



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	Active
Core Processor	MIPS32® M4K™
Core Size	32-Bit Single-Core
Speed	50MHz
Connectivity	I <sup>2</sup> C, IrDA, LINbus, PMP, SPI, UART/USART, USB OTG
Peripherals	Brown-out Detect/Reset, DMA, I <sup>2</sup> S, POR, PWM, WDT
Number of I/O	81
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	16K x 8
Voltage - Supply (Vcc/Vdd)	2.3V ~ 3.6V
Data Converters	A/D 48x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	100-TQFP
Supplier Device Package	100-TQFP (14x14)
Purchase URL	<a href="https://www.e-xfl.com/product-detail/microchip-technology/pic32mx230f128l-50i-pf">https://www.e-xfl.com/product-detail/microchip-technology/pic32mx230f128l-50i-pf</a>

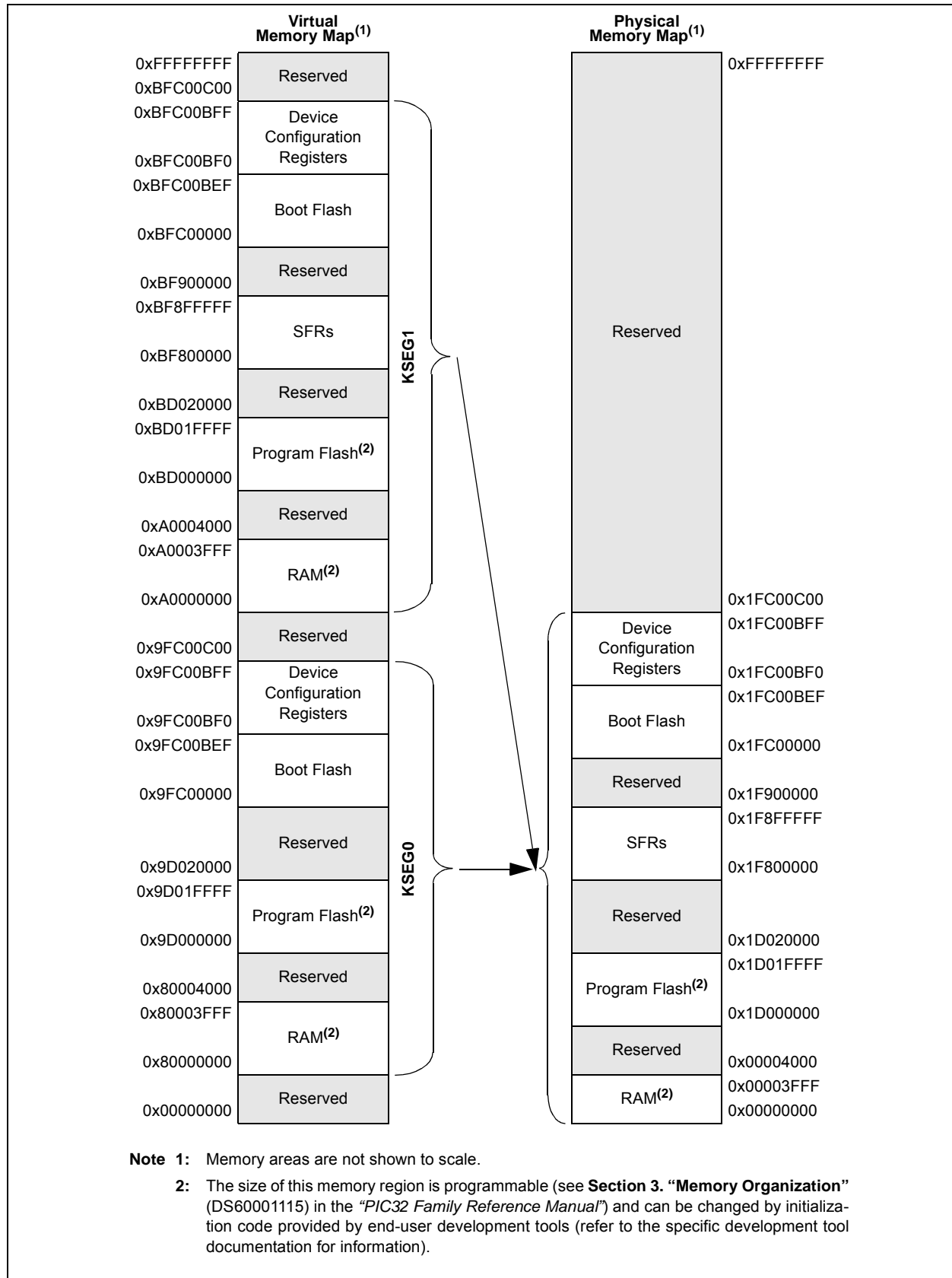
# PIC32MX1XX/2XX/5XX 64/100-PIN FAMILY

---

NOTES:

# PIC32MX1XX/2XX/5XX 64/100-PIN FAMILY

**FIGURE 4-2: MEMORY MAP FOR DEVICES WITH 128 KB OF PROGRAM MEMORY + 16 KB RAM**



# PIC32MX1XX/2XX/5XX 64/100-PIN FAMILY

## 5.0 INTERRUPT CONTROLLER

**Note:** This data sheet summarizes the features of the PIC32MX1XX/2XX/5XX 64/100-pin family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 8. “Interrupt Controller”** (DS60001108) in the *“PIC32 Family Reference Manual”*, which is available from the Microchip web site ([www.microchip.com/PIC32](http://www.microchip.com/PIC32)).

