



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	MIPS32® M4K™
Core Size	32-Bit Single-Core
Speed	50MHz
Connectivity	I ² C, IrDA, LINbus, PMP, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, I ² S, POR, PWM, WDT
Number of I/O	25
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	2.3V ~ 3.6V
Data Converters	A/D 12x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	36-VFTLA Exposed Pad
Supplier Device Package	36-VTLA (5x5)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic32mx120f032ct-50i-tl

PIC32MX1XX/2XX 28/36/44-PIN FAMILY

3.2 Architecture Overview

The MIPS32 M4K processor core contains several logic blocks working together in parallel, providing an efficient high-performance computing engine. The following blocks are included with the core:

- Execution Unit
- Multiply/Divide Unit (MDU)
- System Control Coprocessor (CP0)
- Fixed Mapping Translation (FMT)
- Dual Internal Bus interfaces
- Power Management
- MIPS16e[®] Support
- Enhanced JTAG (EJTAG) Controller

3.2.1 EXECUTION UNIT

The MIPS32 M4K processor core execution unit implements a load/store architecture with single-cycle ALU operations (logical, shift, add, subtract) and an autonomous multiply/divide unit. The core contains thirty-two 32-bit General Purpose Registers (GPRs) used for integer operations and address calculation. The register file consists of two read ports and one write port and is fully bypassed to minimize operation latency in the pipeline.

The execution unit includes:

- 32-bit adder used for calculating the data address
- Address unit for calculating the next instruction address
- Logic for branch determination and branch target address calculation
- Load aligner
- Bypass multiplexers used to avoid stalls when executing instruction streams where data producing instructions are followed closely by consumers of their results
- Leading Zero/One detect unit for implementing the CLZ and CLO instructions
- Arithmetic Logic Unit (ALU) for performing bitwise logical operations
- Shifter and store aligner

3.2.2 MULTIPLY/DIVIDE UNIT (MDU)

The MIPS32 M4K processor core includes a Multiply/Divide Unit (MDU) that contains a separate pipeline for multiply and divide operations. This pipeline operates in parallel with the Integer Unit (IU) pipeline and does not stall when the IU pipeline stalls. This allows MDU operations to be partially masked by system stalls and/or other integer unit instructions.

The high-performance MDU consists of a 32x16 booth recoded multiplier, result/accumulation registers (HI and LO), a divide state machine, and the necessary multiplexers and control logic. The first number shown ('32' of 32x16) represents the *rs* operand. The second number ('16' of 32x16) represents the *rt* operand. The PIC32 core only checks the value of the latter (*rt*) operand to determine how many times the operation must pass through the multiplier. The 16x16 and 32x16 operations pass through the multiplier once. A 32x32 operation passes through the multiplier twice.

The MDU supports execution of one 16x16 or 32x16 multiply operation every clock cycle; 32x32 multiply operations can be issued every other clock cycle. Appropriate interlocks are implemented to stall the issuance of back-to-back 32x32 multiply operations. The multiply operand size is automatically determined by logic built into the MDU.

Divide operations are implemented with a simple 1 bit per clock iterative algorithm. An early-in detection checks the sign extension of the dividend (*rs*) operand. If *rs* is 8 bits wide, 23 iterations are skipped. For a 16-bit wide *rs*, 15 iterations are skipped and for a 24-bit wide *rs*, 7 iterations are skipped. Any attempt to issue a subsequent MDU instruction while a divide is still active causes an IU pipeline stall until the divide operation is completed.

Table 3-1 lists the repeat rate (peak issue rate of cycles until the operation can be reissued) and latency (number of cycles until a result is available) for the PIC32 core multiply and divide instructions. The approximate latency and repeat rates are listed in terms of pipeline clocks.

TABLE 3-1: MIPS32[®] M4K[®] PROCESSOR CORE HIGH-PERFORMANCE INTEGER MULTIPLY/DIVIDE UNIT LATENCIES AND REPEAT RATES

Opcode	Operand Size (mul <i>rt</i>) (div <i>rs</i>)	Latency	Repeat Rate
MULT/MULTU, MADD/MADDU, MSUB/MSUBU	16 bits	1	1
	32 bits	2	2
MUL	16 bits	2	1
	32 bits	3	2
DIV/DIVU	8 bits	12	11
	16 bits	19	18
	24 bits	26	25
	32 bits	33	32

5.1 Flash Controller Control Registers

TABLE 5-1: FLASH CONTROLLER REGISTER MAP

Virtual Address (BF80_#)	Register Name	Bit Range	Bits																All Resets
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
F400	NVMCON ⁽¹⁾	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	WR	WREN	WRERR	LVDERR	LVDSTAT	—	—	—	—	—	—	—	NVMOP<3:0>				0000
F410	NVMKEY	31:16	NVMKEY<31:0>																0000
		15:0																	0000
F420	NVMADDR ⁽¹⁾	31:16	NVMADDR<31:0>																0000
		15:0																	0000
F430	NVMDATA	31:16	NVMDATA<31:0>																0000
		15:0																	0000
F440	NVMSRCADDR	31:16	NVMSRCADDR<31:0>																0000
		15:0																	0000

Legend: x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

Note 1: This register has corresponding CLR, SET and INV registers at its virtual address, plus offsets of 0x4, 0x8 and 0xC, respectively. See **Section 11.2 “CLR, SET and INV Registers”** for more information.

