Microchip Technology - PIC32MZ2048EFH124-E/TL Datasheet

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	MIPS32® M-Class
Core Size	32-Bit Single-Core
Speed	180MHz
Connectivity	CANbus, EBI/EMI, Ethernet, I²C, PMP, SPI, SQI, UART/USART, USB OTG
Peripherals	Brown-out Detect/Reset, DMA, I²S, POR, PWM, WDT
Number of I/O	97
Program Memory Size	2MB (2M x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	512K x 8
Voltage - Supply (Vcc/Vdd)	2.1V ~ 3.6V
Data Converters	A/D 48x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C
Mounting Type	Surface Mount
Package / Case	124-VFTLA Dual Rows, Exposed Pad
Supplier Device Package	124-VTLA (9x9)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic32mz2048efh124-e-tl

TABLE 1-22: JTAG, TRACE, AND PROGRAMMING/DEBUGGING PINOUT I/O DESCRIPTIONS

Pin Name	Pin Number				Pin Type	Buffer Type	Description
	64-pin QFN/ TQFP	100-pin TQFP	124-pin VTLA	144-pin TQFP/ LQFP			
JTAG							
TCK	27	38	B21	56	I	ST	JTAG Test Clock Input Pin
TDI	28	39	A26	57	I	ST	JTAG Test Data Input Pin
TDO	24	40	B22	58	O	—	JTAG Test Data Output Pin
TMS	23	17	A11	22	I	ST	JTAG Test Mode Select Pin
Trace							
TRCLK	57	89	A61	129	O	—	Trace Clock
TRD0	58	97	B55	141	O	—	Trace Data bits 0-3
TRD1	61	96	A65	140	O	—	
TRD2	62	95	B54	139	O	—	
TRD3	63	90	B51	130	O	—	
Programming/Debugging							
PGED1	16	25	A18	36	I/O	ST	Data I/O pin for Programming/Debugging Communication Channel 1
PGEC1	15	24	A17	35	I	ST	Clock input pin for Programming/Debugging Communication Channel 1
PGED2	18	27	A19	38	I/O	ST	Data I/O pin for Programming/Debugging Communication Channel 2
PGEC2	17	26	B14	37	I	ST	Clock input pin for Programming/Debugging Communication Channel 2
$\overline{MCLR}$	9	15	A10	20	I/P	ST	Master Clear (Reset) input. This pin is an active-low Reset to the device.

Legend: CMOS = CMOS-compatible input or output; ST = Schmitt Trigger input with CMOS levels; TTL = Transistor-transistor Logic input buffer; Analog = Analog input; O = Output; PPS = Peripheral Pin Select; P = Power; I = Input

The System Bus arbitration scheme implements a non-programmable, Least Recently Serviced (LRS) priority, which provides Quality Of Service (QOS) for most initiators. However, some initiators can use Fixed High Priority (HIGH) arbitration to guarantee their access to data.

The arbitration scheme for the available initiators is shown in Table 4-5.

TABLE 4-5: INITIATOR ID AND QOS

Name	ID	QOS
CPU	1	LRS(1)
CPU	2	HIGH(1,2)
DMA Read	3	LRS(1)
DMA Read	4	HIGH(1,2)
DMA Write	5	LRS(1)
DMA Write	6	HIGH(1,2)
USB	7	LRS
Ethernet Read	8	LRS
Ethernet Write	9	LRS
CAN1	10	LRS
CAN2	11	LRS
SQI1	12	LRS
Flash Controller	13	HIGH(2)
Crypto	14	LRS

Note 1: When accessing SRAM, the DMAPRI bit (CFGCON<25>) and the CPUPRI bit (CFGCON<24>) provide arbitration control for the DMA and CPU (when servicing an interrupt (i.e., EXL = 1)), respectively, by selecting the use of LRS or HIGH When using HIGH, the DMA and CPU get arbitration preference over all initiators using LRS.

2: Using HIGH arbitration can have serious negative effects on other initiators. Therefore, it is recommended to not enable this type of arbitration for an initiator that uses significant system bandwidth. HIGH arbitration is intended to be used for low bandwidth applications that require low latency, such as LCC graphics applications.

4.3 Permission Access and System Bus Registers

The System Bus on PIC32MZ EF family of microcontrollers provides access control capabilities for the transaction initiators on the System Bus.

The System Bus divides the entire memory space into fourteen target regions and permits access to each target by initiators via permission groups. Four Permission Groups (0 through 3) can be assigned to each initiator. Each permission group is independent of the others and can have exclusive or shared access to a region.

Using the CFGPG register (see Register 34-10 in **Section 34.0 "Special Features"**), Boot firmware can assign a permission group to each initiator, which can make requests on the System Bus.

The available targets and their regions, as well as the associated control registers to assign protection, are described and listed in Table 4-6.

Register 4-2 through Register 4-10 are used for setting and controlling access permission groups and regions.

To change these registers, they must be unlocked in hardware. The register lock is controlled by the PGLOCK Configuration bit (CFGCON<11>). Setting PGLOCK prevents writes to the control registers; clearing PGLOCK allows writes.

