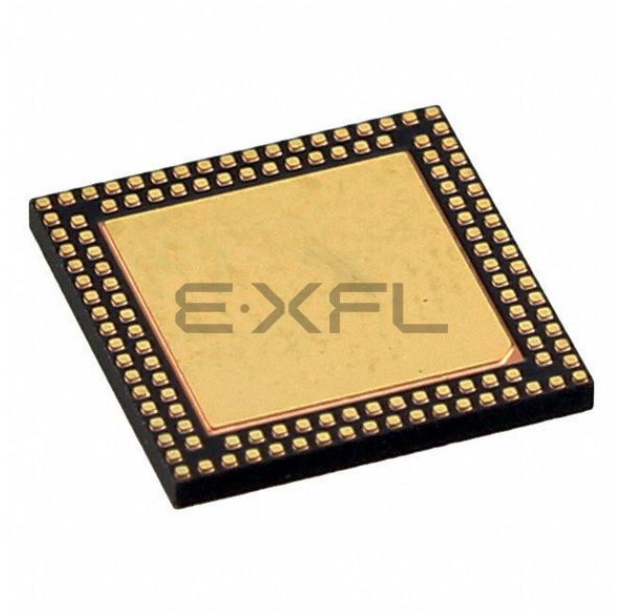


Welcome to E-XFL.COM

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	200MHz
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	1MB (1M x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256K x 8
Voltage - Supply (Vcc/Vdd)	2.1V ~ 3.6V
Data Converters	A/D 48x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
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/pic32mz1024eff124-i-tl

2.0 GUIDELINES FOR GETTING STARTED WITH 32-BIT MICROCONTROLLERS

Note 1: This data sheet summarizes the features of the PIC32MZ EF family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to the *"PIC32 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com/PIC32).

2.1 Basic Connection Requirements

Getting started with the PIC32MZ EF family of 32-bit Microcontrollers (MCUs) requires attention to a minimal set of device pin connections before proceeding with development. The following is a list of pin names, which must always be connected:

- All VDD and VSS pins (see **2.2 "Decoupling Capacitors"**)
- All AVDD and AVSS pins, even if the ADC module is not used (see **2.2 "Decoupling Capacitors"**)
- $\overline{\text{MCLR}}$ pin (see **2.3 "Master Clear (MCLR) Pin"**)
- PGECx/PGEDx pins, used for In-Circuit Serial Programming™ (ICSP™) and debugging purposes (see **2.4 "ICSP Pins"**)
- OSC1 and OSC2 pins, when external oscillator source is used (see **2.7 "External Oscillator Pins"**)

The following pin(s) may be required as well:

VREF+/VREF- pins, used when external voltage reference for the ADC module is implemented.

Note: The AVDD and AVSS pins must be connected, regardless of ADC use and the ADC voltage reference source.

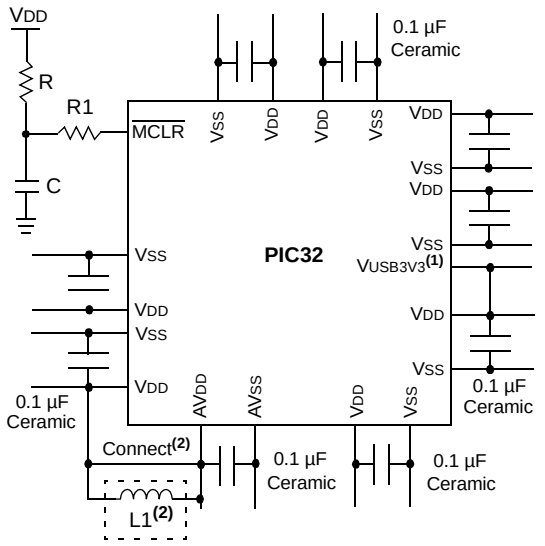
2.2 Decoupling Capacitors

The use of decoupling capacitors on power supply pins, such as VDD, VSS, AVDD and AVSS is required. See Figure 2-1.

Consider the following criteria when using decoupling capacitors:

- **Value and type of capacitor:** A value of 0.1 µF (100 nF), 10-20V is recommended. The capacitor should be a low Equivalent Series Resistance (low-ESR) capacitor and have resonance frequency in the range of 20 MHz and higher. It is further recommended that ceramic capacitors be used.
- **Placement on the printed circuit board:** The decoupling capacitors should be placed as close to the pins as possible. It is recommended that the capacitors be placed on the same side of the board as the device. If space is constricted, the capacitor can be placed on another layer on the PCB using a via; however, ensure that the trace length from the pin to the capacitor is within one-quarter inch (6 mm) in length.
- **Handling high frequency noise:** If the board is experiencing high frequency noise, upward of tens of MHz, add a second ceramic-type capacitor in parallel to the above described decoupling capacitor. The value of the second capacitor can be in the range of 0.01 µF to 0.001 µF. Place this second capacitor next to the primary decoupling capacitor. In high-speed circuit designs, consider implementing a decade pair of capacitances as close to the power and ground pins as possible. For example, 0.1 µF in parallel with 0.001 µF.
- **Maximizing performance:** On the board layout from the power supply circuit, run the power and return traces to the decoupling capacitors first, and then to the device pins. This ensures that the decoupling capacitors are first in the power chain. Equally important is to keep the trace length between the capacitor and the power pins to a minimum thereby reducing PCB track inductance.

FIGURE 2-1: RECOMMENDED MINIMUM CONNECTION



Note 1: If the USB module is not used, this pin must be connected to VSS.

2: As an option, instead of a hard-wired connection, an inductor (L1) can be substituted between VDD and AVDD to improve ADC noise rejection. The inductor impedance should be less than 1Ω and the inductor capacity greater than 10 mA.

Where:

$$f = \frac{FCNV}{2} \quad \text{(i.e., ADC conversion rate/2)}$$

$$f = \frac{1}{(2\pi\sqrt{LC})}$$

$$L = \left(\frac{1}{(2\pi f\sqrt{C})}\right)^2$$

2.2.1 BULK CAPACITORS

The use of a bulk capacitor is recommended to improve power supply stability. Typical values range from 4.7 µF to 47 µF. This capacitor should be located as close to the device as possible.