PIC32MX1XX/2XX 28/36/44-PIN FAMILY

REGISTER 9-5: DCRCDATA: DMA CRC DATA REGISTER

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DCRCDATA<31:24>								
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DCRCDATA<23:16>								
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DCRCDATA<15:8>								
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DCRCDATA<7:0>								

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-0 **DCRCDATA<31:0>**: CRC Data Register bits

Writing to this register will seed the CRC generator. Reading from this register will return the current value of the CRC. Bits greater than PLEN will return '0' on any read.

When CRCTYP (DCRCCON<15>) = 1 (CRC module is in IP Header mode):

Only the lower 16 bits contain IP header checksum information. The upper 16 bits are always '0'. Data written to this register is converted and read back in 1's complement form (i.e., current IP header checksum value).

When CRCTYP (DCRCCON<15>) = 0 (CRC module is in LFSR mode):

Bits greater than PLEN will return '0' on any read.

REGISTER 9-6: DCRCXOR: DMA CRCXOR ENABLE REGISTER

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DCRCXOR<31:24>								
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DCRCXOR<23:16>								
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DCRCXOR<15:8>								
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DCRCXOR<7:0>								

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-0 **DCRCXOR<31:0>**: CRC XOR Register bits

When CRCTYP (DCRCCON<15>) = 1 (CRC module is in IP Header mode):

This register is unused.

When CRCTYP (DCRCCON<15>) = 0 (CRC module is in LFSR mode):

1 = Enable the XOR input to the Shift register

0 = Disable the XOR input to the Shift register; data is shifted in directly from the previous stage in the register

PIC32MX1XX/2XX 28/36/44-PIN FAMILY

REGISTER 9-9: DCHxINT: DMA CHANNEL 'x' INTERRUPT CONTROL REGISTER

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	R/W-0 CHSDIE	R/W-0 CHSHIE	R/W-0 CHDDIE	R/W-0 CHDHIE	R/W-0 CHBCIE	R/W-0 CHCCIE	R/W-0 CHTAIE	R/W-0 CHERIE
15:8	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
7:0	R/W-0 CHSDIF	R/W-0 CHSHIF	R/W-0 CHDDIF	R/W-0 CHDHIF	R/W-0 CHBCIF	R/W-0 CHCCIF	R/W-0 CHTAIF	R/W-0 CHERIF

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-24 **Unimplemented:** Read as '0'

bit 23 **CHSDIE:** Channel Source Done Interrupt Enable bit

1 = Interrupt is enabled

0 = Interrupt is disabled

bit 22 **CHSHIE:** Channel Source Half Empty Interrupt Enable bit

1 = Interrupt is enabled

0 = Interrupt is disabled

bit 21 **CHDDIE:** Channel Destination Done Interrupt Enable bit

1 = Interrupt is enabled

0 = Interrupt is disabled

bit 20 **CHDHIE:** Channel Destination Half Full Interrupt Enable bit

1 = Interrupt is enabled

0 = Interrupt is disabled

bit 19 **CHBCIE:** Channel Block Transfer Complete Interrupt Enable bit

1 = Interrupt is enabled

0 = Interrupt is disabled

bit 18 **CHCCIE:** Channel Cell Transfer Complete Interrupt Enable bit

1 = Interrupt is enabled

0 = Interrupt is disabled

bit 17 **CHTAIE:** Channel Transfer Abort Interrupt Enable bit

1 = Interrupt is enabled

0 = Interrupt is disabled

bit 16 **CHERIE:** Channel Address Error Interrupt Enable bit

1 = Interrupt is enabled

0 = Interrupt is disabled

bit 15-8 **Unimplemented:** Read as '0'

bit 7 **CHSDIF:** Channel Source Done Interrupt Flag bit

1 = Channel Source Pointer has reached end of source (CHSPTR = CHSSIZ)

0 = No interrupt is pending

bit 6 **CHSHIF:** Channel Source Half Empty Interrupt Flag bit

1 = Channel Source Pointer has reached midpoint of source (CHSPTR = CHSSIZ/2)

0 = No interrupt is pending

bit 5 **CHDDIF:** Channel Destination Done Interrupt Flag bit

1 = Channel Destination Pointer has reached end of destination (CHDPTR = CHDSIZ)

0 = No interrupt is pending

PIC32MX1XX/2XX 28/36/44-PIN FAMILY

REGISTER 10-3: U1OTGSTAT: USB OTG STATUS REGISTER

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	R-0	U-0	R-0	U-0	R-0	R-0	U-0	R-0
	ID	—	LSTATE	—	SESVD	SESEND	—	VBUSVD

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-8 **Unimplemented:** Read as '0'

bit 7 **ID:** ID Pin State Indicator bit

1 = No cable is attached or a "type B" cable has been inserted into the USB receptacle

0 = A "type A" OTG cable has been inserted into the USB receptacle

bit 6 **Unimplemented:** Read as '0'

bit 5 **LSTATE:** Line State Stable Indicator bit

1 = USB line state (SE0 (U1CON<6>) bit and JSTATE (U1CON<7>)) bit has been stable for previous 1 ms