To set or clear the PGLOCK bit, an unlock sequence must be executed. Refer to **Section 42. "Oscillators with Enhanced PLL"** in the *"PIC32 Family Reference Manual"* for details.

REGISTER 4-10: SBTxWRy: SYSTEM BUS TARGET ‘x’ REGION ‘y’ WRITE PERMISSIONS REGISTER (‘x’ = 0-13; ‘y’ = 0-8)

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-1	R/W-1	R/W-1
	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0

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

bit 31-4 **Unimplemented:** Read as ‘0’

bit 3 **Group3:** Group 3 Write Permissions bits
1 = Privilege Group 3 has write permission
0 = Privilege Group 3 does not have write permission

bit 2 **Group2:** Group 2 Write Permissions bits
1 = Privilege Group 2 has write permission
0 = Privilege Group 2 does not have write permission

bit 1 **Group1:** Group 1 Write Permissions bits
1 = Privilege Group 1 has write permission
0 = Privilege Group 1 does not have write permission

bit 0 **Group0:** Group 0 Write Permissions bits
1 = Privilege Group 0 has write permission
0 = Privilege Group 0 does not have write permission

Note 1: Refer to Table 4-6 for the list of available targets and their descriptions.
2: For some target regions, certain bits in this register are read-only with preset values. See Table 4-6 for more information.

TABLE 7-3: INTERRUPT REGISTER MAP (CONTINUED)

Virtual Address (BF81_#)	Register Name(1)	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	
01F0	IPC11	31:16	—	—	—	ADCDC2IP<2:0>			ADCDC2IS<1:0>		—	—	—	ADCDC1IP<2:0>			ADCDC1IS<1:0>		0000
		15:0	—	—	—	ADCFIFOIP<2:0>			ADCFIFOIS<1:0>		—	—	—	ADCIP<2:0>			ADCIS<1:0>		0000
0200	IPC12	31:16	—	—	—	ADCDC6IP<2:0>			ADCDC6IS<1:0>		—	—	—	ADCDC5IP<2:0>			ADCDC5IS<1:0>		0000
		15:0	—	—	—	ADCDC4IP<2:0>			ADCDC4IS<1:0>		—	—	—	ADCDC3IP<2:0>			ADCDC3IS<1:0>		0000
0210	IPC13	31:16	—	—	—	ADCDF4IP<2:0>			ADCDF4IS<1:0>		—	—	—	ADCDF3IP<2:0>			ADCDF3IS<1:0>		0000
		15:0	—	—	—	ADCDF2IP<2:0>			ADCDF2IS<1:0>		—	—	—	ADCDF1IP<2:0>			ADCDF1IS<1:0>		0000
0220	IPC14	31:16	—	—	—	ADCD0IP<2:0>			ADCD0IS<1:0>		—	—	—	ADCDFLTIP<2:0>			ADCDFLTIS<1:0>		0000
		15:0	—	—	—	ADCDF6IP<2:0>			ADCDF6IS<1:0>		—	—	—	ADCDF5IP<2:0>			ADCDF5IS<1:0>		0000
0230	IPC15	31:16	—	—	—	ADCD4IP<2:0>			ADCD4IS<1:0>		—	—	—	ADCD3IP<2:0>			ADCD3IS<1:0>		0000
		15:0	—	—	—	ADCD2IP<2:0>			ADCD2IS<1:0>		—	—	—	ADCD1IP<2:0>			ADCD1IS<1:0>		0000
0240	IPC16	31:16	—	—	—	ADCD8IP<2:0>			ADCD8IS<1:0>		—	—	—	ADCD7IP<2:0>			ADCD7IS<1:0>		0000
		15:0	—	—	—	ADCD6IP<2:0>			ADCD6IS<1:0>		—	—	—	ADCD5IP<2:0>			ADCD5IS<1:0>		0000
0250	IPC17	31:16	—	—	—	ADCD12IP<2:0>			ADCD12IS<1:0>		—	—	—	ADCD11IP<2:0>			ADCD11IS<1:0>		0000
		15:0	—	—	—	ADCD10IP<2:0>			ADCD10IS<1:0>		—	—	—	ADCD9IP<2:0>			ADCD9IS<1:0>		0000
0260	IPC18	31:16	—	—	—	ADCD16IP<2:0>			ADCD16IS<1:0>		—	—	—	ADCD15IP<2:0>			ADCD15IS<1:0>		0000
		15:0	—	—	—	ADCD14IP<2:0>			ADCD14IS<1:0>		—	—	—	ADCD13IP<2:0>			ADCD13IS<1:0>		0000
0270	IPC19	31:16	—	—	—	ADCD20IP<2:0>(2)			ADCD20IS<1:0>(2)		—	—	—	ADCD19IP<2:0>(2)			ADCD19IS<1:0>(2)		0000
		15:0	—	—	—	ADCD18IP<2:0>			ADCD18IS<1:0>		—	—	—	ADCD17IP<2:0>			ADCD17IS<1:0>		0000
0280	IPC20	31:16	—	—	—	ADCD24IP<2:0>(2)			ADCD24IS<1:0>(2)		—	—	—	ADCD23IP<2:0>(2)			ADCD23IS<1:0>(2)		0000
		15:0	—	—	—	ADCD22IP<2:0>(2)			ADCD22IS<1:0>(2)		—	—	—	ADCD21IP<2:0>(2)			ADCD21IS<1:0>(2)		0000
0290	IPC21	31:16	—	—	—	ADCD28IP<2:0>(2)			ADCD28IS<1:0>(2)		—	—	—	ADCD27IP<2:0>(2)			ADCD27IS<1:0>(2)		0000
		15:0	—	—	—	ADCD26IP<2:0>(2)			ADCD26IS<1:0>(2)		—	—	—	ADCD25IP<2:0>(2)			ADCD25IS<1:0>(2)		0000
02A0	IPC22	31:16	—	—	—	ADCD32IP<2:0>(2)			ADCD32IS<1:0>(2)		—	—	—	ADCD31IP<2:0>(2)			ADCD31IS<1:0>(2)		0000
		15:0	—	—	—	ADCD30IP<2:0>(2)			ADCD30IS<1:0>(2)		—	—	—	ADCD29IP<2:0>(2)			ADCD29IS<1:0>(2)		0000
02B0	IPC23	31:16	—	—	—	ADCD36IP<2:0>(2,4)			ADCD36IS<1:0>(2,4)		—	—	—	ADCD35IP<2:0>(2,4)			ADCD35IS<1:0>(2,4)		0000
		15:0	—	—	—	ADCD34IP<2:0>(2)			ADCD34IS<1:0>(2)		—	—	—	ADCD33IP<2:0>(2)			ADCD33IS<1:0>(2)		0000
02C0	IPC24	31:16	—	—	—	ADCD40IP<2:0>(2,4)			ADCD40IS<1:0>(2,4)		—	—	—	ADCD39IP<2:0>(2,4)			ADCD39IS<1:0>(2,4)		0000
		15:0	—	—	—	ADCD38IP<2:0>(2,4)			ADCD38IS<1:0>(2,4)		—	—	—	ADCD37IP<2:0>(2,4)			ADCD37IS<1:0>(2,4)		0000
02D0	IPC25	31:16	—	—	—	ADCD44IP<2:0>			ADCD44IS<1:0>		—	—	—	ADCD43IP<2:0>			ADCD43IS<1:0>		0000
		15:0	—	—	—	ADCD42IP<2:0>(2,4)			ADCD42IS<1:0>(2,4)		—	—	—	ADCD41IP<2:0>(2,4)			ADCD41IS<1:0>(2,4)		0000

