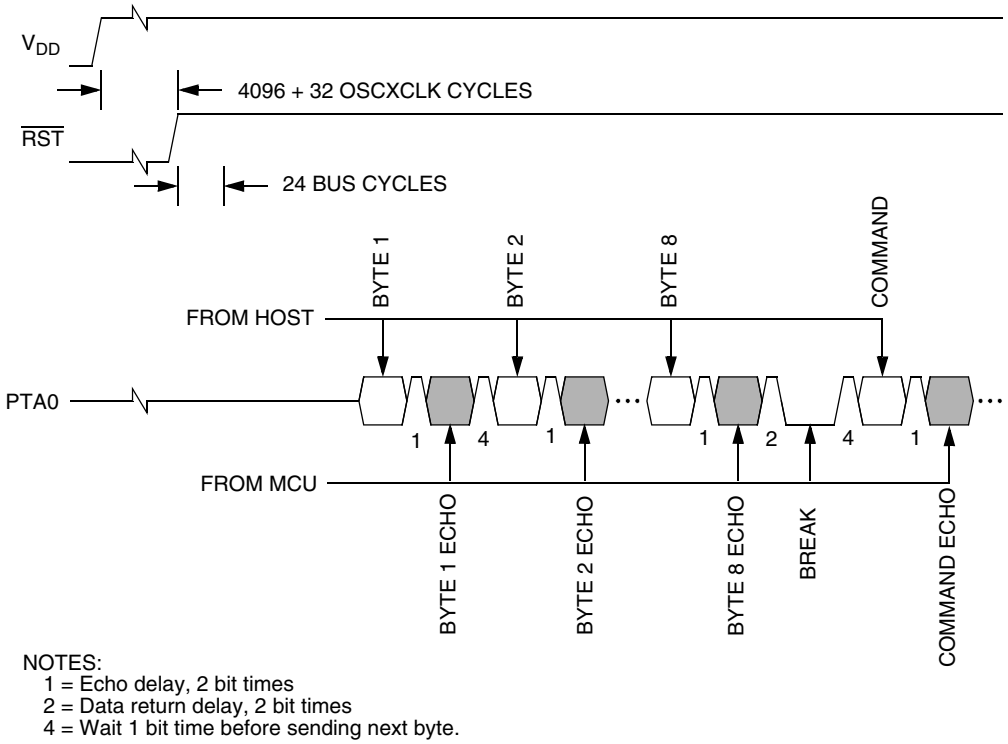


Figure 10-7. Monitor Mode Entry Timing

11.5.3.2 Buffered Output Compare

Channels 0 and 1 can be linked to form a buffered output compare channel whose output appears on the PTE1/TCH0 pin. The TIM channel registers of the linked pair alternately control the output.

Setting the MS0B bit in TIM channel 0 status and control register (TSC0) links channel 0 and channel 1. The output compare value in the TIM channel 0 registers initially controls the output on the PTE1/TCH0 pin. Writing to the TIM channel 1 registers enables the TIM channel 1 registers to synchronously control the output after the TIM overflows. At each subsequent overflow, the TIM channel registers (0 or 1) that control the output are the ones written to last. TSC0 controls and monitors the buffered output compare function, and TIM channel 1 status and control register (TSC1) is unused. While the MS0B bit is set, the channel 1 pin, PTE2/TCH1, is available as a general-purpose I/O pin.

NOTE: *In buffered output compare operation, do not write new output compare values to the currently active channel registers. User software should track the currently active channel to prevent writing a new value to the active channel. Writing to the active channel registers is the same as generating unbuffered output compares.*

11.5.4 Pulse Width Modulation (PWM)

By using the toggle-on-overflow feature with an output compare channel, the TIM can generate a PWM signal. The value in the TIM counter modulo registers determines the period of the PWM signal. The channel pin toggles when the counter reaches the value in the TIM counter modulo registers. The time between overflows is the period of the PWM signal.

As **Figure 11-3** shows, the output compare value in the TIM channel registers determines the pulse width of the PWM signal. The time between overflow and output compare is the pulse width. Program the TIM to clear the channel pin on output compare if the state of the PWM pulse is logic 1. Program the TIM to set the pin if the state of the PWM pulse is logic 0.

11.10.5 TIM Channel Registers

These read/write registers contain the captured TIM counter value of the input capture function or the output compare value of the output compare function. The state of the TIM channel registers after reset is unknown.

In input capture mode (MSxB:MSxA = 0:0), reading the high byte of the TIM channel x registers (TCHxH) inhibits input captures until the low byte (TCHxL) is read.