0 = USB line state (SE0 and JSTATE) has not been stable for previous 1 ms

bit 4 **Unimplemented:** Read as '0'

bit 3 **SESVD:** Session Valid Indicator bit

1 = VBUS voltage is above Session Valid on the A or B device

0 = VBUS voltage is below Session Valid on the A or B device

bit 2 **SESEND:** B-Device Session End Indicator bit

1 = VBUS voltage is below Session Valid on the B device

0 = VBUS voltage is above Session Valid on the B device

bit 1 **Unimplemented:** Read as '0'

bit 0 **VBUSVD:** A-Device VBUS Valid Indicator bit

1 = VBUS voltage is above Session Valid on the A device

0 = VBUS voltage is below Session Valid on the A device

PIC32MX1XX/2XX 28/36/44-PIN FAMILY

REGISTER 10-7: U1IE: USB INTERRUPT ENABLE REGISTER

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	STALLIE	ATTACHIE	RESUMEIE	IDLEIE	TRNIE	SOFIE	UERRIE ⁽¹⁾	URSTIE ⁽²⁾ DETACHIE ⁽³⁾

Legend:

R = Readable bit W = Writable bit U = Unimplemented bit, read as '0'
-n = Value at POR '1' = Bit is set '0' = Bit is cleared x = Bit is unknown

bit 31-8 **Unimplemented:** Read as '0'

bit 7 **STALLIE:** STALL Handshake Interrupt Enable bit

1 = STALL interrupt is enabled
0 = STALL interrupt is disabled

bit 6 **ATTACHIE:** ATTACH Interrupt Enable bit

1 = ATTACH interrupt is enabled
0 = ATTACH interrupt is disabled

bit 5 **RESUMEIE:** RESUME Interrupt Enable bit

1 = RESUME interrupt is enabled
0 = RESUME interrupt is disabled

bit 4 **IDLEIE:** Idle Detect Interrupt Enable bit

1 = Idle interrupt is enabled
0 = Idle interrupt is disabled

bit 3 **TRNIE:** Token Processing Complete Interrupt Enable bit

1 = TRNIF interrupt is enabled
0 = TRNIF interrupt is disabled

bit 2 **SOFIE:** SOF Token Interrupt Enable bit

1 = SOFIF interrupt is enabled
0 = SOFIF interrupt is disabled

bit 1 **UERRIE:** USB Error Interrupt Enable bit⁽¹⁾

1 = USB Error interrupt is enabled
0 = USB Error interrupt is disabled

bit 0 **URSTIE:** USB Reset Interrupt Enable bit⁽²⁾

1 = URSTIF interrupt is enabled
0 = URSTIF interrupt is disabled

DETACHIE: USB Detach Interrupt Enable bit⁽³⁾

1 = DATTCHIF interrupt is enabled
0 = DATTCHIF interrupt is disabled

Note 1: For an interrupt to propagate USBIF, the UERRIE (U1IE<1>) bit must be set.

2: Device mode.

3: Host mode.

PIC32MX1XX/2XX 28/36/44-PIN FAMILY

REGISTER 10-14: U1FRMH: USB FRAME NUMBER HIGH REGISTER

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	U-0	R-0	R-0	R-0
	—	—	—	—	—	FRMH<2:0>		

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-3 **Unimplemented:** Read as '0'

bit 2-0 **FRMH<2:0>:** The Upper 3 bits of the Frame Numbers bits

The register bits are updated with the current frame number whenever a SOF TOKEN is received.

REGISTER 10-15: U1TOK: USB TOKEN REGISTER

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PID<3:0> ⁽¹⁾				EP<3:0>			

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-8 **Unimplemented:** Read as '0'

bit 7-4 **PID<3:0>:** Token Type Indicator bits⁽¹⁾

1101 = SETUP (TX) token type transaction

1001 = IN (RX) token type transaction

0001 = OUT (TX) token type transaction

Note: All other values are reserved and must not be used.

bit 3-0 **EP<3:0>:** Token Command Endpoint Address bits

The four bit value must specify a valid endpoint.

Note 1: All other values are reserved and must not be used.

11.3 Peripheral Pin Select

A major challenge in general purpose devices is providing the largest possible set of peripheral features while minimizing the conflict of features on I/O pins. The challenge is even greater on low pin-count devices. In an application where more than one peripheral needs to be assigned to a single pin, inconvenient workarounds in application code or a complete redesign may be the only option.

The Peripheral Pin Select (PPS) configuration provides an alternative to these choices by enabling peripheral set selection and their placement on a wide range of I/O pins. By increasing the pinout options available on a particular device, users can better tailor the device to their entire application, rather than trimming the application to fit the device.

The PPS configuration feature operates over a fixed subset of digital I/O pins. Users may independently map the input and/or output of most digital peripherals to these I/O pins. PPS is performed in software and generally does not require the device to be reprogrammed. Hardware safeguards are included that prevent accidental or spurious changes to the peripheral mapping once it has been established.

11.3.1 AVAILABLE PINS

The number of available pins is dependent on the particular device and its pin count. Pins that support the PPS feature include the designation “RPn” in their full pin designation, where “RP” designates a remappable peripheral and “n” is the remappable port number.

11.3.2 AVAILABLE PERIPHERALS

The peripherals managed by the PPS are all digital-only peripherals. These include general serial communications (UART and SPI), general purpose timer clock inputs, timer-related peripherals (input capture and output compare) and interrupt-on-change inputs.