2.3 Master Clear (MCLR) Pin

The MCLR pin provides for two specific device functions:

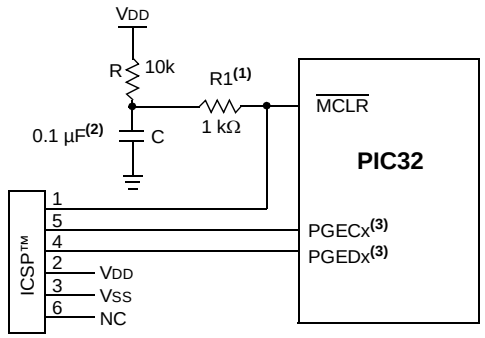
- Device Reset
- Device programming and debugging

Pulling The MCLR pin low generates either a device Reset or a POR, depending on the setting of the SMCLR bit (DEVCFG0<15>). Figure 2-2 illustrates a typical MCLR circuit. During device programming and debugging, the resistance and capacitance that can be added to the pin must be considered. Device programmers and debuggers drive the MCLR pin. Consequently, specific voltage levels (VIH and VIL) and fast signal transitions must not be adversely affected. Therefore, specific values of R and C will need to be adjusted based on the application and PCB requirements.

For example, as illustrated in Figure 2-2, it is recommended that the capacitor C be isolated from the MCLR pin during programming and debugging operations.

Place the components illustrated in Figure 2-2 within one-quarter inch (6 mm) from the MCLR pin.

FIGURE 2-2: EXAMPLE OF MCLR PIN CONNECTIONS



Note 1: 470Ω ≤ R1 ≤ 1kΩ will limit any current flowing into MCLR from the external capacitor C, in the event of MCLR pin breakdown, due to Electrostatic Discharge (ESD) or Electrical Overstress (EOS). Ensure that the MCLR pin VIH and VIL specifications are met without interfering with the Debug/Programmer tools.

2: The capacitor can be sized to prevent unintentional Resets from brief glitches or to extend the device Reset period during POR.

3: No pull-ups or bypass capacitors are allowed on active debug/program PGECx/PGEDx pins.

A block diagram of the PIC32MZ EF family processor core is shown in Figure 3-1.

FIGURE 3-1: PIC32MZ EF FAMILY MICROPROCESSOR CORE BLOCK DIAGRAM

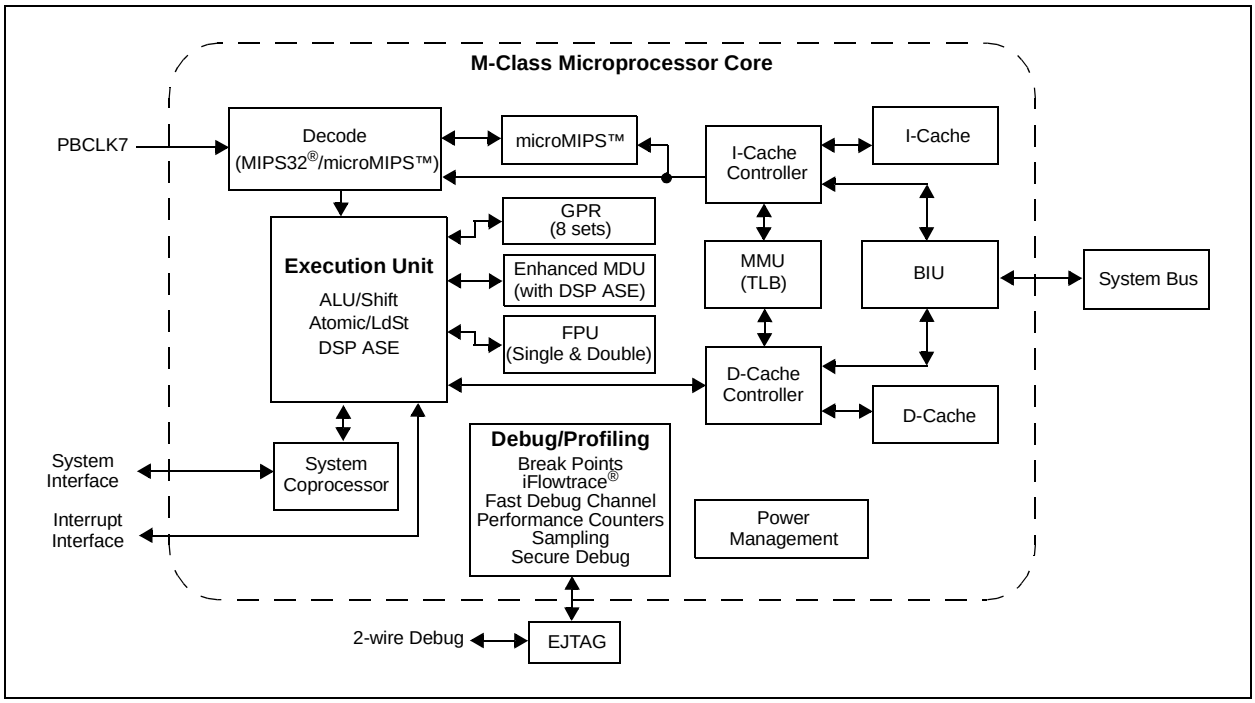


TABLE 7-2: INTERRUPT IRQ, VECTOR, AND BIT LOCATION (CONTINUED)