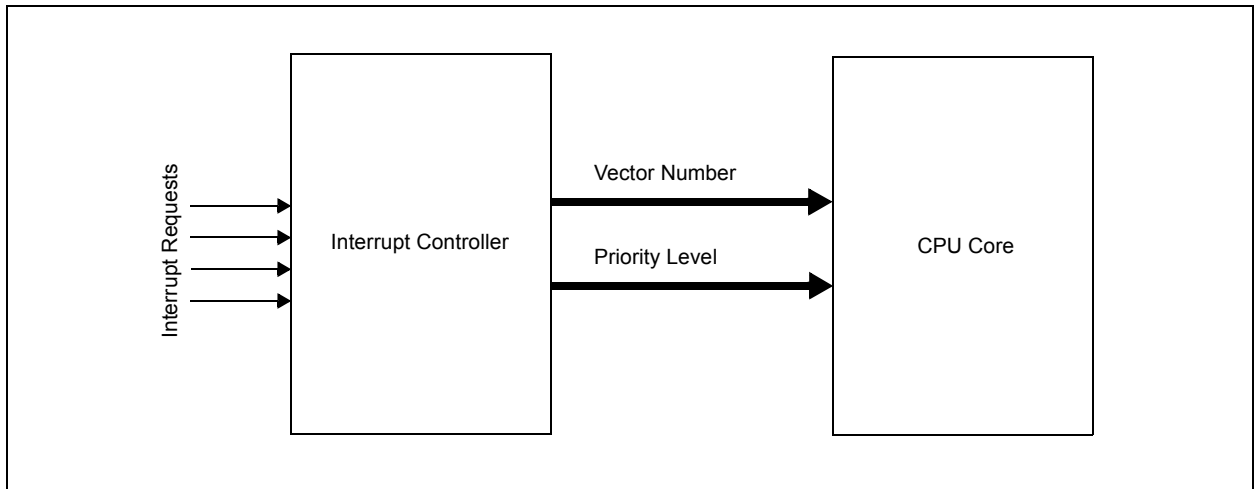
PIC32MX1XX/2XX/5XX 64/100-pin devices generate interrupt requests in response to interrupt events from peripheral modules. The interrupt control module exists externally to the CPU logic and prioritizes the interrupt events before presenting them to the CPU.

The PIC32MX1XX/2XX/5XX 64/100-pin interrupt module includes the following features:

- Up to 76 interrupt sources
- Up to 46 interrupt vectors
- Single and multi-vector mode operations
- Five external interrupts with edge polarity control
- Interrupt proximity timer
- Seven user-selectable priority levels for each vector
- Four user-selectable subpriority levels within each priority
- Software can generate any interrupt
- User-configurable interrupt vector table location
- User-configurable interrupt vector spacing