In output compare mode (MSxB:MSxA ≠ 0:0), writing to the high byte of the TIM channel x registers (TCHxH) inhibits output compares until the low byte (TCHxL) is written.

TCH0H Address: $0011

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

TCH0L Address: $0012

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

TCH1H Address: $0014

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

TCH1L Address: $0015

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

Figure 11-9. TIM Channel Registers (TCH0H/L:TCH1H/L)

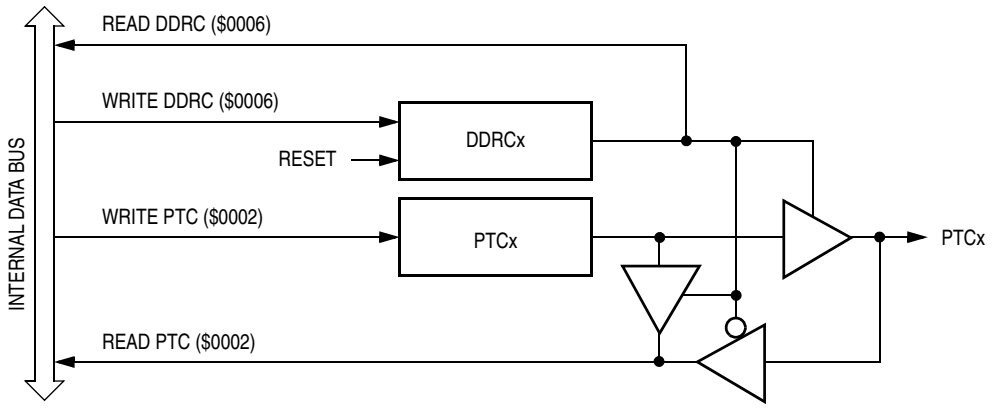


Figure 12-10. Port C I/O Circuit

When bit DDRCx is a logic 1, reading address $0002 reads the PTCx data latch. When bit DDRCx is a logic 0, reading address $0002 reads the voltage level on the pin. The data latch can always be written, regardless of the state of its data direction bit. **Table 12-4** summarizes the operation of the port C pins.

Table 12-4. Port C Pin Functions

DDRC Bit	PTC Bit	I/O Pin Mode	Accesses to DDRC	Accesses to PTC	
			Read/Write	Read	Write
0	X(1)	Input, Hi-Z(2)	DDRC[7:0]	Pin	PTC[7:0](3)
1	X	Output	DDRC[7:0]	PTC[7:0]	PTC[7:0]

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

12.6 Port D

Port D is an 8-bit general-purpose bidirectional I/O port. In 20-pin package, PTD1 and PTD0 internal pads are bonded together to PTD0/1 pin. Port D pins are open-drain when configured as output, and can interface with 5V logic.

Section 14. Keyboard Interrupt Module (KBI)

14.1 Contents

14.2 Introduction

The keyboard interrupt module (KBI) provides eight independently maskable external interrupts which are accessible via PTA0–PTA7 pins.

14.8 Keyboard Module During Break Interrupts

The system integration module (SIM) controls whether the keyboard interrupt latch can be cleared during the break state. The BCFE bit in the break flag control register (BFCR) enables software to clear status bits during the break state.

To allow software to clear the keyboard interrupt latch during a break interrupt, write a logic 1 to the BCFE bit. If a latch is cleared during the break state, it remains cleared when the MCU exits the break state.

To protect the latch during the break state, write a logic 0 to the BCFE bit. With BCFE at logic 0 (its default state), writing to the keyboard acknowledge bit (ACKK) in the keyboard status and control register during the break state has no effect. (See **14.9.1 Keyboard Status and Control Register**.)

14.9 I/O Registers

These registers control and monitor operation of the keyboard module:

- Keyboard status and control register (KBSCR)
- Keyboard interrupt enable register (KBIER)

14.9.1 Keyboard Status and Control Register

The keyboard status and control register:

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

Section 16. Low Voltage Inhibit (LVI)

16.1 Contents

16.2 Introduction

This section describes the low-voltage inhibit module (LVI), which monitors the voltage on the V_{DD} pin and generates a reset when the V_{DD} voltage falls to the LVI trip (V_{LVR}) voltage.

16.3 Functional Description

Figure 16-1 shows the structure of the LVI module. The LVI is enabled after a reset. The LVI module contains a bandgap reference circuit and comparator. Setting LVI disable bit (LVID) disables the LVI to monitor V_{DD} voltage.

The LVI module generates one output signal:

LVI Reset — an reset signal will be generated to reset the CPU when V_{DD} drops to below the set trip point.

