

28/44-Pin, General Purpose, 16-Bit Flash Microcontrollers with Cryptographic Engine, ISO 7816, USB On-The-Go and XLP Technology

Cryptographic Engine

- · AES Engine with 128,192 or 256-Bit Key
- Supports ECB, CBC, OFB, CTR and CFB128 modes
- DES/Triple DES (TDES) Engine: Supports 2-Key and 3-Key EDE or DED TDES
- · Supports up to Three Unique Keys for TDES
- · Programmatically Secure
- · True Random Number Generator
- · Pseudorandom Number Generator
- · Non-Readable, On-Chip, OTP Key Storages

Universal Serial Bus Features

- USB v2.0 On-The-Go (OTG) Compliant
- Dual Role Capable; can Act as Either Host or Peripheral
- Low-Speed (1.5 Mb/s) and Full-Speed (12 Mb/s) USB Operation in Host mode
- · Full-Speed USB Operation in Device mode
- · High-Precision PLL for USB
- USB Device mode Operation from FRC Oscillator:
 - No crystal oscillator required
- Supports up to 32 Endpoints (16 bidirectional):
 - USB module can use any RAM locations on the device as USB endpoint buffers
- · On-Chip USB Transceiver
- Supports Control, Interrupt, Isochronous and Bulk Transfers
- · On-Chip Pull-up and Pull-Down Resistors

Extreme Low-Power Features

- Multiple Power Management Options for Extreme Power Reduction:
 - VBAT allows the device to transition to a backup battery for the lowest power consumption with RTCC
 - Deep Sleep allows near total power-down, with the ability to wake-up on internal or external triggers
 - Sleep and Idle modes selectively shut down peripherals and/or core for substantial power reduction and fast wake-up
 - Doze mode allows CPU to run at a lower clock speed than peripherals
- Alternate Clock modes allow On-the-Fly Switching to a Lower Clock Speed for Selective Power Reduction
- Extreme Low-Power Current Consumption for Deep Sleep:
 - WDT: 270 nA @ 3.3V typical
 - RTCC: 400 nA @ 32 kHz, 3.3V typical
 - Deep Sleep current: 40 nA, 3.3V typical

	Men	nory		Analog Peripherals			Digital Peripherals								T	ohic
Device	Program Flash (bytes)	Data RAM (bytes)	Pins	10/12-Bit A/D (ch)	Comparators	CTMU (ch)	Input Capture	Output Compare/PWM	I ² C TM	IdS	UART w/lrDA® 7816	dSd/dWd3	16-Bit Timers	USB OTG	Deep Sleep w/VBAT	AES/DES Cryptographic
PIC24FJ128GB204	128K	8K	44	12	3	12	6	6	2	3	4	Υ	5	Υ	Υ	Υ
PIC24FJ128GB202	128K	8K	28	9	3	9	6	6	2	3	4	N	5	Υ	Υ	Υ
PIC24FJ64GB204	64K	8K	44	12	3	12	6	6	2	3	4	Υ	5	Υ	Υ	Υ
PIC24FJ64GB202	64K	8K	28	9	3	9	6	6	2	3	4	N	5	Υ	Υ	Υ

Analog Features

- 10/12-Bit, 12-Channel Analog-to-Digital (A/D) Converter:
 - Conversion rate of 500 ksps (10-bit), 200 ksps (12-bit)
 - Conversion available during Sleep and Idle
- Three Rail-to-Rail, Enhanced Analog Comparators with Programmable Input/Output Configuration
- Three On-Chip Programmable Voltage References
- · Charge Time Measurement Unit (CTMU):
 - Used for capacitive touch sensing, up to 12 channels
 - Time measurement down to 100 ps resolution
 - Operation in Sleep mode