In comparison, some digital-only peripheral modules are never included in the PPS feature. This is because the peripheral's function requires special I/O circuitry on a specific port and cannot be easily connected to multiple pins. These modules include I²C among others. A similar requirement excludes all modules with analog inputs, such as the Analog-to-Digital Converter (ADC).

A key difference between remappable and non-remappable peripherals is that remappable peripherals are not associated with a default I/O pin. The peripheral must always be assigned to a specific I/O pin before it can be used. In contrast, non-remappable peripherals are always available on a default pin, assuming that the peripheral is active and not conflicting with another peripheral.

When a remappable peripheral is active on a given I/O pin, it takes priority over all other digital I/O and digital communication peripherals associated with the pin.

Priority is given regardless of the type of peripheral that is mapped. Remappable peripherals never take priority over any analog functions associated with the pin.

11.3.3 CONTROLLING PERIPHERAL PIN SELECT

PPS features are controlled through two sets of SFRs: one to map peripheral inputs, and one to map outputs. Because they are separately controlled, a particular peripheral's input and output (if the peripheral has both) can be placed on any selectable function pin without constraint.

The association of a peripheral to a peripheral-selectable pin is handled in two different ways, depending on whether an input or output is being mapped.

11.3.4 INPUT MAPPING

The inputs of the PPS options are mapped on the basis of the peripheral. That is, a control register associated with a peripheral dictates the pin it will be mapped to. The [pin name]R registers, where [pin name] refers to the peripheral pins listed in Table 11-1, are used to configure peripheral input mapping (see Register 11-1). Each register contains sets of 4 bit fields. Programming these bit fields with an appropriate value maps the RPn pin with the corresponding value to that peripheral. For any given device, the valid range of values for any bit field is shown in Table 11-1.

For example, Figure 11-2 illustrates the remappable pin selection for the U1RX input.

FIGURE 11-2: REMAPPABLE INPUT EXAMPLE FOR U1RX

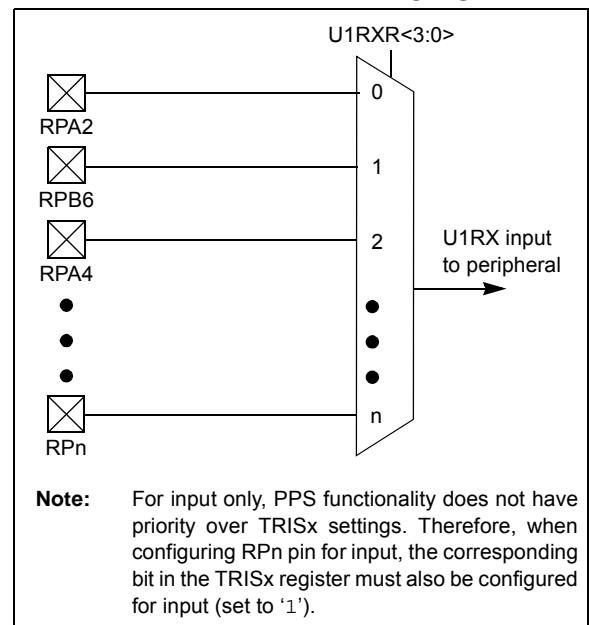


TABLE 11-7: PERIPHERAL PIN SELECT OUTPUT REGISTER MAP (CONTINUED)

Virtual Address (BF80_#)	Register Name	Bit Range	Bits																All Resets
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
FB4C	RPB8R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPB8<3:0>				0000
FB50	RPB9R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPB9<3:0>				0000
FB54	RPB10R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPB10<3:0>				0000
FB58	RPB11R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPB11<3:0>				0000
FB60	RPB13R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPB13<3:0>				0000
FB64	RPB14R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPB14<3:0>				0000
FB68	RPB15R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPB15<3:0>				0000
FB6C	RPC0R ⁽³⁾	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPC0<3:0>				0000
FB70	RPC1R ⁽³⁾	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPC1<3:0>				0000
FB74	RPC2R ⁽¹⁾	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPC2<3:0>				0000
FB78	RPC3R ⁽³⁾	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPC3<3:0>				0000
FB7C	RPC4R ⁽¹⁾	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPC4<3:0>				0000
FB80	RPC5R ⁽¹⁾	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPC5<3:0>				0000
FB84	RPC6R ⁽¹⁾	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPC6<3:0>				0000
FB88	RPC7R ⁽¹⁾	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPC7<3:0>				0000

Legend: x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

- Note**
- 1: This register is only available on 44-pin devices.
 - 2: This register is only available on PIC32MX1XX devices.
 - 3: This register is only available on 36-pin and 44-pin devices.

PIC32MX1XX/2XX 28/36/44-PIN FAMILY

REGISTER 11-1: [pin name]R: PERIPHERAL PIN SELECT INPUT REGISTER

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	[pin name]R<3:0>			

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-4 **Unimplemented:** Read as '0'

bit 3-0 **[pin name]R<3:0>**: Peripheral Pin Select Input bits

Where [pin name] refers to the pins that are used to configure peripheral input mapping. See Table 11-1 for input pin selection values.

Note: Register values can only be changed if the Configuration bit, IOLOCK (CFGCON<13>), = 0.

REGISTER 11-2: RPnR: PERIPHERAL PIN SELECT OUTPUT REGISTER

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	RPnR<3:0>			

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-4 **Unimplemented:** Read as '0'

bit 3-0 **RPnR<3:0>**: Peripheral Pin Select Output bits

See Table 11-2 for output pin selection values.