**Note:** The dedicated shadow register set is not available on these devices.

**FIGURE 5-1: INTERRUPT CONTROLLER MODULE BLOCK DIAGRAM**



# PIC32MX1XX/2XX/5XX 64/100-PIN FAMILY

**TABLE 5-1: INTERRUPT IRQ, VECTOR AND BIT LOCATION (CONTINUED)**

Interrupt Source <sup>(1)</sup>	IRQ #	Vector #	Interrupt Bit Location				Persistent Interrupt
			Flag	Enable	Priority	Sub-priority	
CNA – PORTA Input Change Interrupt	44	33	IFS1<12>	IEC1<12>	IPC8<12:10>	IPC8<9:8>	Yes
CNB – PORTB Input Change Interrupt	45	33	IFS1<13>	IEC1<13>	IPC8<12:10>	IPC8<9:8>	Yes
CNC – PORTC Input Change Interrupt	46	33	IFS1<14>	IEC1<14>	IPC8<12:10>	IPC8<9:8>	Yes
CND – PORTD Input Change Interrupt	47	33	IFS1<15>	IEC1<15>	IPC8<12:10>	IPC8<9:8>	Yes
CNE – PORTE Input Change Interrupt	48	33	IFS1<16>	IEC1<16>	IPC8<12:10>	IPC8<9:8>	Yes
CNF – PORTF Input Change Interrupt	49	33	IFS1<17>	IEC1<17>	IPC8<12:10>	IPC8<9:8>	Yes
CNG – PORTG Input Change Interrupt	50	33	IFS1<18>	IEC1<18>	IPC8<12:10>	IPC8<9:8>	Yes
PMP – Parallel Master Port	51	34	IFS1<19>	IEC1<19>	IPC8<20:18>	IPC8<17:16>	Yes
PMPE – Parallel Master Port Error	52	34	IFS1<20>	IEC1<20>	IPC8<20:18>	IPC8<17:16>	Yes
SPI2E – SPI2 Fault	53	35	IFS1<21>	IEC1<21>	IPC8<28:26>	IPC8<25:24>	Yes
SPI2RX – SPI2 Receive Done	54	35	IFS1<22>	IEC1<22>	IPC8<28:26>	IPC8<25:24>	Yes
SPI2TX – SPI2 Transfer Done	55	35	IFS1<23>	IEC1<23>	IPC8<28:26>	IPC8<25:24>	Yes
U2E – UART2 Error	56	36	IFS1<24>	IEC1<24>	IPC9<4:2>	IPC9<1:0>	Yes
U2RX – UART2 Receiver	57	36	IFS1<25>	IEC1<25>	IPC9<4:2>	IPC9<1:0>	Yes
U2TX – UART2 Transmitter	58	36	IFS1<26>	IEC1<26>	IPC9<4:2>	IPC9<1:0>	Yes
I2C2B – I2C2 Bus Collision Event	59	37	IFS1<27>	IEC1<27>	IPC9<12:10>	IPC9<9:8>	Yes
I2C2S – I2C2 Slave Event	60	37	IFS1<28>	IEC1<28>	IPC9<12:10>	IPC9<9:8>	Yes
I2C2M – I2C2 Master Event	61	37	IFS1<29>	IEC1<29>	IPC9<12:10>	IPC9<9:8>	Yes
U3E – UART3 Error	62	38	IFS1<30>	IEC1<30>	IPC9<20:18>	IPC9<17:16>	Yes
U3RX – UART3 Receiver	63	38	IFS1<31>	IEC1<31>	IPC9<20:18>	IPC9<17:16>	Yes
U3TX – UART3 Transmitter	64	38	IFS2<0>	IEC2<0>	IPC9<20:18>	IPC9<17:16>	Yes
U4E – UART4 Error	65	39	IFS2<1>	IEC2<1>	IPC9<28:26>	IPC9<25:24>	Yes
U4RX – UART4 Receiver	66	39	IFS2<2>	IEC2<2>	IPC9<28:26>	IPC9<25:24>	Yes
U4TX – UART4 Transmitter	67	39	IFS2<3>	IEC2<3>	IPC9<28:26>	IPC9<25:24>	Yes
U5E – UART5 Error <sup>(2)</sup>	68	40	IFS2<4>	IEC2<4>	IPC10<4:2>	IPC10<1:0>	Yes
U5RX – UART5 Receiver <sup>(2)</sup>	69	40	IFS2<5>	IEC2<5>	IPC10<4:2>	IPC10<1:0>	Yes
U5TX – UART5 Transmitter <sup>(2)</sup>	70	40	IFS2<6>	IEC2<6>	IPC10<4:2>	IPC10<1:0>	Yes
CTMU – CTMU Event <sup>(2)</sup>	71	41	IFS2<7>	IEC2<7>	IPC10<12:10>	IPC10<9:8>	Yes
DMA0 – DMA Channel 0	72	42	IFS2<8>	IEC2<8>	IPC10<20:18>	IPC10<17:16>	No
DMA1 – DMA Channel 1	73	43	IFS2<9>	IEC2<9>	IPC10<28:26>	IPC10<25:24>	No
DMA2 – DMA Channel 2	74	44	IFS2<10>	IEC2<10>	IPC11<4:2>	IPC11<1:0>	No
DMA3 – DMA Channel 3	75	45	IFS2<11>	IEC2<11>	IPC11<12:10>	IPC11<9:8>	No
CMP3 – Comparator 3 Interrupt	76	46	IFS2<12>	IEC2<12>	IPC11<20:18>	IPC11<17:16>	No
CAN1 – CAN1 Event	77	47	IFS2<13>	IEC2<13>	IPC11<28:26>	IPC11<25:24>	Yes
SPI3E – SPI3 Fault	78	48	IFS2<14>	IEC2<14>	IPC12<4:2>	IPC12<1:0>	Yes
SPI3RX – SPI3 Receive Done	79	48	IFS2<15>	IEC2<15>	IPC12<4:2>	IPC12<1:0>	Yes
SPI3TX – SPI3 Transfer Done	80	48	IFS2<16>	IEC2<16>	IPC12<4:2>	IPC12<1:0>	Yes
SPI4E – SPI4 Fault <sup>(2)</sup>	81	49	IFS2<17>	IEC2<17>	IPC12<12:10>	IPC12<9:8>	Yes
SPI4RX – SPI4 Receive Done <sup>(2)</sup>	82	49	IFS2<18>	IEC2<18>	IPC12<12:10>	IPC12<9:8>	Yes
SPI4TX – SPI4 Transfer Done <sup>(2)</sup>	83	49	IFS2<19>	IEC2<19>	IPC12<12:10>	IPC12<9:8>	Yes

Lowest Natural Order Priority

**Note 1:** Not all interrupt sources are available on all devices. See **TABLE 1: “PIC32MX1XX/2XX/5XX 64/100-pin Controller Family Features”** for the list of available peripherals.

**2:** This interrupt source is not available on 64-pin devices.

# PIC32MX1XX/2XX/5XX 64/100-PIN FAMILY

**REGISTER 5-2: INTSTAT: INTERRUPT 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	R/W-0	R/W-0	R/W-0
	—	—	—	—	—	SRIPL<2:0> <sup>(1)</sup>		
7:0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	VEC<5:0> <sup>(1)</sup>					

**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 31-11 **Unimplemented:** Read as '0'

bit 10-8 **SRIPL<2:0>:** Requested Priority Level bits<sup>(1)</sup>

111-000 = The priority level of the latest interrupt presented to the CPU

bit 7-6 **Unimplemented:** Read as '0'

bit 5-0 **VEC<5:0>:** Interrupt Vector bits<sup>(1)</sup>

11111-00000 = The interrupt vector that is presented to the CPU

**Note 1:** This value should only be used when the interrupt controller is configured for Single Vector mode.

**REGISTER 5-3: IPTMR: INTERRUPT PROXIMITY TIMER 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
	IPTMR<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
	IPTMR<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
	IPTMR<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
	IPTMR<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 **IPTMR<31:0>:** Interrupt Proximity Timer Reload bits

Used by the Interrupt Proximity Timer as a reload value when the Interrupt Proximity timer is triggered by an interrupt event.

# PIC32MX1XX/2XX/5XX 64/100-PIN FAMILY

## REGISTER 6-2: NVMKEY: PROGRAMMING UNLOCK 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	W-0	W-0	W-0	W-0	W-0	W-0	W-0	W-0
	NVMKEY<31:24>							
23:16	W-0	W-0	W-0	W-0	W-0	W-0	W-0	W-0
	NVMKEY<23:16>							
15:8	W-0	W-0	W-0	W-0	W-0	W-0	W-0	W-0
	NVMKEY<15:8>							
7:0	W-0	W-0	W-0	W-0	W-0	W-0	W-0	W-0
	NVMKEY<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 **NVMKEY<31:0>**: Unlock Register bits

These bits are write-only, and read as '0' on any read.