Interrupt Source(1)	XC32 Vector Name	IRQ #	Vector #	Interrupt Bit Location				Persistent Interrupt
				Flag	Enable	Priority	Sub-priority	
ADC Digital Comparator 5	_ADC_DC5_VECTOR	50	OFF050<17:1>	IFS1<18>	IEC1<18>	IPC12<20:18>	IPC12<17:16>	Yes
ADC Digital Comparator 6	_ADC_DC6_VECTOR	51	OFF051<17:1>	IFS1<19>	IEC1<19>	IPC12<28:26>	IPC12<25:24>	Yes
ADC Digital Filter 1	_ADC_DF1_VECTOR	52	OFF052<17:1>	IFS1<20>	IEC1<20>	IPC13<4:2>	IPC13<1:0>	Yes
ADC Digital Filter 2	_ADC_DF2_VECTOR	53	OFF053<17:1>	IFS1<21>	IEC1<21>	IPC13<12:10>	IPC13<9:8>	Yes
ADC Digital Filter 3	_ADC_DF3_VECTOR	54	OFF054<17:1>	IFS1<22>	IEC1<22>	IPC13<20:18>	IPC13<17:16>	Yes
ADC Digital Filter 4	_ADC_DF4_VECTOR	55	OFF055<17:1>	IFS1<23>	IEC1<23>	IPC13<28:26>	IPC13<25:24>	Yes
ADC Digital Filter 5	_ADC_DF5_VECTOR	56	OFF056<17:1>	IFS1<24>	IEC1<24>	IPC14<4:2>	IPC14<1:0>	Yes
ADC Digital Filter 6	_ADC_DF6_VECTOR	57	OFF057<17:1>	IFS1<25>	IEC1<25>	IPC14<12:10>	IPC14<9:8>	Yes
ADC Fault	_ADC_FAULT_VECTOR	58	OFF058<17:1>	IFS1<26>	IEC1<26>	IPC14<20:18>	IPC14<17:16>	No
ADC Data 0	_ADC_DATA0_VECTOR	59	OFF059<17:1>	IFS1<27>	IEC1<27>	IPC14<28:26>	IPC14<25:24>	Yes
ADC Data 1	_ADC_DATA1_VECTOR	60	OFF060<17:1>	IFS1<28>	IEC1<28>	IPC15<4:2>	IPC15<1:0>	Yes
ADC Data 2	_ADC_DATA2_VECTOR	61	OFF061<17:1>	IFS1<29>	IEC1<29>	IPC15<12:10>	IPC15<9:8>	Yes
ADC Data 3	_ADC_DATA3_VECTOR	62	OFF062<17:1>	IFS1<30>	IEC1<30>	IPC15<20:18>	IPC15<17:16>	Yes
ADC Data 4	_ADC_DATA4_VECTOR	63	OFF063<17:1>	IFS1<31>	IEC1<31>	IPC15<28:26>	IPC15<25:24>	Yes
ADC Data 5	_ADC_DATA5_VECTOR	64	OFF064<17:1>	IFS2<0>	IEC2<0>	IPC16<4:2>	IPC16<1:0>	Yes
ADC Data 6	_ADC_DATA6_VECTOR	65	OFF065<17:1>	IFS2<1>	IEC2<1>	IPC16<12:10>	IPC16<9:8>	Yes
ADC Data 7	_ADC_DATA7_VECTOR	66	OFF066<17:1>	IFS2<2>	IEC2<2>	IPC16<20:18>	IPC16<17:16>	Yes
ADC Data 8	_ADC_DATA8_VECTOR	67	OFF067<17:1>	IFS2<3>	IEC2<3>	IPC16<28:26>	IPC16<25:24>	Yes
ADC Data 9	_ADC_DATA9_VECTOR	68	OFF068<17:1>	IFS2<4>	IEC2<4>	IPC17<4:2>	IPC17<1:0>	Yes
ADC Data 10	_ADC_DATA10_VECTOR	69	OFF069<17:1>	IFS2<5>	IEC2<5>	IPC17<12:10>	IPC17<9:8>	Yes
ADC Data 11	_ADC_DATA11_VECTOR	70	OFF070<17:1>	IFS2<6>	IEC2<6>	IPC17<20:18>	IPC17<17:16>	Yes
ADC Data 12	_ADC_DATA12_VECTOR	71	OFF071<17:1>	IFS2<7>	IEC2<7>	IPC17<28:26>	IPC17<25:24>	Yes
ADC Data 13	_ADC_DATA13_VECTOR	72	OFF072<17:1>	IFS2<8>	IEC2<8>	IPC18<4:2>	IPC18<1:0>	Yes
ADC Data 14	_ADC_DATA14_VECTOR	73	OFF073<17:1>	IFS2<9>	IEC2<9>	IPC18<12:10>	IPC18<9:8>	Yes
ADC Data 15	_ADC_DATA15_VECTOR	74	OFF074<17:1>	IFS2<10>	IEC2<10>	IPC18<20:18>	IPC18<17:16>	Yes
ADC Data 16	_ADC_DATA16_VECTOR	75	OFF075<17:1>	IFS2<11>	IEC2<11>	IPC18<28:26>	IPC18<25:24>	Yes
ADC Data 17	_ADC_DATA17_VECTOR	76	OFF076<17:1>	IFS2<12>	IEC2<12>	IPC19<4:2>	IPC19<1:0>	Yes
ADC Data 18	_ADC_DATA18_VECTOR	77	OFF077<17:1>	IFS2<13>	IEC2<13>	IPC19<12:10>	IPC19<9:8>	Yes

Note 1: Not all interrupt sources are available on all devices. See **TABLE 1: "PIC32MZ EF Family Features"** for the list of available peripherals.

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

3: This interrupt source is not available on 100-pin devices.

4: This interrupt source is not available on 124-pin devices.

REGISTER 11-1: USBCSR0: USB CONTROL STATUS REGISTER 0

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-0, HS	R-0, HS	R-0, HS	R-0, HS	R-0, HS	R-0, HS	R-0, HS	R-0, HS
	EP7TXIF	EP6TXIF	EP5TXIF	EP4TXIF	EP3TXIF	EP2TXIF	EP1TXIF	EP0IF
15:8	R/W-0	R/W-0	R/W-1	R-0, HS	R-0	R/W-0	R-0, HC	R/W-0
	ISOUPD	SOFTCONN	HSEN	HSMODE	RESET	RESUME	SUSPMODE	SUSPEN
	—	—						
7:0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	FUNC<6:0>						
		—	—	—	—	—	—	—

Legend:	HS = Hardware Set	HC = Hardware 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	x = Bit is unknown

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

bit 23-17 **EP7TXIF:EP1TXIF:** Endpoint 'n' TX Interrupt Flag bit
1 = Endpoint has a transmit interrupt to be serviced
0 = No interrupt event

bit 16 **EP0IF:** Endpoint 0 Interrupt bit
1 = Endpoint 0 has an interrupt to be serviced
0 = No interrupt event

All EPxTX and EP0 bits are cleared when the byte is read. Therefore, these bits must be read independently from the remaining bits in this register to avoid accidental clearing.

bit 15 **ISOUPD:** ISO Update bit (*Device mode only; unimplemented in Host mode*)
1 = USB module will wait for a SOF token from the time TXPKTRDY is set before sending the packet
0 = No change in behavior