Note: Register values can only be changed if the Configuration bit, IOLOCK (CFGCON<13>), = 0.

15.1 Input Capture Control Registers

TABLE 15-1: INPUT CAPTURE 1-INPUT CAPTURE 5 REGISTER MAP

Virtual Address (BF80..#)	Register Name	Bit Range	Bits															All Resets	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1		16/0
2000	IC1CON ⁽¹⁾	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	FEDGE	C32	ICTMR	ICI<1:0>		ICOV	ICBNE	ICM<2:0>			0000
2010	IC1BUF	31:16	IC1BUF<31:0>															xxxx	
		15:0	IC1BUF<31:0>															xxxx	
2200	IC2CON ⁽¹⁾	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	FEDGE	C32	ICTMR	ICI<1:0>		ICOV	ICBNE	ICM<2:0>			0000
2210	IC2BUF	31:16	IC2BUF<31:0>															xxxx	
		15:0	IC2BUF<31:0>															xxxx	
2400	IC3CON ⁽¹⁾	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	FEDGE	C32	ICTMR	ICI<1:0>		ICOV	ICBNE	ICM<2:0>			0000
2410	IC3BUF	31:16	IC3BUF<31:0>															xxxx	
		15:0	IC3BUF<31:0>															xxxx	
2600	IC4CON ⁽¹⁾	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	FEDGE	C32	ICTMR	ICI<1:0>		ICOV	ICBNE	ICM<2:0>			0000
2610	IC4BUF	31:16	IC4BUF<31:0>															xxxx	
		15:0	IC4BUF<31:0>															xxxx	
2800	IC5CON ⁽¹⁾	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	FEDGE	C32	ICTMR	ICI<1:0>		ICOV	ICBNE	ICM<2:0>			0000
2810	IC5BUF	31:16	IC5BUF<31:0>															xxxx	
		15:0	IC5BUF<31:0>															xxxx	

Legend: x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

Note 1: This register has corresponding CLR, SET and INV registers at its virtual address, plus an offset of 0x4, 0x8 and 0xC, respectively. See **Section 11.2 “CLR, SET and INV Registers”** for more information.

PIC32MX1XX/2XX 28/36/44-PIN FAMILY

REGISTER 18-2: I2CxSTAT: I²C STATUS REGISTER

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	R-0, HSC ACKSTAT	R-0, HSC TRSTAT	U-0 —	U-0 —	U-0 —	R/C-0, HS BCL	R-0, HSC GCSTAT	R-0, HSC ADD10
7:0	R/C-0, HS IWCOL	R/C-0, HS I2COV	R-0, HSC D_A	R/C-0, HSC P	R/C-0, HSC S	R-0, HSC R_W	R-0, HSC RBF	R-0, HSC TBF

Legend:	HS = Set in hardware	HSC = Hardware set/cleared
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared
		C = Clearable bit

bit 31-16 **Unimplemented:** Read as '0'

bit 15 **ACKSTAT:** Acknowledge Status bit (when operating as I²C master, applicable to master transmit operation)
 1 = Acknowledge was not received from slave
 0 = Acknowledge was received from slave
 Hardware set or clear at end of slave Acknowledge.

bit 14 **TRSTAT:** Transmit Status bit (when operating as I²C master, applicable to master transmit operation)
 1 = Master transmit is in progress (8 bits + ACK)
 0 = Master transmit is not in progress
 Hardware set at beginning of master transmission. Hardware clear at end of slave Acknowledge.

bit 13-11 **Unimplemented:** Read as '0'

bit 10 **BCL:** Master Bus Collision Detect bit
 1 = A bus collision has been detected during a master operation
 0 = No collision
 Hardware set at detection of bus collision. This condition can only be cleared by disabling (ON bit = 0) and re-enabling (ON bit = 1) the module.

bit 9 **GCSTAT:** General Call Status bit
 1 = General call address was received
 0 = General call address was not received
 Hardware set when address matches general call address. Hardware clear at Stop detection.

bit 8 **ADD10:** 10-bit Address Status bit
 1 = 10-bit address was matched
 0 = 10-bit address was not matched
 Hardware set at match of 2nd byte of matched 10-bit address. Hardware clear at Stop detection.

bit 7 **IWCOL:** Write Collision Detect bit
 1 = An attempt to write the I2CxTRN register failed because the I²C module is busy
 0 = No collision
 Hardware set at occurrence of write to I2CxTRN while busy (cleared by software).

bit 6 **I2COV:** Receive Overflow Flag bit
 1 = A byte was received while the I2CxRCV register is still holding the previous byte
 0 = No overflow
 Hardware set at attempt to transfer I2CxRSR to I2CxRCV (cleared by software).

bit 5 **D_A:** Data/Address bit (when operating as I²C slave)
 1 = Indicates that the last byte received was data
 0 = Indicates that the last byte received was device address
 Hardware clear at device address match. Hardware set by reception of slave byte.