Peripheral Features

- · Up to Five External Interrupt Sources
- Peripheral Pin Select (PPS); Allows Independent I/O Mapping of many Peripherals
- Five 16-Bit Timers/Counters with Prescaler:
 - Can be paired as 32-bit timers/counters
- · Six-Channel DMA supports All Peripheral modules:
 - Minimizes CPU overhead and increases data throughput
- Six Input Capture modules, each with a Dedicated 16-Bit Timer
- Six Output Compare/PWM modules, each with a Dedicated 16-Bit Timer
- Enhanced Parallel Master/Slave Port (EPMP/EPSP)
- Hardware Real-Time Clock/Calendar (RTCC):
 - Runs in Sleep, Deep Sleep and VBAT modes
- Three 3-Wire/4-Wire SPI modules:
 - Support four Frame modes
 - Variable FIFO buffer
 - I²S mode
 - Variable width from 2-bit to 32-bit
- Two I²C[™] modules Support Multi-Master/ Slave mode and 7-Bit/10-Bit Addressing
- · Four UART modules:
 - Support RS-485, RS-232 and LIN/J2602
 - On-chip hardware encoder/decoder for IrDA®
 - Smart Card ISO 7816 support on UART1 and UART2 only:
 - T = 0 protocol with automatic error handling
 - T = 1 protocol
 - Dedicated Guard Time Counter (GTC)
 - Dedicated Waiting Time Counter (WTC)
 - Auto-wake-up on Auto-Baud Detect (ABD)
 - 4-level deep FIFO buffer
- Programmable 32-Bit Cyclic Redundancy Check (CRC) Generator
- Digital Signal Modulator provides On-Chip FSK and PSK Modulation for a Digital Signal Stream
- High-Current Sink/Source (18 mA/18 mA) on All I/O Pins
- · Configurable Open-Drain Outputs on Digital I/O Pins
- 5.5V Tolerant Inputs on Most Pins

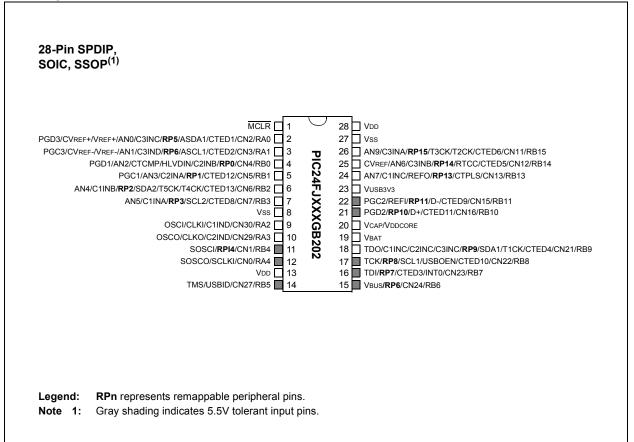
High-Performance CPU

- · Modified Harvard Architecture
- Up to 16 MIPS Operation @ 32 MHz
- · 8 MHz Internal Oscillator:
 - 96 MHz PLL option
 - Multiple clock divide options
 - Run-time self-calibration capability for maintaining better than ±0.20% accuracy
 - Fast start-up
- 17-Bit x 17-Bit Single-Cycle Hardware Fractional/Integer Multiplier
- 32-Bit by 16-Bit Hardware Divider
- 16 x 16-Bit Working Register Array
- C Compiler Optimized Instruction Set Architecture (ISA)
- Two Address Generation Units (AGUs) for Separate Read and Write Addressing of Data Memory