This bit only affects endpoints performing isochronous transfers when in *Device mode*. This bit is unimplemented in *Host mode*.

bit 14 **SOFTCONN:** Soft Connect/Disconnect Feature Selection bit
1 = The USB D+/D- lines are enabled and active
0 = The USB D+/D- lines are disabled and are tri-stated

This bit is only available in *Device mode*.

bit 13 **HSEN:** Hi-Speed Enable bit
1 = The USB module will negotiate for Hi-Speed mode when the device is reset by the hub
0 = Module only operates in Full-Speed mode

bit 12 **HSMODE:** Hi-Speed Mode Status bit
1 = Hi-Speed mode successfully negotiated during USB reset
0 = Module is not in Hi-Speed mode

In *Device mode*, this bit becomes valid when a USB reset completes. In *Host mode*, it becomes valid when the RESET bit is cleared.

bit 11 **RESET:** Module Reset Status bit
1 = Reset signaling is present on the bus
0 = Normal module operation

In *Device mode*, this bit is read-only. In *Host mode*, this bit is read/write.

REGISTER 11-9: USBIENCSR1: USB INDEXED ENDPOINT CONTROL STATUS REGISTER 1 (ENDPOINT 1-7) (CONTINUED)

bit 18 **OVERRUN:** Data Overrun Status bit (*Device mode*)

1 = An OUT packet cannot be loaded into the RX FIFO.
0 = Written by software to clear this bit

This bit is only valid when the endpoint is operating in ISO mode. In Bulk mode, it always returns zero.

ERROR: No Data Packet Received Status bit (*Host mode*)

1 = Three attempts have been made to receive a packet and no data packet has been received. An interrupt is generated.

0 = Written by the software to clear this bit.

This bit is only valid when the RX endpoint is operating in Bulk or Interrupt mode. In ISO mode, it always returns zero.

bit 17 **FIFOFULL:** FIFO Full Status bit

1 = No more packets can be loaded into the RX FIFO
0 = The RX FIFO has at least one free space

bit 16 **RXPKTRDY:** Data Packet Reception Status bit

1 = A data packet has been received. An interrupt is generated.
0 = Written by software to clear this bit when the packet has been unloaded from the RX FIFO.

bit 15-11 **MULT<4:0>:** Multiplier Control bits

For Isochronous/Interrupt endpoints or of packet splitting on Bulk endpoints, multiplies TXMAXP by MULT+1 for the payload size.

For Bulk endpoints, MULT can be up to 32 and defines the number of “USB” packets of the specified payload into which a single data packet placed in the FIFO should be split, prior to transfer. The data packet is required to be an exact multiple of the payload specified by TXMAXP.

For Isochronous/Interrupts endpoints operating in Hi-Speed mode, MULT may be either 2 or 3 and specifies the maximum number of such transactions that can take place in a single microframe.

bit 10-0 **RXMAXP<10:0>:** Maximum RX Payload Per Transaction Control bits

This field sets the maximum payload (in bytes) transmitted in a single transaction. The value is subject to the constraints placed by the USB Specification on packet sizes for Bulk, Interrupt and Isochronous transfers in Full-Speed and Hi-Speed operations.

RXMAXP must be set to an even number of bytes for proper interrupt generation in DMA Mode 1.

REGISTER 11-30: USBCRCON: USB CLOCK/RESET 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	R-0, HS, HC	R-0, HS, HC	R/W-1, HS
	—	—	—	—	—	USBIF	USBRF	USBWKUP
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	r-1	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
	—	—	—	—	—	—	USB IDOVEN	USB IDVAL
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
	PHYIDEN	VBUS MONEN	ASVAL MONEN	BSVAL MONEN	SEND MONEN	USBIE	USBRIE	USB WKUPEN

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 31-27 **Unimplemented:** Read as '0'

bit 26 **USBIF:** USB General Interrupt Flag bit
1 = An event on the USB Bus has occurred
0 = No interrupt from USB module or interrupts have not been enabled

bit 25 **USBRF:** USB Resume Flag bit
1 = Resume from Suspend state. Device wake-up activity can be started.
0 = No Resume activity detected during Suspend, or not in Suspend state

bit 24 **USBWK:** USB Activity Status bit
1 = Connect, disconnect, or other activity on USB detected since last cleared
0 = No activity detected on USB

Note: This bit should be cleared just prior to entering sleep, but it should be checked that no activity has already occurred on USB before actually entering sleep.

bit 23-14 **Unimplemented:** Read as '0'

bit 15 **Reserved:** Read as '1'

bit 14-10 **Unimplemented:** Read as '0'

bit 9 **USBIDOVEN:** USB ID Override Enable bit
1 = Enable use of USBIDVAL bit
0 = Disable use of USBIDVAL and instead use the PHY value

bit 8 **USBIDVAL:** USB ID Value bit
1 = ID override value is 1
0 = ID override value is 0

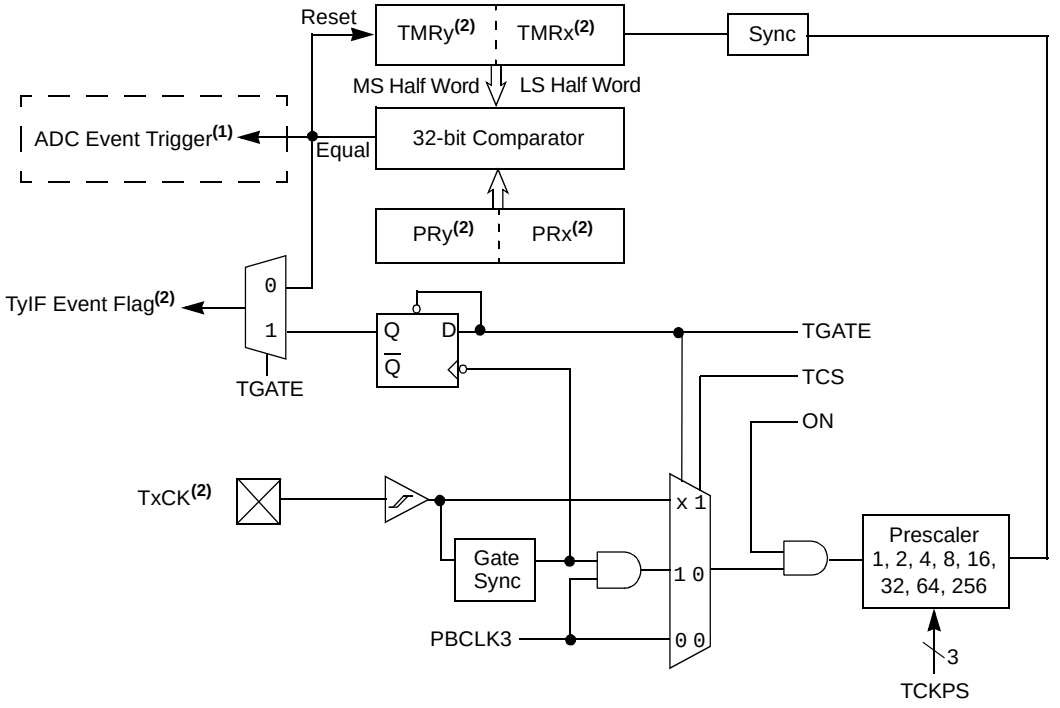
bit 7 **PHYIDEN:** PHY ID Monitoring Enable bit
1 = Enable monitoring of the ID bit from the USB PHY
0 = Disable monitoring of the ID bit from the USB PHY