**Note:** This register is used as part of the unlock sequence to prevent inadvertent writes to the PFM.

## REGISTER 6-3: NVMADDR: FLASH ADDRESS 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
	NVMADDR<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
	NVMADDR<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
	NVMADDR<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
	NVMADDR<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 **NVMADDR<31:0>**: Flash Address bits

Bulk/Chip/PFM Erase: Address is ignored

Page Erase: Address identifies the page to erase

Row Program: Address identifies the row to program

Word Program: Address identifies the word to program

# PIC32MX1XX/2XX/5XX 64/100-PIN FAMILY

**REGISTER 9-8: DCHxECON: DMA CHANNEL 'x' EVENT 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-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
	CHAIRQ<7:0> <sup>(1)</sup>							
15:8	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
	CHSIRQ<7:0> <sup>(1)</sup>							
7:0	S-0 CFORCE	S-0 CABORT	R/W-0 PATEN	R/W-0 SIRQEN	R/W-0 AIRQEN	U-0 —	U-0 —	U-0 —

<b>Legend:</b>	S = Settable bit
R = Readable bit	W = Writable bit
-n = Value at POR	'1' = Bit is set
	U = Unimplemented bit, read as '0'
	'0' = Bit is cleared
	x = Bit is unknown

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

bit 23-16 **CHAIRQ<7:0>**: Channel Transfer Abort IRQ bits<sup>(1)</sup>

11111111 = Interrupt 255 will abort any transfers in progress and set CHAIF flag

•  
•  
•

00000001 = Interrupt 1 will abort any transfers in progress and set CHAIF flag

00000000 = Interrupt 0 will abort any transfers in progress and set CHAIF flag

bit 15-8 **CHSIRQ<7:0>**: Channel Transfer Start IRQ bits<sup>(1)</sup>

11111111 = Interrupt 255 will initiate a DMA transfer

•  
•  
•

00000001 = Interrupt 1 will initiate a DMA transfer

00000000 = Interrupt 0 will initiate a DMA transfer

bit 7 **CFORCE**: DMA Forced Transfer bit

1 = A DMA transfer is forced to begin when this bit is written to a '1'

0 = This bit always reads '0'

bit 6 **CABORT**: DMA Abort Transfer bit

1 = A DMA transfer is aborted when this bit is written to a '1'

0 = This bit always reads '0'

bit 5 **PATEN**: Channel Pattern Match Abort Enable bit

1 = Abort transfer and clear CHEN on pattern match

0 = Pattern match is disabled

bit 4 **SIRQEN**: Channel Start IRQ Enable bit

1 = Start channel cell transfer if an interrupt matching CHSIRQ occurs

0 = Interrupt number CHSIRQ is ignored and does not start a transfer

bit 3 **AIRQEN**: Channel Abort IRQ Enable bit

1 = Channel transfer is aborted if an interrupt matching CHAIRQ occurs

0 = Interrupt number CHAIRQ is ignored and does not terminate a transfer

bit 2-0 **Unimplemented:** Read as '0'

**Note 1:** See Table 5-1: "Interrupt IRQ, Vector and Bit Location" for the list of available interrupt IRQ sources.



# PIC32MX1XX/2XX/5XX 64/100-PIN FAMILY

**REGISTER 9-16: DCHxCSIZ: DMA CHANNEL 'x' CELL-SIZE 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/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CHCSIZ<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
	CHCSIZ<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 31-16 **Unimplemented:** Read as '0'

bit 15-0 **CHCSIZ<15:0>:** Channel Cell-Size bits

1111111111111111 = 65,535 bytes transferred on an event

.

.

.

0000000000000010 = 2 bytes transferred on an event

0000000000000001 = 1 byte transferred on an event

0000000000000000 = 65,536 bytes transferred on an event

**REGISTER 9-17: DCHxCPTR: DMA CHANNEL 'x' CELL POINTER 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-0	R-0	R-0	R-0	R-0	R-0	R-0
	CHCPTR<15:8>							
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	CHCPTR<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 31-16 **Unimplemented:** Read as '0'

bit 15-0 **CHCPTR<7:0>:** Channel Cell Progress Pointer bits

1111111111111111 = 65,535 bytes have been transferred since the last event

.

.

.

0000000000000001 = 1 byte has been transferred since the last event

0000000000000000 = 0 bytes have been transferred since the last event

**Note:** When in Pattern Detect mode, this register is reset on a pattern detect.

**TABLE 10-1: USB REGISTER MAP (CONTINUED)**

Virtual Address (BF88_#)	Register Name <sup>(f)</sup>	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	
5390	U1EP9	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
53A0	U1EP10	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
53B0	U1EP11	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
53C0	U1EP12	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
53D0	U1EP13	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
53E0	U1EP14	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
53F0	U1EP15	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000

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

- Note** 1: With the exception of those noted, all registers in this table (except as noted) have 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.
- 2: This register does not have associated SET and INV registers.
- 3: This register does not have associated CLR, SET and INV registers.
- 4: Reset value for this bit is undefined.

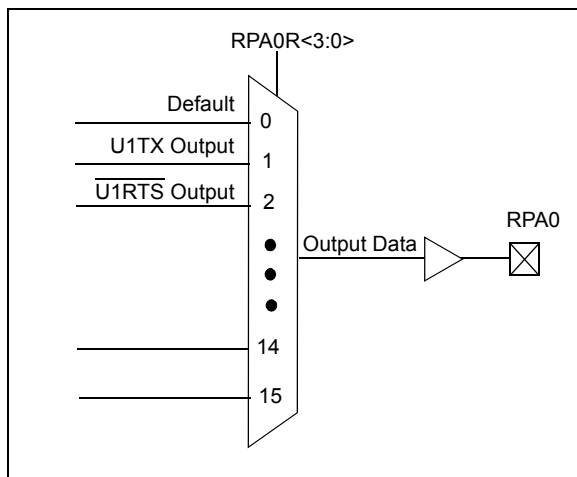
# PIC32MX1XX/2XX/5XX 64/100-PIN FAMILY

## 11.3.5 OUTPUT MAPPING

In contrast to inputs, the outputs of the peripheral pin select options are mapped on the basis of the pin. In this case, a control register associated with a particular pin dictates the peripheral output to be mapped. The RPNR registers (Register 11-2) are used to control output mapping. Like the *[pin name]*R registers, each register contains sets of 4 bit fields. The value of the bit field corresponds to one of the peripherals, and that peripheral's output is mapped to the pin (see Table 11-2 and Figure 11-3).