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

Note 1: All registers in this table with the exception of the OFFx registers, have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8 and 0xC, respectively. See **Section 12.3 "CLR, SET, and INV Registers"** for more information.

2: This bit or register is not available on 64-pin devices.

3: This bit or register is not available on devices without a CAN module.

4: This bit or register is not available on 100-pin devices.

5: Bits 31 and 30 are not available on 64-pin and 100-pin devices; bits 29 through 14 are not available on 64-pin devices.

6: Bits 31, 30, 29, and bits 5 through 0 are not available on 64-pin and 100-pin devices; bit 31 is not available on 124-pin devices; bit 22 is not available on 64-pin devices.

7: This bit or register is not available on devices without a Crypto module.

8: This bit or register is not available on 124-pin devices.

TABLE 7-3: INTERRUPT REGISTER MAP (CONTINUED)

Virtual Address (BF81_#)	Register Name(1)	Bit Range	Bits 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	All Resets
07A4	OFF153	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>		0000
		15:0	VOFF<15:1>															—	0000
07A8	OFF154	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>		0000
		15:0	VOFF<15:1>															—	0000
07AC	OFF155	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>		0000
		15:0	VOFF<15:1>															—	0000
07B0	OFF156	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>		0000
		15:0	VOFF<15:1>															—	0000
07B4	OFF157	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>		0000
		15:0	VOFF<15:1>															—	0000
07B8	OFF158	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>		0000
		15:0	VOFF<15:1>															—	0000
07BC	OFF159	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>		0000
		15:0	VOFF<15:1>															—	0000
07C0	OFF160	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>		0000
		15:0	VOFF<15:1>															—	0000
07C4	OFF161	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>		0000
		15:0	VOFF<15:1>															—	0000
07C8	OFF162	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>		0000
		15:0	VOFF<15:1>															—	0000
07CC	OFF163	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>		0000
		15:0	VOFF<15:1>															—	0000
07D0	OFF164	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>		0000
		15:0	VOFF<15:1>															—	0000
07D4	OFF165	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>		0000
		15:0	VOFF<15:1>															—	0000
07D8	OFF166	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>		0000
		15:0	VOFF<15:1>															—	0000
07DC	OFF167	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>		0000
		15:0	VOFF<15:1>															—	0000

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

Note 1: All registers in this table with the exception of the OFFx registers, have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8 and 0xC, respectively. See **Section 12.3 "CLR, SET, and INV Registers"** for more information.