bit 6 **VBUSMONEN:** VBUS Monitoring for OTG Enable bit
1 = Enable monitoring for VBUS in VBUS Valid range (between 4.4V and 4.75V)
0 = Disable monitoring for VBUS in VBUS Valid range

bit 5 **ASVALMONEN:** A-Device VBUS Monitoring for OTG Enable bit
1 = Enable monitoring for VBUS in Session Valid range for A-device (between 0.8V and 2.0V)
0 = Disable monitoring for VBUS in Session Valid range for A-device

bit 4 **BSVALMONEN:** B-Device VBUS Monitoring for OTG Enable bit
1 = Enable monitoring for VBUS in Session Valid range for B-device (between 0.8V and 4.0V)
0 = Disable monitoring for VBUS in Session Valid range for B-device

FIGURE 14-2: TIMER2/3, TIMER4/5, TIMER6/7, AND TIMER8/9 BLOCK DIAGRAM (32-BIT)



Note 1: ADC event trigger is available only on the Timer2/3 and TImer4/5 pairs.
2: In this diagram, 'x' represents Timer2, 4, 6, or 8, and 'y' represents Timer3, 5, 7, or 9.

REGISTER 15-3: DMTCLR: DEADMAN TIMER CLEAR 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
	STEP2<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 31-8 **Unimplemented:** Read as '0'

bit 7-0 **STEP2<7:0>:** Clear Timer bits

00001000 = Clears STEP1<7:0>, STEP2<7:0> and the Deadman Timer if, and only if, preceded by correct loading of STEP1<7:0> bits in the correct sequence. The write to these bits may be verified by reading DMTCNT and observing the counter being reset.

All other write patterns = Set BAD2 bit, the value of STEP1<7:0> will remain unchanged, and the new value being written STEP2<7:0> will be captured. These bits are also cleared when a DMT reset event occurs.

REGISTER 20-3: SQI1CFG: SQI CONFIGURATION 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	R/W-0	R/W-0
	—	—	—	—	—	—	CSEN<1:0>	
23:16	R/W-0	U-0	R/W-0	R/W-0	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC
	SQIEN	—	DATAEN<1:0>		CON FIFORST	RX FIFORST	TX FIFORST	RESET
15:8	U-0	r-0	r-0	R/W-0	r-0	R/W-0	R/W-0	U-0
	—	—	—	BURSTEN(1)	—	HOLD	WP	—
7:0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	LSBF	CPOL	CPHA	MODE<2:0>		

Legend:	HC = Hardware Cleared	r = Reserved	
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 31-26 **Unimplemented:** Read as '0'

bit 25-24 **CSEN<1:0>:** Chip Select Output Enable bits
11 = Chip Select 0 and Chip Select 1 are used
10 = Chip Select 1 is used (Chip Select 0 is not used)
01 = Chip Select 0 is used (Chip Select 1 is not used)
00 = Chip Select 0 and Chip Select 1 are not used

bit 23 **SQIEN:** SQI Enable bit
1 = SQI module is enabled
0 = SQI module is disabled

bit 22 **Unimplemented:** Read as '0'

bit 21-20 **DATAEN<1:0>:** Data Output Enable bits
11 = Reserved
10 = SQID3-SQID0 outputs are enabled
01 = SQID1 and SQID0 data outputs are enabled
00 = SQID0 data output is enabled

bit 19 **CONFIFORST:** Control FIFO Reset bit
1 = A reset pulse is generated clearing the control FIFO
0 = A reset pulse is not generated

bit 18 **RXFIFORST:** Receive FIFO Reset bit
1 = A reset pulse is generated clearing the receive FIFO
0 = A reset pulse is not generated

bit 17 **TXFIFORST:** Transmit FIFO Reset bit
1 = A reset pulse is generated clearing the transmit FIFO
0 = A reset pulse is not generated

bit 16 **RESET:** Software Reset Select bit
This bit is automatically cleared by the SQI module. All of the internal state machines and FIFO pointers are reset by this reset pulse.
1 = A reset pulse is generated
0 = A reset pulse is not generated

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

bit 14-13 **Reserved:** Must be programmed as '0'

Note 1: This bit must be programmed as '1'.

REGISTER 20-14: SQI1BDCON: SQI BUFFER DESCRIPTOR 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	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/W-0	R/W-0	R/W-0
	—	—	—	—	—	START	POLLEN	DMAEN

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 **START:** Buffer Descriptor Processor Start bit
- 1 = Start the buffer descriptor processor
- 0 = Disable the buffer descriptor processor

bit 1 **POLLEN:** Buffer Descriptor Poll Enable bit
- 1 = BDP poll is enabled
- 0 = BDP poll is not enabled

bit 0 **DMAEN:** DMA Enable bit
- 1 = DMA is enabled
- 0 = DMA is disabled

REGISTER 20-15: SQI1BDCURADD: SQI BUFFER DESCRIPTOR CURRENT 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-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	BDCURRADDR<31:24>							
23:16	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	BDCURRADDR<23:16>							
15:8	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	BDCURRADDR<15:8>							
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	BDCURRADDR<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 **BDCURRADDR<31:0>:** Current Buffer Descriptor Address bits

These bits contain the address of the current descriptor being processed by the Buffer Descriptor Processor.