A null output is associated with the output register reset value of '0'. This is done to ensure that remappable outputs remain disconnected from all output pins by default.

**FIGURE 11-3: EXAMPLE OF MULTIPLEXING OF REMAPPABLE OUTPUT FOR RPA0**



## 11.3.6 CONTROLLING CONFIGURATION CHANGES

Because peripheral remapping can be changed during run time, some restrictions on peripheral remapping are needed to prevent accidental configuration changes. PIC32 devices include two features to prevent alterations to the peripheral map:

- Control register lock sequence
- Configuration bit select lock

### 11.3.6.1 Control Register Lock

Under normal operation, writes to the RPNR and *[pin name]*R registers are not allowed. Attempted writes appear to execute normally, but the contents of the registers remain unchanged. To change these registers, they must be unlocked in hardware. The register lock is controlled by the IOLOCK Configuration bit (CFGCON<13>). Setting IOLOCK prevents writes to the control registers; clearing IOLOCK allows writes.

To set or clear the IOLOCK bit, an unlock sequence must be executed. Refer to **Section 6. "Oscillator"** (DS60001112) in the *"PIC32 Family Reference Manual"* for details.

### 11.3.6.2 Configuration Bit Select Lock

As an additional level of safety, the device can be configured to prevent more than one write session to the RPNR and *[pin name]*R registers. The IOL1WAY Configuration bit (DEVCFG3<29>) blocks the IOLOCK bit from being cleared after it has been set once. If IOLOCK remains set, the register unlock procedure does not execute, and the peripheral pin select control registers cannot be written to. The only way to clear the bit and re-enable peripheral remapping is to perform a device Reset.

In the default (unprogrammed) state, IOL1WAY is set, restricting users to one write session.

## 18.0 INTER-INTEGRATED CIRCUIT (I<sup>2</sup>C)

**Note:** This data sheet summarizes the features of the PIC32MX1XX/2XX/5XX 64/100-pin family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 24. “Inter-Integrated Circuit (I<sup>2</sup>C)”** (DS60001116) in the *“PIC32 Family Reference Manual”*, which is available from the Microchip web site ([www.microchip.com/PIC32](http://www.microchip.com/PIC32)).

The I<sup>2</sup>C module provides complete hardware support for both Slave and Multi-Master modes of the I<sup>2</sup>C serial communication standard. Figure 18-1 illustrates the I<sup>2</sup>C module block diagram.

Each I<sup>2</sup>C module has a 2-pin interface: the SCLx pin is clock and the SDAx pin is data.

Each I<sup>2</sup>C module offers the following key features:

- I<sup>2</sup>C interface supporting both master and slave operation
- I<sup>2</sup>C Slave mode supports 7-bit and 10-bit addressing
- I<sup>2</sup>C Master mode supports 7-bit and 10-bit addressing
- I<sup>2</sup>C port allows bidirectional transfers between master and slaves
- Serial clock synchronization for the I<sup>2</sup>C port can be used as a handshake mechanism to suspend and resume serial transfer (SCLREL control)
- I<sup>2</sup>C supports multi-master operation; detects bus collision and arbitrates accordingly
- Provides support for address bit masking

# PIC32MX1XX/2XX/5XX 64/100-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	R/W-0	R/W-0	R/W-0
	BUSY	IRQM<1:0>		INCM<1:0>		MODE16	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> <sup>(1)</sup>		WAITM<3:0> <sup>(1)</sup>				WAITE<1:0> <sup>(1)</sup>	

### 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<15:0> by 1 every read/write cycle<sup>(2)</sup>

01 = Increment ADDR<15:0> by 1 every read/write cycle<sup>(2)</sup>

00 = No increment or decrement of address

bit 10 **MODE16:** 8/16-bit Mode bit

1 = 16-bit mode: a read or write to the data register invokes a single 16-bit transfer

0 = 8-bit mode: a read or write to the data register invokes a single 8-bit transfer

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

11 = Master mode 1 (PMCSx, PMRD/PMWR, PMENB, PMA<x:0>, PMD<7:0> and PMD<8:15><sup>(3)</sup>)

10 = Master mode 2 (PMCSx, PMRD, PMWR, PMA<x:0>, PMD<7:0> and PMD<8:15><sup>(3)</sup>)

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

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

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

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)

**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 bits, A15 and A14, are not subject to automatic increment/decrement if configured as Chip Select CS2 and CS1.