2: This bit or register is not available on 64-pin devices.

3: This bit or register is not available on devices without a CAN module.

4: This bit or register is not available on 100-pin devices.

5: Bits 31 and 30 are not available on 64-pin and 100-pin devices; bits 29 through 14 are not available on 64-pin devices.

6: Bits 31, 30, 29, and bits 5 through 0 are not available on 64-pin and 100-pin devices; bit 31 is not available on 124-pin devices; bit 22 is not available on 64-pin devices.

7: This bit or register is not available on devices without a Crypto module.

8: This bit or register is not available on 124-pin devices.

REGISTER 8-7: SLEWCON: OSCILLATOR SLEW 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	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	SYSDIV<3:0>(1)			
15:8	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-1	R/W-0
	—	—	—	—	—	SLWDIV<2:0>		
7:0	U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R-0, HS, HC
	—	—	—	—	—	UPEN	DNEN	BUSY

Legend:	HC = Hardware Cleared	HS = Hardware Set	
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 31-20 **Unimplemented:** Read as ‘0’

bit 19-16 **SYSDIV<3:0>:** System Clock Divide Control bits(1)
1111 = SYSCLK is divided by 16
1110 = SYSCLK is divided by 15
•
•
•
0010 = SYSCLK is divided by 3
0001 = SYSCLK is divided by 2
0000 = SYSCLK is not divided

bit 15-11 **Unimplemented:** Read as ‘0’

bit 10-8 **SLWDIV<2:0>:** Slew Divisor Steps Control bits
These bits control the maximum division steps used when slewing during a frequency change.
111 = Steps are divide by 128, 64, 32, 16, 8, 4, 2, and then no divisor
110 = Steps are divide by 64, 32, 16, 8, 4, 2, and then no divisor
101 = Steps are divide by 32, 16, 8, 4, 2, and then no divisor
100 = Steps are divide by 16, 8, 4, 2, and then no divisor
011 = Steps are divide by 8, 4, 2, and then no divisor
010 = Steps are divide by 4, 2, and then no divisor
001 = Steps are divide by 2, and then no divisor
000 = No divisor is used during slewing

Note: The steps apply in reverse order (i.e., 2, 4, 8, etc.) during a downward frequency change.

bit 7-3 **Unimplemented:** Read as ‘0’

bit 2 **UPEN:** Upward Slew Enable bit
1 = Slewing enabled for switching to a higher frequency
0 = Slewing disabled for switching to a higher frequency

bit 1 **DNEN:** Downward Slew Enable bit
1 = Slewing enabled for switching to a lower frequency
0 = Slewing disabled for switching to a lower frequency

bit 0 **BUSY:** Clock Switching Slewing Active Status bit
1 = Clock frequency is being actively slewed to the new frequency
0 = Clock switch has reached its final value

Note 1: The SYSDIV<3:0> bit settings are ignored if both UPEN and DNEN = 0, and SYSCLK will be divided by 1.

REGISTER 11-22: USBDMAxA: USB DMA CHANNEL ‘x’ MEMORY ADDRESS REGISTER (‘x’ = 1-8)

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
	DMAADDR<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
	DMAADDR<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
	DMAADDR<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R-0	R-0
	DMAADDR<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 **DMAADDR<31:0>:** DMA Memory Address bits

This register identifies the current memory address of the corresponding DMA channel. The initial memory address written to this register during initialization must have a value such that its modulo 4 value is equal to ‘0’. The lower two bits of this register are read only and cannot be set by software. As the DMA transfer progresses, the memory address will increment as bytes are transferred.

REGISTER 11-23: USBDMAxN: USB DMA CHANNEL ‘x’ COUNT REGISTER (‘X’ = 1-8)

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
	DMACOUNT<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
	DMACOUNT<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
	DMACOUNT<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
	DMACOUNT<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 **DMACOUNT<31:0>:** DMA Transfer Count bits

This register identifies the current DMA count of the transfer. Software will set the initial count of the transfer which identifies the entire transfer length. As the count progresses this count is decremented as bytes are transferred.

TABLE 12-23: PERIPHERAL PIN SELECT OUTPUT 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	
1538	RPA14R(1)	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPA14R<3:0>				0000
153C	RPA15R(1)	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPA15R<3:0>				0000
1540	RPB0R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPB0R<3:0>				0000
1544	RPB1R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPB1R<3:0>				0000
1548	RPB2R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPB2R<3:0>				0000
154C	RPB3R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPB3R<3:0>				0000
1554	RPB5R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPB5R<3:0>				0000
1558	RPB6R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPB6R<3:0>				0000
155C	RPB7R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPB7R<3:0>				0000
1560	RPB8R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPB8R<3:0>				0000
1564	RPB9R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPB9R<3:0>				0000
1568	RPB10R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPB10R<3:0>				0000
1578	RPB14R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPB14R<3:0>				0000
157C	RPB15R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPB15R<3:0>				0000
1584	RPC1R(1)	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPC1R<3:0>				0000
1588	RPC2R(1)	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPC2R<3:0>				0000
158C	RPC3R(1)	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPC3R<3:0>				0000
1590	RPC4R(1)	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPC4R<3:0>				0000

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

Note 1: This register is not available on 64-pin devices.