REGISTER 24-3: EBISMTx: EXTERNAL BUS INTERFACE STATIC MEMORY TIMING REGISTER ('x' = 0-2)

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	R/W-1	R/W-0	R/W-0
	—	—	—	—	—	RDYMODE	PAGESIZE<1:0>	
23:16	R/W-0	R/W-0	R/W-0	R/W-1	R/W-1	R/W-1	R/W-0	R/W-0
	PAGEMODE	TPRC<3:0>(1)				TBTA<2:0>(1)		
15:8	R/W-0	R/W-0	R/W-1	R/W-0	R/W-1	R/W-1	R/W-0	R/W-1
	TWP<5:0>(1)						TWR<1:0>(1)	
7:0	R/W-0	R/W-1	R/W-0	R/W-0	R/W-1	R/W-0	R/W-1	R/W-1
	TAS<1:0>(1)		TRC<5:0>(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 31-27 **Unimplemented:** Read as '0'

bit 26 **RDYMODE:** Data Ready Device Select bit
The device associated with register set 'x' is a data-ready device, and will use the EBIRDYx pin.
1 = EBIRDYx input is used
0 = EBIRDYx input is not used

bit 25-24 **PAGESIZE<1:0>:** Page Size for Page Mode Device bits
11 = 32-word page
10 = 16-word page
01 = 8-word page
00 = 4-word page

bit 23 **PAGEMODE:** Memory Device Page Mode Support bit
1 = Device supports Page mode
0 = Device does not support Page mode

bit 22-19 **TPRC<3:0>:** Page Mode Read Cycle Time bits(1)
Read cycle time is TPRC + 1 clock cycle.

bit 18-16 **TBTA<2:0>:** Data Bus Turnaround Time bits(1)
Clock cycles (0-7) for static memory between read-to-write, write-to-read, and read-to-read when Chip Select changes.

bit 15-10 **TWP<5:0>:** Write Pulse Width bits(1)
Write pulse width is TWP + 1 clock cycle.

bit 9-8 **TWR<1:0>:** Write Address/Data Hold Time bits(1)
Number of clock cycles to hold address or data on the bus.

bit 7-6 **TAS<1:0>:** Write Address Setup Time bits(1)
Clock cycles for address setup time. A value of '0' is only valid in the case of SSRAM.

bit 5-0 **TRC<5:0>:** Read Cycle Time bits(1)
Read cycle time is TRC + 1 clock cycle.

Note 1: Refer to the **Section 47. "External Bus Interface (EBI)"** in the *"PIC32 Family Reference Manual"* for the EBI timing diagrams and additional information.

REGISTER 27-3: RNGPOLYx: RANDOM NUMBER GENERATOR POLYNOMIAL REGISTER 'x' ('x' = 1 OR 2)

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-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
	POLY<31:24>							
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
	POLY<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
	POLY<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
	POLY<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 **POLY<31:0>:** PRNG LFSR Polynomial MSb/LSb bits (RNGPOLY1 = LSb, RNGPOLY2 = MSb)

REGISTER 27-4: RNGNUMGENx: RANDOM NUMBER GENERATOR REGISTER 'x' ('x' = 1 OR 2)

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-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
	RNG<31:24>							
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
	RNG<23:16>							
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
	RNG<15:8>							
7:0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
	RNG<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 **RNG<31:0>:** Current PRNG MSb/LSb Value bits (RNGNUMGEN1 = LSb, RNGNUMGEN2 = MSb)

REGISTER 30-9: ETHPMCS: ETHERNET CONTROLLER PATTERN MATCH CHECKSUM 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
	PMCS<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
	PMCS<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 15-8 **PMCS<15:8>:** Pattern Match Checksum 1 bits

bit 7-0 **PMCS<7:0>:** Pattern Match Checksum 0 bits

Note 1: This register is only used for RX operations.

2: The bits in this register may only be changed while the RXEN bit (ETHCON1<8>) = 0 or the PMMODE bit (ETHRXFC<11:8>) = 0.

REGISTER 30-10: ETHPMO: ETHERNET CONTROLLER PATTERN MATCH OFFSET 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
	PMO<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
	PMO<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 **PMO<15:0>:** Pattern Match Offset 1 bits

Note 1: This register is only used for RX operations.

2: The bits in this register may only be changed while the RXEN bit (ETHCON1<8>) = 0 or the PMMODE bit (ETHRXFC<11:8>) = 0.

REGISTER 30-11: ETHRXFC: ETHERNET CONTROLLER RECEIVE FILTER CONFIGURATION REGISTER (CONTINUED)

bit 7 **CRCERREN:** CRC Error Collection Enable bit
1 = The received packet CRC must be invalid for the packet to be accepted
0 = Disable CRC Error Collection filtering
This bit allows the user to collect all packets that have an invalid CRC.

bit 6 **CRCOKEN:** CRC OK Enable bit
1 = The received packet CRC must be valid for the packet to be accepted
0 = Disable CRC filtering
This bit allows the user to reject all packets that have an invalid CRC.

bit 5 **RUNTERREN:** Runt Error Collection Enable bit
1 = The received packet must be a runt packet for the packet to be accepted
0 = Disable Runt Error Collection filtering

This bit allows the user to collect all packets that are runt packets. For this filter, a runt packet is defined as any packet with a size of less than 64 bytes (when CRCOKEN = 0) or any packet with a size of less than 64 bytes that has a valid CRC (when CRCOKEN = 1).

bit 4 **RUNTEN:** Runt Enable bit
1 = The received packet must not be a runt packet for the packet to be accepted
0 = Disable Runt filtering

This bit allows the user to reject all runt packets. For this filter, a runt packet is defined as any packet with a size of less than 64 bytes.

bit 3 **UCEN:** Unicast Enable bit
1 = Enable Unicast Filtering
0 = Disable Unicast Filtering

This bit allows the user to accept all unicast packets whose Destination Address matches the Station Address.

bit 2 **NOTMEEN:** Not Me Unicast Enable bit
1 = Enable Not Me Unicast Filtering
0 = Disable Not Me Unicast Filtering

This bit allows the user to accept all unicast packets whose Destination Address does not match the Station Address.

bit 1 **MCEN:** Multicast Enable bit
1 = Enable Multicast Filtering
0 = Disable Multicast Filtering

This bit allows the user to accept all Multicast Address packets.

bit 0 **BCEN:** Broadcast Enable bit
1 = Enable Broadcast Filtering
0 = Disable Broadcast Filtering

This bit allows the user to accept all Broadcast Address packets.