**3:** These pins are active when MODE16 = 1 (16-bit mode).

# PIC32MX1XX/2XX/5XX 64/100-PIN FAMILY

## 21.0 REAL-TIME CLOCK AND CALENDAR (RTCC)

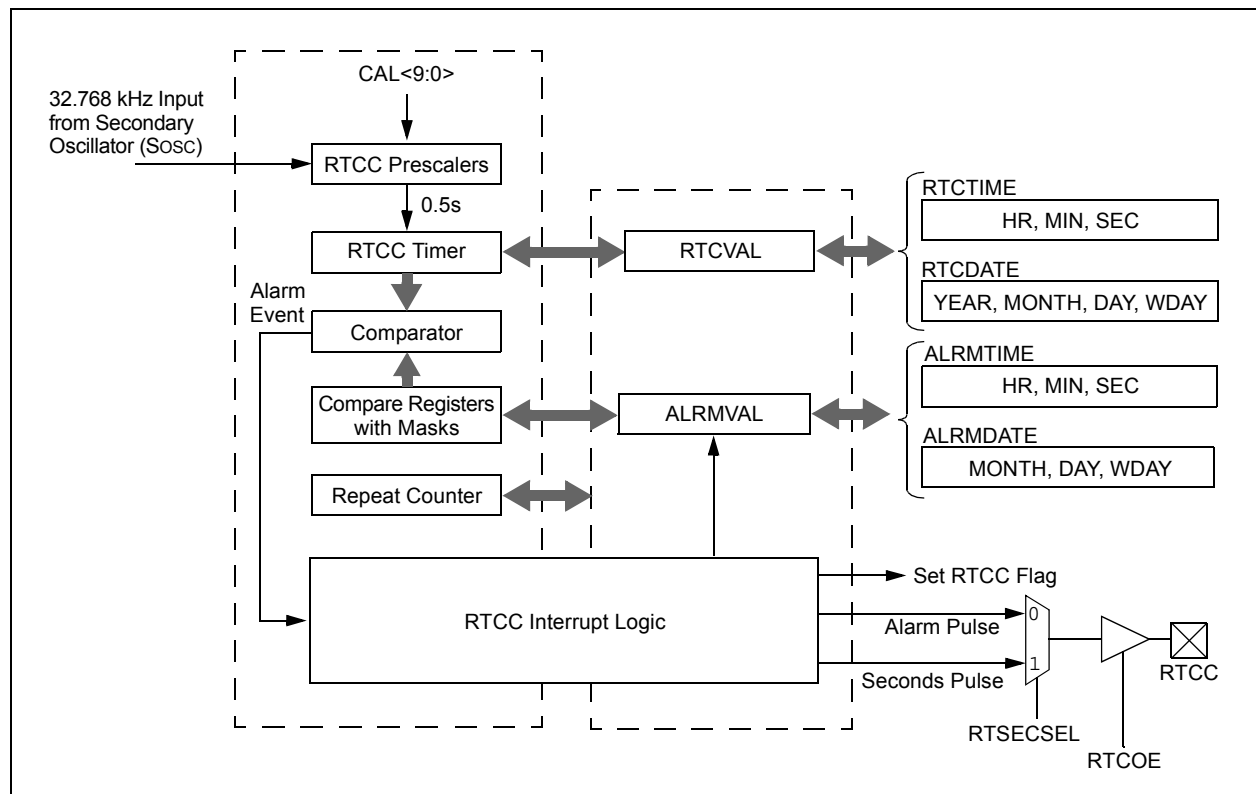
**Note:** This data sheet summarizes the features of the PIC32MX1XX/2XX/5XX 64/100-pin family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 29. “Real-Time Clock and Calendar (RTCC)”** (DS60001125) in the “PIC32 Family Reference Manual”, which is available from the Microchip web site ([www.microchip.com/PIC32](http://www.microchip.com/PIC32)).

The PIC32 RTCC module is intended for applications in which accurate time must be maintained for extended periods of time with minimal or no CPU intervention. Low-power optimization provides extended battery lifetime while keeping track of time.

The following are the key features of this module:

- Time: hours, minutes and seconds
- 24-hour format (military time)
- Visibility of one-half second period
- Provides calendar: Weekday, date, month and year
- Alarm intervals are configurable for half of a second, one second, 10 seconds, one minute, 10 minutes, one hour, one day, one week, one month and one year
- Alarm repeat with decrementing counter
- Alarm with indefinite repeat: Chime
- Year range: 2000 to 2099
- Leap year correction
- BCD format for smaller firmware overhead
- Optimized for long-term battery operation
- Fractional second synchronization
- User calibration of the clock crystal frequency with auto-adjust
- Calibration range:  $\pm 0.66$  seconds error per month
- Calibrates up to 260 ppm of crystal error
- Requirements: External 32.768 kHz clock crystal
- Alarm pulse or seconds clock output on RTCC pin

**FIGURE 21-1: RTCC BLOCK DIAGRAM**



# PIC32MX1XX/2XX/5XX 64/100-PIN FAMILY

## REGISTER 23-11: C1FLTCON1: CAN FILTER CONTROL REGISTER 1

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
	FLTEN7	MSEL7<1:0>		FSEL7<4:0>				
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
	FLTEN6	MSEL6<1:0>		FSEL6<4:0>				
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
	FLTEN5	MSEL5<1:0>		FSEL5<4:0>				
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
	FLTEN4	MSEL4<1:0>		FSEL4<4:0>				

### 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 **FLTEN7**: Filter 7 Enable bit

- 1 = Filter is enabled
- 0 = Filter is disabled

bit 30-29 **MSEL7<1:0>**: Filter 7 Mask Select bits

- 11 = Acceptance Mask 3 selected
- 10 = Acceptance Mask 2 selected
- 01 = Acceptance Mask 1 selected
- 00 = Acceptance Mask 0 selected

bit 28-24 **FSEL7<4:0>**: FIFO Selection bits

- 11111 = Reserved
- .
- .
- .
- 10000 = Reserved
- 01111 = Message matching filter is stored in FIFO buffer 15
- .
- .
- .
- 00000 = Message matching filter is stored in FIFO buffer 0

bit 23 **FLTEN6**: Filter 6 Enable bit

- 1 = Filter is enabled
- 0 = Filter is disabled

bit 22-21 **MSEL6<1:0>**: Filter 6 Mask Select bits

- 11 = Acceptance Mask 3 selected
- 10 = Acceptance Mask 2 selected
- 01 = Acceptance Mask 1 selected
- 00 = Acceptance Mask 0 selected

**Note:** The bits in this register can only be modified if the corresponding filter enable (FLTENn) bit is '0'.

# PIC32MX1XX/2XX/5XX 64/100-PIN FAMILY

## 27.4 Peripheral Module Disable

The Peripheral Module Disable (PMD) registers provide a method to disable a peripheral module by stopping all clock sources supplied to that module. When a peripheral is disabled using the appropriate PMD control bit, the peripheral is in a minimum power consumption state. The control and status registers associated with the peripheral are also disabled, so writes to those registers do not have effect and read values are invalid.