2: This register is not available on 64-pin and 100-pin devices.

NOTES:

18.1 Output Compare Control Registers

TABLE 18-2: OUTPUT COMPARE 1 THROUGH OUTPUT COMPARE 9 REGISTER MAP

Virtual Address (BF84_#)	Register Name(1)	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	
4000	OC1CON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	—	—	—	—	OC32	OCFLT	OCTSEL	OCM<2:0>			0000
4010	OC1R	31:16	OC1R<31:0>																xxxx
		15:0																	xxxx
4020	OC1RS	31:16	OC1RS<31:0>																xxxx
		15:0																	xxxx
4200	OC2CON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	—	—	—	—	OC32	OCFLT	OCTSEL	OCM<2:0>			0000
4210	OC2R	31:16	OC2R<31:0>																xxxx
		15:0																	xxxx
4220	OC2RS	31:16	OC2RS<31:0>																xxxx
		15:0																	xxxx
4400	OC3CON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	—	—	—	—	OC32	OCFLT	OCTSEL	OCM<2:0>			0000
4410	OC3R	31:16	OC3R<31:0>																xxxx
		15:0																	xxxx
4420	OC3RS	31:16	OC3RS<31:0>																xxxx
		15:0																	xxxx
4600	OC4CON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	—	—	—	—	OC32	OCFLT	OCTSEL	OCM<2:0>			0000
4610	OC4R	31:16	OC4R<31:0>																xxxx
		15:0																	xxxx
4620	OC4RS	31:16	OC4RS<31:0>																xxxx
		15:0																	xxxx
4800	OC5CON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	—	—	—	—	OC32	OCFLT	OCTSEL	OCM<2:0>			0000
4810	OC5R	31:16	OC5R<31:0>																xxxx
		15:0																	xxxx
4820	OC5RS	31:16	OC5RS<31:0>																xxxx
		15:0																	xxxx

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

Note 1: All registers in this table have corresponding CLR, SET and INV registers at their virtual addresses, plus offsets of 0x4, 0x8 and 0xC, respectively. See **Section 12.3 "CLR, SET, and INV Registers"** for more information.

REGISTER 18-1: OCxCON: OUTPUT COMPARE ‘x’ 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	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
	ON	—	SIDL	—	—	—	—	—
7:0	U-0	U-0	R/W-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	OC32	OCFLT(1)	OCTSEL(2)	OCM<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 31-16 **Unimplemented:** Read as ‘0’

bit 15 **ON:** Output Compare Peripheral On bit
1 = Output Compare peripheral is enabled
0 = Output Compare peripheral is disabled

bit 14 **Unimplemented:** Read as ‘0’

bit 13 **SIDL:** Stop in Idle Mode bit
1 = Discontinue operation when CPU enters Idle mode
0 = Continue operation in Idle mode

bit 12-6 **Unimplemented:** Read as ‘0’

bit 5 **OC32:** 32-bit Compare Mode bit
1 = OCxR<31:0> and/or OCxRS<31:0> are used for comparisons to the 32-bit timer source
0 = OCxR<15:0> and OCxRS<15:0> are used for comparisons to the 16-bit timer source

bit 4 **OCFLT:** PWM Fault Condition Status bit(1)
1 = PWM Fault condition has occurred (cleared in HW only)
0 = No PWM Fault condition has occurred

bit 3 **OCTSEL:** Output Compare Timer Select bit(2)
1 = Timery is the clock source for this Output Compare module
0 = Timerx is the clock source for this Output Compare module

bit 2-0 **OCM<2:0>:** Output Compare Mode Select bits
111 = PWM mode on OCx; Fault pin is enabled
110 = PWM mode on OCx; Fault pin is disabled
101 = Initialize OCx pin low; generate continuous output pulses on OCx pin
100 = Initialize OCx pin low; generate single output pulse on OCx pin
011 = Compare event toggles OCx pin
010 = Initialize OCx pin high; compare event forces OCx pin low
001 = Initialize OCx pin low; compare event forces OCx pin high
000 = Output compare peripheral is disabled but continues to draw current

Note 1: This bit is only used when OCM<2:0> = ‘111’. It is read as ‘0’ in all other modes.
2: Refer to Table 18-1 for Timerx and Timery selections.

REGISTER 20-10: SQI1TXDATA: SQI TRANSMIT DATA BUFFER 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
	TXDATA<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
	TXDATA<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
	TXDATA<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
	TXDATA<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 **TXDATA<31:0>:** Transmit Command Data bits

Data is loaded into this register before being transmitted. Prior to the data transfer, the data in TXDATA is loaded into the shift register (SFDR).

Multiple writes to TXDATA can occur while a transfer is in progress. There can be a maximum of eight commands that can be queued.

REGISTER 20-11: SQI1RXDATA: SQI RECEIVE DATA BUFFER 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-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	RXDATA<31:24>							
23:16	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	RXDATA<23:16>							
15:8	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	RXDATA<15:8>							
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	RXDATA<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 **RXDATA<31:0>:** Receive Data Buffer bits

At the end of a data transfer, the data in the shift register is loaded into the RXDATA register. This register works like a FIFO. The depth of the receive buffer is eight words.