Note 1: XOR = True when either one or the other conditions are true, but not both.
2: This Hash Table Filter match is active regardless of the value of the HTEN bit.
3: This Magic Packet Filter match is active regardless of the value of the MPEN bit.

Note 1: This register is only used for RX operations.
2: The bits in this register may only be changed while the RXEN bit (ETHCON1<8>) = 0.

REGISTER 30-14: ETHIRQ: ETHERNET CONTROLLER INTERRUPT REQUEST REGISTER

bit 7 **RXDONE:** Receive Done Interrupt bit(2)

1 = RX packet was successfully received
0 = No interrupt pending

This bit is set whenever an RX packet is successfully received. It is cleared by either a Reset or CPU write of a '1' to the CLR register.

bit 6 **PKTPEND:** Packet Pending Interrupt bit(2)

1 = RX packet pending in memory
0 = RX packet is not pending in memory

This bit is set when the BUFCNT counter has a value other than '0'. It is cleared by either a Reset or by writing the BUFCDEC bit to decrement the BUFCNT counter. Writing a '0' or a '1' has no effect.

bit 5 **RXACT:** Receive Activity Interrupt bit(2)

1 = RX packet data was successfully received
0 = No interrupt pending

This bit is set whenever RX packet data is stored in the RXBM FIFO. It is cleared by either a Reset or CPU write of a '1' to the CLR register.

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

bit 3 **TXDONE:** Transmit Done Interrupt bit(2)

1 = TX packet was successfully sent
0 = No interrupt pending

This bit is set when the currently transmitted TX packet completes transmission, and the Transmit Status Vector is loaded into the first descriptor used for the packet. It is cleared by either a Reset or CPU write of a '1' to the CLR register.

bit 2 **TXABORT:** Transmit Abort Condition Interrupt bit(2)

1 = TX abort condition occurred on the last TX packet
0 = No interrupt pending

This bit is set when the MAC aborts the transmission of a TX packet for one of the following reasons:

- Jumbo TX packet abort
- Underrun abort
- Excessive defer abort
- Late collision abort
- Excessive collisions abort

This bit is cleared by either a Reset or CPU write of a '1' to the CLR register.

bit 1 **RXBUFNA:** Receive Buffer Not Available Interrupt bit(2)

1 = RX Buffer Descriptor Not Available condition has occurred
0 = No interrupt pending

This bit is set by a RX Buffer Descriptor Overrun condition. It is cleared by either a Reset or a CPU write of a '1' to the CLR register.

bit 0 **RXOVFLW:** Receive FIFO Over Flow Error bit(2)

1 = RX FIFO Overflow Error condition has occurred
0 = No interrupt pending

RXOVFLW is set by the RXBM Logic for an RX FIFO Overflow condition. It is cleared by either a Reset or CPU write of a '1' to the CLR register.

Note 1: This bit is only used for TX operations.
2: This bit is are only used for RX operations.

Note:	It is recommended to use the SET, CLR, or INV registers to set or clear any bit in this register. Setting or clearing any bits in this register should only be done for debug/test purposes.

REGISTER 30-27: EMAC1CLRT: ETHERNET CONTROLLER MAC COLLISION WINDOW/RETRY LIMIT 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	R/W-1	R/W-1	R/W-0	R/W-1	R/W-1	R/W-1
	—	—	CWINDOW<5:0>					
7:0	U-0	U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1
	—	—	—	—	RETX<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 31-14 **Unimplemented:** Read as '0'

bit 13-8 **CWINDOW<5:0>:** Collision Window bits

This is a programmable field representing the slot time or collision window during which collisions occur in properly configured networks. Since the collision window starts at the beginning of transmission, the preamble and SFD is included. Its default of 0x37 (55d) corresponds to the count of frame bytes at the end of the window.

bit 7-4 **Unimplemented:** Read as '0'

bit 3-0 **RETX<3:0>:** Retransmission Maximum bits

This is a programmable field specifying the number of retransmission attempts following a collision before aborting the packet due to excessive collisions. The Standard specifies the maximum number of attempts (attemptLimit) to be 0xF (15d). Its default is '0xF'.

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.

REGISTER 30-32: EMAC1MCMD: ETHERNET CONTROLLER MAC MII MANAGEMENT COMMAND 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	U-0	R/W-0	R/W-0
	—	—	—	—	—	—	SCAN	READ

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 31-2 **Unimplemented:** Read as '0'