18.6 DC Electrical Characteristics

Characteristic[1]	Symbol	Min	Typ[2]	Max	Unit
Regulator output voltage	V_{REG}	3.0	3.3	3.6	V
Output high voltage (I_{Load} = −2.0 mA) PTA0–PTA7, PTB0–PTB7, PTC0–PTC7, PTE0–PTE2	V_{OH}	V_{REG}−0.8	—	—	V
Output low voltage (I_{Load} = 1.6 mA) All I/O pins (I_{Load} = 25 mA) PTD0–PTD1 in ILDD mode (I_{Load} = 10 mA) PTE3–PTE4 with USB disabled	V_{OL}	— — —	— — —	0.4 0.5 0.4	V
Input high voltage All ports, OSC1 $\overline{IRQ}$, $\overline{RST}$	V_{IH}	$0.7 \times V_{REG}$ $0.7 \times V_{DD}$	— —	V_{REG} V_{DD}	V
Input low voltage All ports, OSC1 $\overline{IRQ}$, $\overline{RST}$	V_{IL}	V_{SS} V_{SS}	— —	$0.3 \times V_{REG}$ $0.3 \times V_{DD}$	V
Output low current (V_{OL} = 2.0 V) PTD2–PTD5 in LDD mode	I_{OL}	10	13	20	mA
V_{DD} supply current, V_{DD} = 5.25V, f_{OP} = 3MHz Run, with low speed USB[3] Run, with USB suspended[3] Wait, with low speed USB[4] Wait, with USB suspended[4] Stop[5] 0 °C to 70°C	I_{DD}	 — — — — —	 5.0 4.5 3.0 2.5 300	 7.5 6.5 5.0 4.0 350	 mA mA mA mA µA
I/O ports Hi-Z leakage current	I_{IL}	—	—	± 10	µA
Input current	I_{IN}	—	—	± 1	µA
Capacitance Ports (as input or output)	C_{Out} C_{In}	— —	— —	12 8	pF
POR re-arm voltage[6]	V_{POR}	0	—	100	mV
POR rise-time ramp rate[7]	R_{POR}	0.035	—	—	V/ms
Monitor mode entry voltage	$V_{DD}+V_{HI}$	$1.4 \times V_{DD}$	—	$2 \times V_{DD}$	V
Pullup resistors Port A, port B, port C, PTE0–PTE2, $\overline{RST}$, $\overline{IRQ}$ PTE3–PTE4 (with USB module disabled) D– (with USB module enabled)	R_{PU}	25 4 1.2	40 5 1.5	55 6 2	kΩ
LVI reset	V_{LVR}	2.8	3.3	3.8	V

18.10 USB Low-Speed Source Electrical Characteristics

Characteristic[1]	Symbol	Conditions	Min	Typ	Max	Unit
Internal operating frequency	f_{OP}	—	—	3	—	MHz
Transition time[2] Rise time Fall time	 t_R t_F	 C_L = 200 pF C_L = 600pF C_L = 200 pF C_L = 600pF	 75 75	 — —	 300 300	ns
Rise/Fall time matching	t_{RFM}	t_R/t_F	80	—	120	%
Low speed data rate	t_{DRATE}	1.5 Mbs ± 1.5%	1.4775 676.8	1.500 666.0	1.5225 656.8	Mbs ns
Source differential driver jitter To next transition For paired transitions	 t_{DDJ1} t_{DDJ2}	C_L = 600 pF Measured at crossover point	 –25 –10	 — —	 25 10	ns
Receiver data jitter tolerance To next transition For paired transitions	 t_{DJR1} t_{DJR2}	C_L = 600 pF Measured at crossover point	 –75 –45	 — —	 75 45	ns
Source SEO interval of EOP	t_{LEOPT}	Measured at crossover point	1.25	—	1.50	µs
Source jitter for differential transition to SE0 transition[3]		Measured at crossover point		667		ns
Receiver SEO interval of EOP Must reject as EOP Must accept	 t_{LEOPR1} t_{LEOPR2}	Measured at crossover point	 210 670	 — —	 — —	ns
Width of SEO interval during differential transition	t_{LST}	Measured at crossover point	—	—	210	ns

NOTES:
1. All voltages are measured from local ground, unless otherwise specified. All timings use a capacitive load of 50 pF, unless otherwise specified. Low-speed timings have a 1.5kΩ pullup to 2.8 V on the D– data line.
2. Transition times are measured from 10% to 90% of the data signal. The rising and falling edges should be smoothly transitioning (monotonic). Capacitive loading includes 50 pF of tester capacitance.
3. The two transitions are a (nominal) bit time apart.