To disable a peripheral, the associated PMDx bit must be set to '1'. To enable a peripheral, the associated PMDx bit must be cleared (default). See Table 27-1 for more information.

**Note:** Disabling a peripheral module while its ON bit is set, may result in undefined behavior. The ON bit for the associated peripheral module must be cleared prior to disable a module via the PMDx bits.

**TABLE 27-1: PERIPHERAL MODULE DISABLE BITS AND LOCATIONS**

Peripheral <sup>(1)</sup>	PMDx bit Name <sup>(1)</sup>	Register Name and Bit Location
ADC1	AD1MD	PMD1<0>
CTMU	CTMUMD	PMD1<8>
Comparator Voltage Reference	CVRMD	PMD1<12>
Comparator 1	CMP1MD	PMD2<0>
Comparator 2	CMP2MD	PMD2<1>
Comparator 3	CMP3MD	PMD2<2>
Input Capture 1	IC1MD	PMD3<0>
Input Capture 2	IC2MD	PMD3<1>
Input Capture 3	IC3MD	PMD3<2>
Input Capture 4	IC4MD	PMD3<3>
Input Capture 5	IC5MD	PMD3<4>
Output Compare 1	OC1MD	PMD3<16>
Output Compare 2	OC2MD	PMD3<17>
Output Compare 3	OC3MD	PMD3<18>
Output Compare 4	OC4MD	PMD3<19>
Output Compare 5	OC5MD	PMD3<20>
Timer1	T1MD	PMD4<0>
Timer2	T2MD	PMD4<1>
Timer3	T3MD	PMD4<2>
Timer4	T4MD	PMD4<3>
Timer5	T5MD	PMD4<4>
UART1	U1MD	PMD5<0>
UART2	U2MD	PMD5<1>
UART3	U3MD	PMD5<2>
UART4	U4MD	PMD5<3>
UART5	U5MD	PMD5<4>
SPI1	SPI1MD	PMD5<8>
SPI2	SPI2MD	PMD5<9>
SPI3	SPI3MD	PMD5<10>
SPI4	SPI4MD	PMD5<11>
I2C1	I2C1MD	PMD5<16>
I2C2	I2C2MD	PMD5<17>
USB <sup>(2)</sup>	USBMD	PMD5<24>
CAN	CAN1MD	PMD5<28>
RTCC	RTCCMD	PMD6<0>
Reference Clock Output	REFOMD	PMD6<1>
PMP	PMPMD	PMD6<16>

**Note 1:** Not all modules and associated PMDx bits are available on all devices. See **TABLE 1: "PIC32MX1XX/2XX/5XX 64/100-pin Controller Family Features"** for the list of available peripherals.

**2:** Module must not be busy after clearing the associated ON bit and prior to setting the USBMD bit.



# PIC32MX1XX/2XX/5XX 64/100-PIN FAMILY

---

## REGISTER 28-3: DEVCFG2: DEVICE CONFIGURATION WORD 2 (CONTINUED)

bit 2-0 **FPLLIDIV<2:0>**: PLL Input Divider bits

111 = 12x divider

110 = 10x divider

101 = 6x divider

100 = 5x divider

011 = 4x divider

010 = 3x divider

001 = 2x divider

000 = 1x divider

**Note 1:** This bit is available on PIC32MX2XX/5XX devices only.

## 29.0 INSTRUCTION SET

The PIC32MX1XX/2XX/5XX 64/100-pin family instruction set complies with the MIPS32® Release 2 instruction set architecture. The PIC32 device family does not support the following features:

- Core extend instructions
- Coprocessor 1 instructions
- Coprocessor 2 instructions

<p><b>Note:</b> Refer to “MIPS32® Architecture for Programmers Volume II: The MIPS32® Instruction Set” at <a href="http://www.imgtec.com">www.imgtec.com</a> for more information.</p>
--

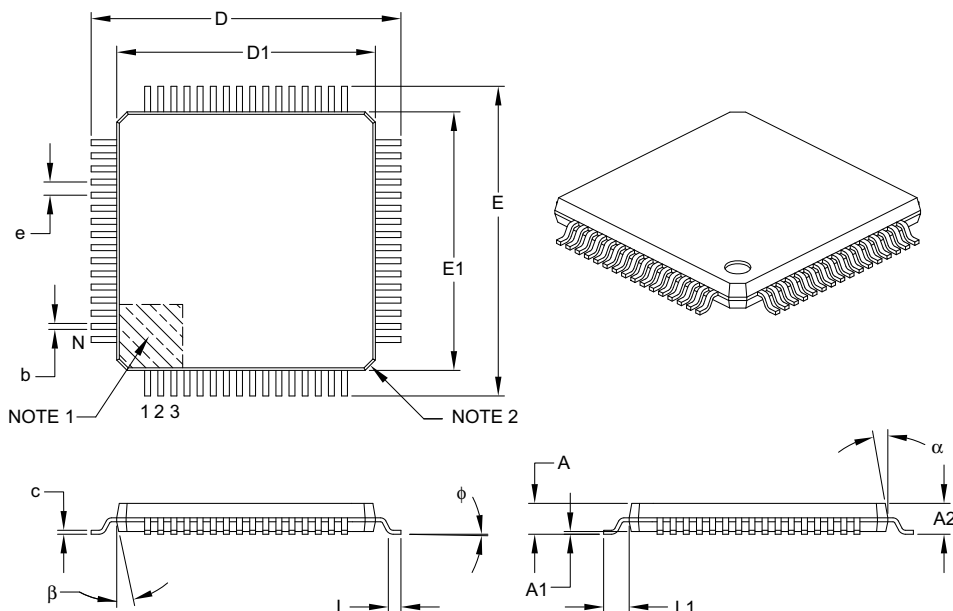
# PIC32MX1XX/2XX/5XX 64/100-PIN FAMILY

## 34.2 Package Details

The following sections give the technical details of the packages.