PIC32MX1XX/2XX 28/36/44-PIN FAMILY

REGISTER 20-2: PMMODE: PARALLEL PORT MODE REGISTER

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
	BUSY	IRQM<1:0>		INCM<1:0>		—	MODE<1:0>	
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	WAITB<1:0> ⁽¹⁾		WAITM<3:0> ⁽¹⁾			WAITE<1:0> ⁽¹⁾		

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-16 **Unimplemented:** Read as '0'

bit 15 **BUSY:** Busy bit (Master mode only)

1 = Port is busy

0 = Port is not busy

bit 14-13 **IRQM<1:0>:** Interrupt Request Mode bits

11 = Reserved, do not use

10 = Interrupt generated when Read Buffer 3 is read or Write Buffer 3 is written (Buffered PSP mode) or on a read or write operation when PMA<1:0> = 11 (Addressable Slave mode only)

01 = Interrupt generated at the end of the read/write cycle

00 = No Interrupt generated

bit 12-11 **INCM<1:0>:** Increment Mode bits

11 = Slave mode read and write buffers auto-increment (MODE<1:0> = 00 only)

10 = Decrement ADDR<10:2> and ADDR<14> by 1 every read/write cycle⁽²⁾

01 = Increment ADDR<10:2> and ADDR<14> by 1 every read/write cycle⁽²⁾

00 = No increment or decrement of address

bit 10 **Unimplemented:** Read as '0'

bit 9-8 **MODE<1:0>:** Parallel Port Mode Select bits

11 = Master mode 1 (PMCS1, PMRD/PMWR, PMENB, PMA<x:0>, and PMD<7:0>)

10 = Master mode 2 (PMCS1, PMRD, PMWR, PMA<x:0>, and PMD<7:0>)

01 = Enhanced Slave mode, control signals (PMRD, PMWR, PMCS1, PMD<7:0>, and PMA<1:0>)

00 = Legacy Parallel Slave Port, control signals (PMRD, PMWR, PMCS1, and PMD<7:0>)

bit 7-6 **WAITB<1:0>:** Data Setup to Read/Write Strobe Wait States bits⁽¹⁾

11 = Data wait of 4 TPB; multiplexed address phase of 4 TPB

10 = Data wait of 3 TPB; multiplexed address phase of 3 TPB

01 = Data wait of 2 TPB; multiplexed address phase of 2 TPB

00 = Data wait of 1 TPB; multiplexed address phase of 1 TPB (default)

bit 5-2 **WAITM<3:0>:** Data Read/Write Strobe Wait States bits⁽¹⁾

1111 = Wait of 16 TPB

•
•
•

0001 = Wait of 2 TPB

0000 = Wait of 1 TPB (default)

Note 1: Whenever WAITM<3:0> = 0000, WAITB and WAITE bits are ignored and forced to 1 TPBCLK cycle for a write operation; WAITB = 1 TPBCLK cycle, WAITE = 0 TPBCLK cycles for a read operation.

2: Address bit A14 is not subject to auto-increment/decrement if configured as Chip Select CS1.

PIC32MX1XX/2XX 28/36/44-PIN FAMILY

REGISTER 21-1: RTCCON: RTC CONTROL REGISTER (CONTINUED)

- bit 5-4 **Unimplemented:** Read as '0'
- bit 3 **RTCWREN:** RTC Value Registers Write Enable bit⁽⁴⁾
1 = RTC Value registers can be written to by the user
0 = RTC Value registers are locked out from being written to by the user
- bit 2 **RTCSYNC:** RTCC Value Registers Read Synchronization bit
1 = RTC Value registers can change while reading, due to a rollover ripple that results in an invalid data read
 If the register is read twice and results in the same data, the data can be assumed to be valid
0 = RTC Value registers can be read without concern about a rollover ripple
- bit 1 **HALFSEC:** Half-Second Status bit⁽⁵⁾
1 = Second half period of a second
0 = First half period of a second
- bit 0 **RTCOE:** RTCC Output Enable bit
1 = RTCC clock output enabled – clock presented onto an I/O
0 = RTCC clock output disabled

- Note 1:** The ON bit is only writable when RTCWREN = 1.
- 2:** When using the 1:1 PBCLK divisor, the user's software should not read/write the peripheral's SFRs in the SYSCLK cycle immediately following the instruction that clears the module's ON bit.
- 3:** Requires RTCOE = 1 (RTCCON<0>) for the output to be active.
- 4:** The RTCWREN bit can be set only when the write sequence is enabled.
- 5:** This bit is read-only. It is cleared to '0' on a write to the seconds bit fields (RTCTIME<14:8>).

Note: This register is reset only on a Power-on Reset (POR).

PIC32MX1XX/2XX 28/36/44-PIN FAMILY

REGISTER 21-4: RTCDATE: RTC DATE VALUE REGISTER

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
	YEAR10<3:0>				YEAR01<3:0>			
23:16	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
	—	—	—	MONTH10	MONTH01<3:0>			
15:8	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
	—	—	DAY10<1:0>		DAY01<3:0>			
7:0	U-0	U-0	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x
	—	—	—	—	—	WDAY01<2:0>		

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 31-28 **YEAR10<3:0>**: Binary-Coded Decimal Value of Years bits, 10s place digit; contains a value from 0 to 9

bit 27-24 **YEAR01<3:0>**: Binary-Coded Decimal Value of Years bits, 1s place digit; contains a value from 0 to 9

bit 23-21 **Unimplemented**: Read as '0'

bit 20 **MONTH10**: Binary-Coded Decimal Value of Months bits, 10s place digit; contains a value of 0 or 1

bit 19-16 **MONTH01<3:0>**: Binary-Coded Decimal Value of Months bits, 1s place digit; contains a value from 0 to 9

bit 15-14 **Unimplemented**: Read as '0'

bit 13-12 **DAY10<1:0>**: Binary-Coded Decimal Value of Days bits, 10s place digit; contains a value of 0 to 3

bit 11-8 **DAY01<3:0>**: Binary-Coded Decimal Value of Days bits, 1s place digit; contains a value from 0 to 9