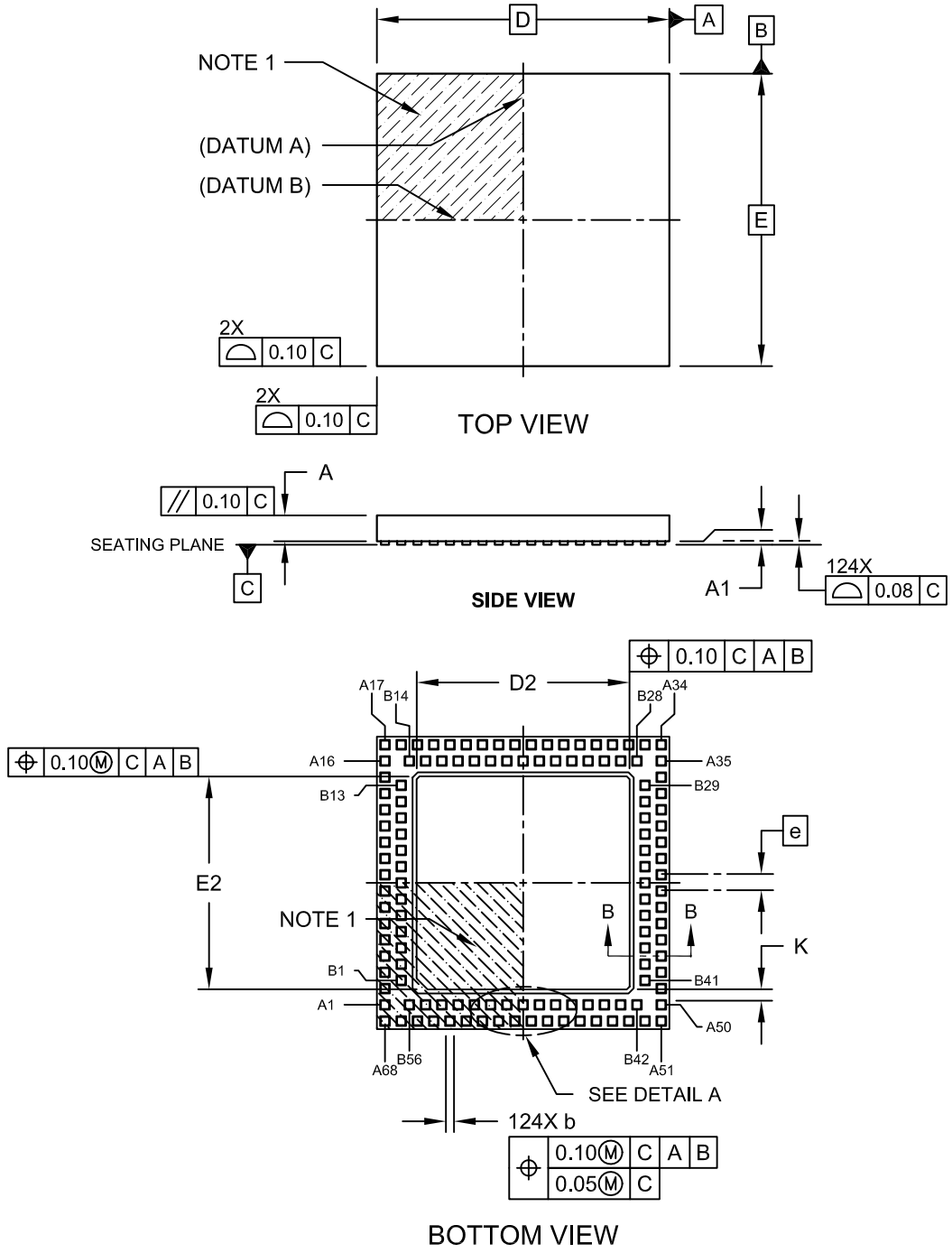
bit 1 **SCAN:** MII Management Scan Mode bit
1 = The MII Management module will perform read cycles continuously (for example, useful for monitoring the Link Fail)
0 = Normal Operation

bit 0 **READ:** MII Management Read Command bit
1 = The MII Management module will perform a single read cycle. The read data is returned in the EMAC1MRDD register
0 = The MII Management module will perform a write cycle. The write data is taken from the EMAC1MWTD register

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.

124-Terminal Very Thin Leadless Array Package (TL) – 9x9x0.9 mm Body [VTLA]

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



Microchip Technology Drawing C04-193A Sheet 1 of 2

B.2 Analog-to-Digital Converter (ADC)

The PIC32MZ EC family features a Pipelined ADC module, while the PIC32MZ EF family of devices has an entirely new 12-bit High-Speed SAR ADC module. Nearly all registers in this new ADC module differ from the registers in PIC32MZ EC devices. Due to this difference, code will not port from PIC32MZ EC devices to PIC32MZ EF devices. Table B-2 lists some of the differences in registers to note to adapt code as quickly as possible.

TABLE B-2: ADC DIFFERENCES

PIC32MZ EC Feature	PIC32MZ EF Feature
Clock Selection and Operating Frequency (TAD)	
On PIC32MZ EC devices, there are three possible sources of the ADC clock: FRC, REFCLKO3, and SYSCLK.	On PIC32MZ EF devices, there are four sources for the ADC clock. In addition to the ones for PIC32MZ EC, PBCLK4 is added as a source. Also, the clock source selection is in a different register.
ADCSEL<1:0> (AD1CON1<9:8>) 11 = FRC 10 = REFCLKO3 01 = SYSCLK 00 = Reserved	ADCSEL<1:0> (ADCCON3<31:30>) 11 = FRC 10 = REFCLKO3 01 = SYSCLK **00 = PBCLK4**
Scan Trigger Sources	
On PIC32MZ EC devices, there are 10 available trigger sources for starting ADC sampling and conversion.	On PIC32MZ EF devices, two new sources have been added. One is a shared trigger source (STRIG). The other is a Global Level Software Trigger (GLSWTRG). With the GLSWTRG, the conversions continue until the bit is cleared in software.
STRGSRC<4:0> (AD1CON1<26:22>) 11111 = Reserved • • • 01101 = Reserved 01100 = Comparator 2 COUT 01011 = Comparator 1 COUT 01010 = OCMP5 01001 = OCMP3 01000 = OCMP1 00111 = TMR5 match 00110 = TMR3 match 00101 = TMR1 match 00100 = INT0 **00011 = Reserved** **00010 = Reserved** 00001 = Global Software Trigger (GSWTRG) 00000 = No trigger	TRGSRC<4:0> (ADCTRGx<y:z>) 11111 = Reserved • • • 01101 = Reserved 01100 = Comparator 2 COUT 01011 = Comparator 1 COUT 01010 = OCMP5 01001 = OCMP3 01000 = OCMP1 00111 = TMR5 match 00110 = TMR3 match 00101 = TMR1 match 00100 = INT0 **00011 = STRIG** **00010 = Global Level Software Trigger (GLSWTRG)** 00001 = Global Software Trigger (GSWTRG) 00000 = No trigger
Debug Mode	
On PIC32MZ EC devices, the ADC module continues operating when stopping on a breakpoint during debugging.	On PIC32MZ EF devices, the ADC module will stop during debugging when stopping on a breakpoint.
Electrical Specifications and Timing Requirements	
Refer to the **"Electrical Characteristics"** chapter in the PIC32MZ EC 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.
ADC Calibration	
PIC32MZ EC devices require calibration values be copied into the AD1CALx registers before turning on the ADC. These values come from the DEVADCx registers.	PIC32MZ EF devices also require ADC calibration values, but the destination registers are named ADCxCAL.