REGISTER 20-22: SQI1INTSIGEN: SQI INTERRUPT SIGNAL 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	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	DMAEISE	PKT DONEISE	BD DONEISE	CON THRISE
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
	CON EMPTYISE	CON FULLISE	RX THRISE	RX FULLISE	RX EMPTYISE	TX THRISE	TX FULLISE	TX EMPTYISE

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 31-12 **Unimplemented:** Read as ‘0’

bit 11 **DMAEISE:** DMA Bus Error Interrupt Signal Enable bit
1 = Interrupt signal is enabled
0 = Interrupt signal is disabled

bit 10 **PKTDONEISE:** Receive Error Interrupt Signal Enable bit
1 = Interrupt signal is enabled
0 = Interrupt signal is disabled

bit 9 **BDDONEISE:** Transmit Error Interrupt Signal Enable bit
1 = Interrupt signal is enabled
0 = Interrupt signal is disabled

bit 8 **CONTHRISE:** Control Buffer Threshold Interrupt Signal Enable bit
1 = Interrupt signal is enabled
0 = Interrupt signal is disabled

bit 7 **CONEMPTYISE:** Control Buffer Empty Interrupt Signal Enable bit
1 = Interrupt signal is enabled
0 = Interrupt signal is disabled

bit 6 **CONFULLISE:** Control Buffer Full Interrupt Signal Enable bit
1 = Interrupt signal is enabled
0 = Interrupt signal is disabled

bit 5 **RXTHRISE:** Receive Buffer Threshold Interrupt Signal Enable bit
1 = Interrupt signal is enabled
0 = Interrupt signal is disabled

bit 4 **RXFULLISE:** Receive Buffer Full Interrupt Signal Enable bit
1 = Interrupt signal is enabled
0 = Interrupt signal is disabled

bit 3 **RXEMPTYISE:** Receive Buffer Empty Interrupt Signal Enable bit
1 = Interrupt signal is enabled
0 = Interrupt signal is disabled

bit 2 **TXTHRISE:** Transmit Buffer Threshold Interrupt Signal Enable bit
1 = Interrupt signal is enabled
0 = Interrupt signal is disabled

bit 1 **TXFULLISE:** Transmit Buffer Full Interrupt Signal Enable bit
1 = Interrupt signal is enabled
0 = Interrupt signal is disabled

bit 0 **TXEMPTYISE:** Transmit Buffer Empty Interrupt Signal Enable bit
1 = Interrupt signal is enabled
0 = Interrupt signal is disabled

REGISTER 28-1: ADCCON1: ADC CONTROL REGISTER 1 (CONTINUED)

bit 3 **STRGLVL:** Scan Trigger High Level/Positive Edge Sensitivity bit
1 = Scan trigger is high level sensitive. Once STRIG mode is selected (TRGSRCx<4:0> in the ADCTRGx register), the scan trigger will continue for all selected analog inputs, until the STRIG option is removed.
0 = Scan trigger is positive edge sensitive. Once STRIG mode is selected (TRGSRCx<4:0> in the ADCTRGx register), only a single scan trigger will be generated, which will complete the scan of all selected analog inputs.

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

REGISTER 30-33: EMAC1MADR: ETHERNET CONTROLLER MAC MII MANAGEMENT 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	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	R/W-0	R/W-0	R/W-0	R/W-0	R/W-1
	—	—	—	PHYADDR<4:0>				
7:0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	REGADDR<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 31-13 **Unimplemented:** Read as ’0’

bit 12-8 **PHYADDR<4:0>:** MII Management PHY Address bits
This field represents the 5-bit PHY Address field of Management cycles. Up to 31 PHYs can be addressed (0 is reserved).

bit 7-5 **Unimplemented:** Read as ’0’

bit 4-0 **REGADDR<4:0>:** MII Management Register Address bits
This field represents the 5-bit Register Address field of Management cycles. Up to 32 registers can be accessed.

Note: Both 16-bit and 32-bit accesses are allowed to these registers (including the SET, CLR and INV registers). 8-bit accesses are not allowed and are ignored by the hardware.

36.2 MPLAB XC Compilers

The MPLAB XC Compilers are complete ANSI C compilers for all of Microchip's 8, 16, and 32-bit MCU and DSC devices. These compilers provide powerful integration capabilities, superior code optimization and ease of use. MPLAB XC Compilers run on Windows, Linux or MAC OS X.

For easy source level debugging, the compilers provide debug information that is optimized to the MPLAB X IDE.

The free MPLAB XC Compiler editions support all devices and commands, with no time or memory restrictions, and offer sufficient code optimization for most applications.

MPLAB XC Compilers include an assembler, linker and utilities. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. MPLAB XC Compiler uses the assembler to produce its object file. Notable features of the assembler include:

- Support for the entire device instruction set
- Support for fixed-point and floating-point data
- Command-line interface
- Rich directive set
- Flexible macro language
- MPLAB X IDE compatibility

36.3 MPASM Assembler

The MPASM Assembler is a full-featured, universal macro assembler for PIC10/12/16/18 MCUs.