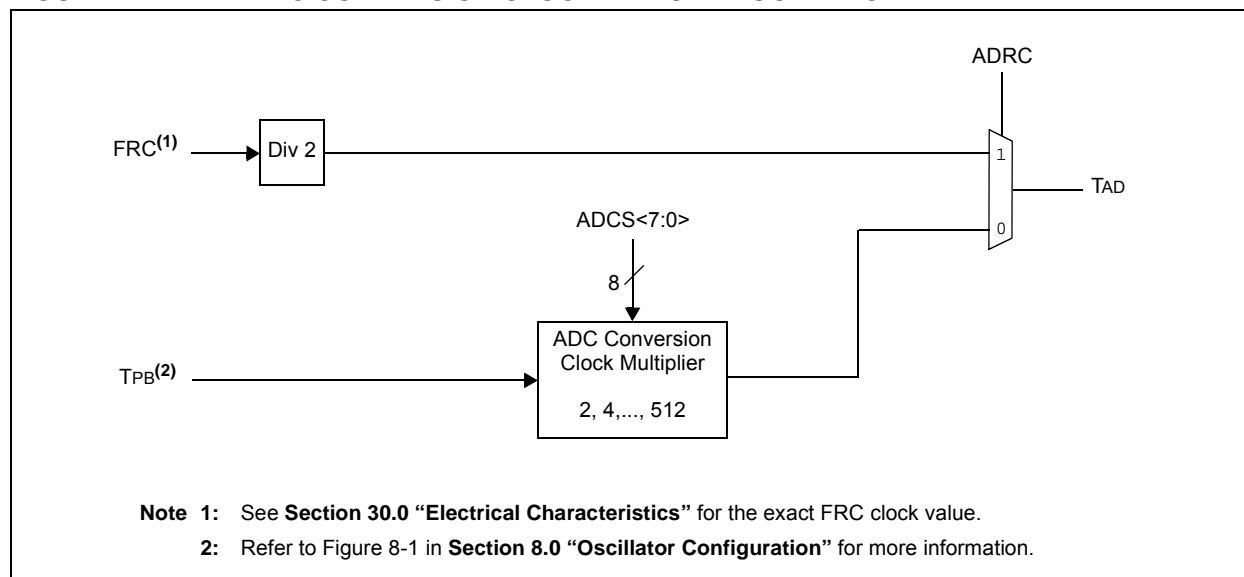
bit 7-3 **Unimplemented**: Read as '0'

bit 2-0 **WDAY01<2:0>**: Binary-Coded Decimal Value of Weekdays bits; contains a value from 0 to 6

Note: This register is only writable when RTCWREN = 1 (RTCCON<3>).

PIC32MX1XX/2XX 28/36/44-PIN FAMILY

FIGURE 22-2: ADC CONVERSION CLOCK PERIOD BLOCK DIAGRAM



29.0 DEVELOPMENT SUPPORT

The PIC® microcontrollers (MCU) and dsPIC® digital signal controllers (DSC) are supported with a full range of software and hardware development tools:

- Integrated Development Environment
 - MPLAB® X IDE Software
- Compilers/Assemblers/Linkers
 - MPLAB XC Compiler
 - MPASM™ Assembler
 - MPLINK™ Object Linker/
MPLIB™ Object Librarian
 - MPLAB Assembler/Linker/Librarian for
Various Device Families
- Simulators
 - MPLAB X SIM Software Simulator
- Emulators
 - MPLAB REAL ICE™ In-Circuit Emulator
- In-Circuit Debuggers/Programmers
 - MPLAB ICD 3
 - PICKit™ 3
- Device Programmers
 - MPLAB PM3 Device Programmer
- Low-Cost Demonstration/Development Boards,
Evaluation Kits and Starter Kits
- Third-party development tools

29.1 MPLAB X Integrated Development Environment Software

The MPLAB X IDE is a single, unified graphical user interface for Microchip and third-party software, and hardware development tool that runs on Windows®, Linux and Mac OS® X. Based on the NetBeans IDE, MPLAB X IDE is an entirely new IDE with a host of free software components and plug-ins for high-performance application development and debugging. Moving between tools and upgrading from software simulators to hardware debugging and programming tools is simple with the seamless user interface.

With complete project management, visual call graphs, a configurable watch window and a feature-rich editor that includes code completion and context menus, MPLAB X IDE is flexible and friendly enough for new users. With the ability to support multiple tools on multiple projects with simultaneous debugging, MPLAB X IDE is also suitable for the needs of experienced users.