### 64-Lead Plastic Thin Quad Flatpack (PT) – 10x10x1 mm Body, 2.00 mm [TQFP]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Number of Leads	N	64		
Lead Pitch	e	0.50 BSC		
Overall Height	A	–	–	1.20
Molded Package Thickness	A2	0.95	1.00	1.05
Standoff	A1	0.05	–	0.15
Foot Length	L	0.45	0.60	0.75
Footprint	L1	1.00 REF		
Foot Angle	φ	0°	3.5°	7°
Overall Width	E	12.00 BSC		
Overall Length	D	12.00 BSC		
Molded Package Width	E1	10.00 BSC		
Molded Package Length	D1	10.00 BSC		
Lead Thickness	c	0.09	–	0.20
Lead Width	b	0.17	0.22	0.27
Mold Draft Angle Top	α	11°	12°	13°
Mold Draft Angle Bottom	β	11°	12°	13°

#### Notes:

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- Chamfers at corners are optional; size may vary.
- Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.25 mm per side.
- Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

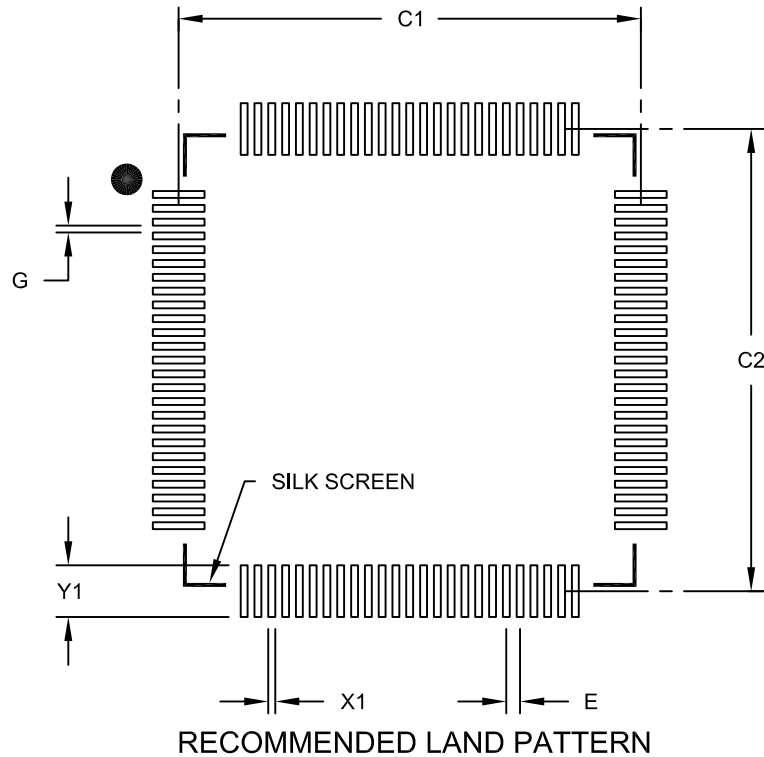
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-085B

# PIC32MX1XX/2XX/5XX 64/100-PIN FAMILY

100-Lead Plastic Thin Quad Flatpack (PT)-12x12x1mm Body, 2.00 mm Footprint [TQFP]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>



		Units	MILLIMETERS		
Dimension Limits			MIN	NOM	MAX
Contact Pitch	E			0.40 BSC	
Contact Pad Spacing	C1			13.40	
Contact Pad Spacing	C2			13.40	
Contact Pad Width (X100)	X1				0.20
Contact Pad Length (X100)	Y1				1.50
Distance Between Pads	G		0.20		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2100B

---

**Note the following details of the code protection feature on Microchip devices:**

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

---

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE. Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

*Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.*

**QUALITY MANAGEMENT SYSTEM  
CERTIFIED BY DNV  
= ISO/TS 16949 =**

### Trademarks

The Microchip name and logo, the Microchip logo, AnyRate, dsPIC, FlashFlex, flexPWR, Helder, JukeBlox, KeeLoq, KeeLoq logo, Klear, LANCheck, LINK MD, MediaLB, MOST, MOST logo, MPLAB, OptoLyzer, PIC, PICSTART, PIC32 logo, RightTouch, SpyNIC, SST, SST Logo, SuperFlash and UNI/O are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

ClockWorks, The Embedded Control Solutions Company, ETHERSYNCH, Hyper Speed Control, HyperLight Load, IntelliMOS, mTouch, Precision Edge, and QUIET-WIRE are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Analog-for-the-Digital Age, Any Capacitor, AnyIn, AnyOut, BodyCom, chipKIT, chipKIT logo, CodeGuard, dsPICDEM, dsPICDEM.net, Dynamic Average Matching, DAM, ECAN, EtherGREEN, In-Circuit Serial Programming, ICSP, Inter-Chip Connectivity, JitterBlocker, KlearNet, KlearNet logo, MiWi, motorBench, MPASM, MPF, MPLAB Certified logo, MPLIB, MPLINK, MultiTRAK, NetDetach, Omniscient Code Generation, PICDEM, PICDEM.net, PICKit, PICtail, PureSilicon, RightTouch logo, REAL ICE, Ripple Blocker, Serial Quad I/O, SQL, SuperSwitcher, SuperSwitcher II, Total Endurance, TSHARC, USBCheck, VariSense, ViewSpan, WiperLock, Wireless DNA, and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

Silicon Storage Technology is a registered trademark of Microchip Technology Inc. in other countries.

GestIC is a registered trademarks of Microchip Technology Germany II GmbH & Co. KG, a subsidiary of Microchip Technology Inc., in other countries.

All other trademarks mentioned herein are property of their respective companies.

© 2014-2016, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

ISBN: 978-1-5224-0488-0