The MPASM Assembler generates relocatable object files for the MPLINK Object Linker, Intel® standard HEX files, MAP files to detail memory usage and symbol reference, absolute LST files that contain source lines and generated machine code, and COFF files for debugging.

The MPASM Assembler features include:

- Integration into MPLAB X IDE projects
- User-defined macros to streamline assembly code
- Conditional assembly for multipurpose source files
- Directives that allow complete control over the assembly process

36.4 MPLINK Object Linker/ MPLIB Object Librarian

The MPLINK Object Linker combines relocatable objects created by the MPASM Assembler. It can link relocatable objects from precompiled libraries, using directives from a linker script.

The MPLIB Object Librarian manages the creation and modification of library files of precompiled code. When a routine from a library is called from a source file, only the modules that contain that routine will be linked in with the application. This allows large libraries to be used efficiently in many different applications.

The object linker/library features include:

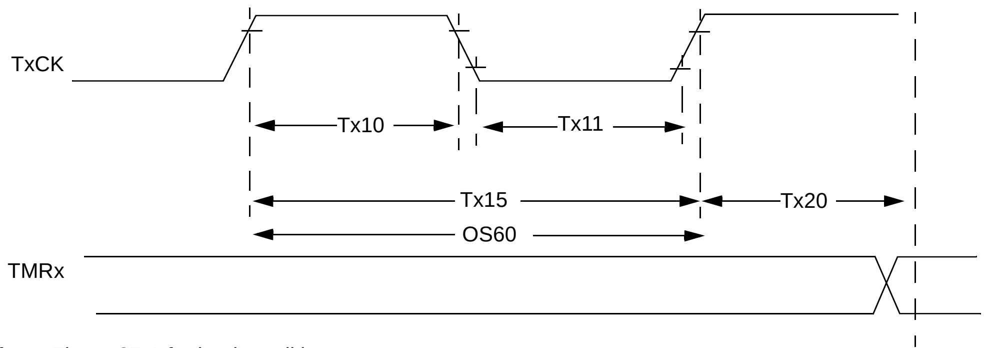
- Efficient linking of single libraries instead of many smaller files
- Enhanced code maintainability by grouping related modules together
- Flexible creation of libraries with easy module listing, replacement, deletion and extraction

36.5 MPLAB Assembler, Linker and Librarian for Various Device Families

MPLAB Assembler produces relocatable machine code from symbolic assembly language for PIC24, PIC32 and dsPIC DSC devices. MPLAB XC Compiler uses the assembler to produce its object file. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. Notable features of the assembler include:

- Support for the entire device instruction set
- Support for fixed-point and floating-point data
- Command-line interface
- Rich directive set
- Flexible macro language
- MPLAB X IDE compatibility

FIGURE 37-6: TIMER1-TIMER9 EXTERNAL CLOCK TIMING CHARACTERISTICS



Note: Refer to Figure 37-1 for load conditions.

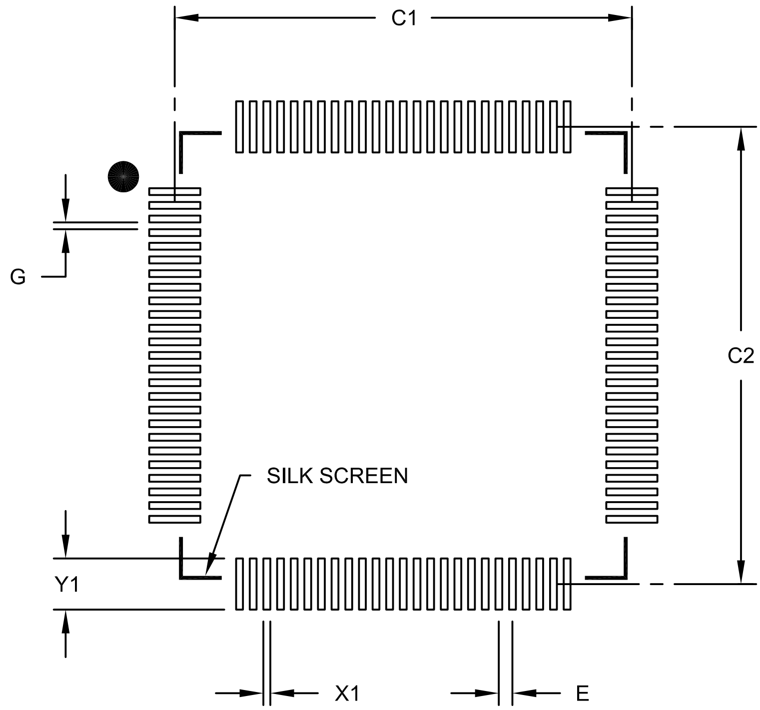
TABLE 37-25: TIMER1 EXTERNAL CLOCK TIMING REQUIREMENTS(1)