Feature-Rich Editor:

- Color syntax highlighting
- Smart code completion makes suggestions and provides hints as you type
- Automatic code formatting based on user-defined rules
- Live parsing

User-Friendly, Customizable Interface:

- Fully customizable interface: toolbars, toolbar buttons, windows, window placement, etc.
- Call graph window

Project-Based Workspaces:

- Multiple projects
- Multiple tools
- Multiple configurations
- Simultaneous debugging sessions

File History and Bug Tracking:

- Local file history feature
- Built-in support for Bugzilla issue tracker

PIC32MX1XX/2XX 28/36/44-PIN FAMILY

TABLE 30-10: DC CHARACTERISTICS: I/O PIN OUTPUT SPECIFICATIONS

DC CHARACTERISTICS			Standard Operating Conditions: 2.3V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \leq T_A \leq +105^{\circ}\text{C}$ for V-temp				
Param.	Symbol	Characteristic	Min.	Typ.	Max.	Units	Conditions
DO10	VOL	Output Low Voltage I/O Pins	—	—	0.4	V	$I_{OL} \leq 10 \text{ mA}$, $V_{DD} = 3.3\text{V}$
DO20	VOH	Output High Voltage I/O Pins	1.5 ⁽¹⁾	—	—	V	$I_{OH} \geq -14 \text{ mA}$, $V_{DD} = 3.3\text{V}$
			2.0 ⁽¹⁾	—	—		$I_{OH} \geq -12 \text{ mA}$, $V_{DD} = 3.3\text{V}$
			2.4	—	—		$I_{OH} \geq -10 \text{ mA}$, $V_{DD} = 3.3\text{V}$
			3.0 ⁽¹⁾	—	—		$I_{OH} \geq -7 \text{ mA}$, $V_{DD} = 3.3\text{V}$

Note 1: Parameters are characterized, but not tested.

TABLE 30-11: ELECTRICAL CHARACTERISTICS: BOR

DC CHARACTERISTICS			Standard Operating Conditions: 2.3V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \leq T_A \leq +105^{\circ}\text{C}$ for V-temp				
Param. No.	Symbol	Characteristics	Min. ⁽¹⁾	Typical	Max.	Units	Conditions
BO10	VBOR	BOR Event on VDD transition high-to-low ⁽²⁾	2.0	—	2.3	V	—

Note 1: Parameters are for design guidance only and are not tested in manufacturing.

2: Overall functional device operation at $V_{BORMIN} < V_{DD} < V_{DDMIN}$ is tested, but not characterized. All device Analog modules, such as ADC, etc., will function, but with degraded performance below V_{DDMIN} .

PIC32MX1XX/2XX 28/36/44-PIN FAMILY

TABLE 30-31: SPIx MODULE SLAVE MODE (CKE = 1) TIMING REQUIREMENTS (CONTINUED)

AC CHARACTERISTICS			Standard Operating Conditions: 2.3V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \leq T_A \leq +105^{\circ}\text{C}$ for V-temp				
Param. No.	Symbol	Characteristics ⁽¹⁾	Min.	Typical ⁽²⁾	Max.	Units	Conditions
SP51	TssH2boZ	$\overline{\text{SS}}_x \uparrow$ to SDOx Output High-Impedance (Note 4)	5	—	25	ns	—
SP52	Tsch2ssH TscL2ssH	$\overline{\text{SS}}_x \uparrow$ after SCKx Edge	Tsck + 20	—	—	ns	—
SP60	TssL2boV	SDOx Data Output Valid after $\overline{\text{SS}}_x$ Edge	—	—	25	ns	—

Note 1: These parameters are characterized, but not tested in manufacturing.

2: Data in “Typical” column is at 3.3V, 25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

3: The minimum clock period for SCKx is 50 ns.

4: Assumes 50 pF load on all SPIx pins.

PIC32MX1XX/2XX 28/36/44-PIN FAMILY

TABLE 30-39: PARALLEL MASTER PORT WRITE TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 2.3V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \leq T_A \leq +105^{\circ}\text{C}$ for V-temp				
Param. No.	Symbol	Characteristics ⁽¹⁾	Min.	Typ.	Max.	Units	Conditions
PM11	TWR	PMWR Pulse Width	—	1 TPB	—	—	—
PM12	TDVSU	Data Out Valid before PMWR or PMENB goes Inactive (data setup time)	—	2 TPB	—	—	—
PM13	TDVHOLD	PMWR or PMEMB Invalid to Data Out Invalid (data hold time)	—	1 TPB	—	—	—

Note 1: These parameters are characterized, but not tested in manufacturing.

TABLE 30-40: OTG ELECTRICAL SPECIFICATIONS

AC CHARACTERISTICS			Standard Operating Conditions: 2.3V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \leq T_A \leq +105^{\circ}\text{C}$ for V-temp				
Param. No.	Symbol	Characteristics ⁽¹⁾	Min.	Typ.	Max.	Units	Conditions
USB313	VUSB3V3	USB Voltage	3.0	—	3.6	V	Voltage on VUSB3V3 must be in this range for proper USB operation
USB315	VILUSB	Input Low Voltage for USB Buffer	—	—	0.8	V	—
USB316	VIHUSB	Input High Voltage for USB Buffer	2.0	—	—	V	—
USB318	VDIFS	Differential Input Sensitivity	—	—	0.2	V	The difference between D+ and D- must exceed this value while VCM is met
USB319	VCM	Differential Common Mode Range	0.8	—	2.5	V	—
USB320	ZOUT	Driver Output Impedance	28.0	—	44.0	Ω	—
USB321	VoL	Voltage Output Low	0.0	—	0.3	V	1.425 k Ω load connected to VUSB3V3
USB322	VoH	Voltage Output High	2.8	—	3.6	V	1.425 k Ω load connected to ground

Note 1: These parameters are characterized, but not tested in manufacturing.