AC CHARACTERISTICS				Standard Operating Conditions: 2.1V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param. No.	**Symbol**	**Characteristics(2)**		**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**
TA10	TTXH	TxCK High Time	Synchronous, with prescaler	[(12.5 ns or 1 TPBCLK3) /N] + 20 ns	—	—	ns	Must also meet parameter TA15 (**Note 3**)
			Asynchronous, with prescaler	10	—	—	ns	—
TA11	TTXL	TxCK Low Time	Synchronous, with prescaler	[(12.5 ns or 1 TPBCLK3) /N] + 20 ns	—	—	ns	Must also meet parameter TA15 (**Note 3**)
			Asynchronous, with prescaler	10	—	—	ns	—
TA15	TTXP	TxCK Input Period	Synchronous, with prescaler	[(Greater of 20 ns or 2 TPBCLK3)/N] + 30 ns	—	—	ns	VDD > 2.7V (**Note 3**)
				[(Greater of 20 ns or 2 TPBCLK3)/N] + 50 ns	—	—	ns	VDD < 2.7V (**Note 3**)
			Asynchronous, with prescaler	20	—	—	ns	VDD > 2.7V
				50	—	—	ns	VDD < 2.7V
OS60	FT1	SOSC1/T1CK Oscillator Input Frequency Range (oscillator enabled by setting TCS bit (T1CON<1>))		32	—	50	kHz	—
TA20	TCKEXTMRL	Delay from External TxCK Clock Edge to Timer Increment		—		1	TPBCLK3	—

Note 1: Timer1 is a Type A timer.
2: This parameter is characterized, but not tested in manufacturing.
3: N = Prescale Value (1, 8, 64, 256).

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



RECOMMENDED LAND PATTERN

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

TABLE A-3: ADC DIFFERENCES (CONTINUED)

PIC32MX5XX/6XX/7XX Feature	PIC32MZ EF Feature
ADC Calibration	
On PIC32MX devices, the ADC module can be used immediately, once it is enabled.	PIC32MZ devices require a calibration step prior to operation. This is done by copying the calibration data from DEVADCx to the corresponding ADCxCFG register.
I/O Pin Analog Function Selection	
On PIC32MX devices, the analog function of an I/O pin was determined by the PCFGx bit in the AD1PCFG register. PCFGx (AD1PCFG<x>) 1 = Analog input pin in Digital mode 0 = Analog input pin in Analog mode	On PIC32MZ EF devices, the analog selection function has been moved into a separate register on each I/O port. Note that the sense of the bit is different. ANSxy (ANSELx<y>) 1 = Analog input pin in Analog mode 0 = Analog input pin in Digital mode
Electrical Specifications and Timing Requirements	
Refer to **"Section 31. Electrical Characteristics"** in the PIC32MX5XX/6XX/7XX Data Sheet for ADC module specifications and timing requirements.	On PIC32MZ EF devices, the ADC module sampling and conversion time and other specifications have changed. Refer to **37.0 "Electrical Characteristics"** for more information.

TABLE A-9: FLASH PROGRAMMING DIFFERENCES (CONTINUED)

PIC32MX5XX/6XX/7XX Feature	PIC32MZ EF Feature
Flash Programming	
NVMOP<3:0> (NVMCON<3:0>) `1111` = Reserved • • • **`0111` = Reserved** **`0110` = No operation** **`0101` = Program Flash (PFM) erase operation** `0100` = Page erase operation `0011` = Row program operation **`0010` = No operation** `0001` = Word program operation `0000` = No operation	The op codes for programming the Flash memory have been changed to accommodate the new quad-word programming and dual-panel features. The row size has changed to 2 KB (512 IW) from 128 IW. The page size has changed to 16 KB (4K IW) from 4 KB (1K IW). Note that the NVMOP register is now protected, and requires the WREN bit be set to enable modification. NVMOP<3:0> (NVMCON<3:0>) `1111` = Reserved • • • `1000` = Reserved **`0111` = Program erase operation** **`0110` = Upper program Flash memory erase operation** **`0101` = Lower program Flash memory erase operation** `0100` = Page erase operation `0011` = Row program operation **`0010` = Quad Word (128-bit) program operation** `0001` = Word program operation `0000` = No operation
PIC32MX devices feature a single NVMDATA register for word programming. NVMDATA	On PIC32MZ EF devices, to support quad word programming, the NVMDATA register has been expanded to four words. NVMDATA**x**, where 'x' = 0 through 3
Flash Endurance and Retention	
PIC32MX devices support Flash endurance and retention of up to 20K E/W cycles and 20 years.	On PIC32MZ EF devices, ECC must be enabled to support the same endurance and retention as PIC32MX devices.
Configuration Words	
On PIC32MX devices, Configuration Words can be programmed with **Word or Row program** operation.	On PIC32MZ EF devices, all Configuration Words must be programmed with **Quad Word or Row Program** operations.
Configuration Words Reserved Bit	
On PIC32MX devices, the **DEVCFG0<15>** bit is Reserved and must be programmed to '`0`'.	On PIC32MZ EF devices, this bit is **DEVSIGN0<31>**.