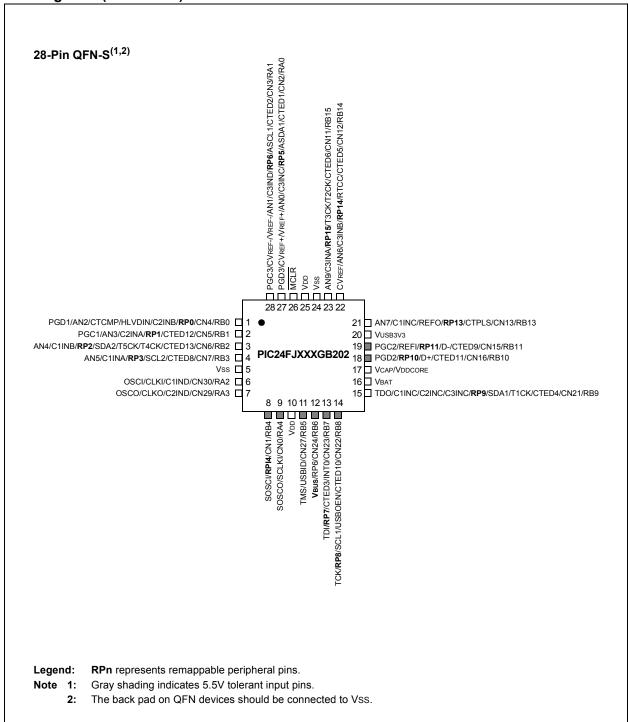
Special Microcontroller Features

- Supply Voltage Range of 2.0V to 3.6V
- Two On-Chip Voltage Regulators (1.8V and 1.2V) for Regular and Extreme Low-Power Operation
- 20,000 Erase/Write Cycle Endurance Flash Program Memory, Typical
- Flash Data Retention: 20 Years Minimum
- · Self-Programmable under Software Control
- · Programmable Reference Clock Output
- In-Circuit Serial Programming™ (ICSP™) and In-Circuit Emulation (ICE) via 2 Pins
- JTAG Programming and Boundary Scan Support
- Fail-Safe Clock Monitor (FSCM) Operation:
 - Detects clock failure and switches to on-chip, Low-Power RC Oscillator
- Power-on Reset (POR), Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
- Separate Brown-out Reset (BOR) and Deep Sleep Brown-out Reset (DSBOR) Circuits
- Programmable High/Low-Voltage Detect (HLVD)
- Flexible Watchdog Timer (WDT) with its Own RC Oscillator for Reliable Operation
- Standard and Ultra Low-Power Watchdog Timers (ULPW) for Reliable Operation in Standard and Deep Sleep modes

Pin Diagrams



Pin Diagrams (Continued)



Pin Diagrams (Continued)

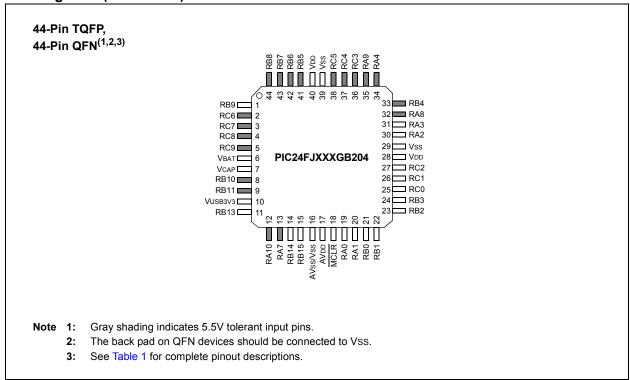


TABLE 1: PIC24FJXXXGB204 PIN FUNCTION DESCRIPTIONS

Pin	Function	Pin	Function
1	C1INC/C2INC/C3INC/RP9/SDA1/T1CK/CTED4/PMD3/CN21/RB9	23	AN4/C1INB/ RP2 /SDA2/T5CK/T4CK/CTED13/CN6/PMD2/RB2
2	RP22/PMA1/PMALH/CN18/RC6	24	AN5/C1INA/RP3/SCL2/CTED8/CN7/PMWR/RB3
3	RP23/PMA0/PMALL/CN17/RC7	25	AN10/ RP16 /PMBE1/CN8/RC0
4	RP24/PMA5/CN20/RC8	26	AN11/ RP17 /CN9/RC1
5	RP25/CTED7/PMA6/CN19/RC9	27	AN12/ RP18 /PMACK1/CN10/RC2
6	VBAT	28	VDD
7	VCAP	29	Vss
8	RP10/CTED11/CN16/PGD2/D+/RB10	30	OSCI/C1IND/CLKI/PMCS1/CN30/RA2
9	REFI/RP11/CTED9/CN15/PGC2/D-/RB11	31	OSCO/C2IND/CLKO/CN29/RA3
10	VUSB3V3	32	TDO/PMA8/CN34/RA8
11	AN7/C1INC/REFO/RP13/CTPLS/PMRD/CN13/RB13	33	SOSCI/CN1/ RPI4 /RB4
12	TMS/PMA2/PMALU/CN36/RA10	34	SOSCO/SCLKI/CN0/RA4
13	TCK/PMA7/CN33/RA7	35	TDI/PMA9/CN35/RA9
14	CVREF/AN6/C3INB/ RP14 /RTCC/CTED5/CN12/RB14	36	RP19/PMBE0/CN28/RC3
15	AN9/C3INA/ RP15 /T3CK/T2CK/CTED6/PMA14/CS1/CN11/PMCS/ PMCS1/RB15	37	RP20/PMA4/CN25/RC4
16	AVss/Vss	38	RP21/PMA3/CN26/RC5
17	AVDD	39	Vss
18	MCLR	40	VDD
19	CVREF+/VREF+/AN0/C3INC/RP5/ASDA1 ⁽¹⁾ /CTED1/CN2/PMD7/PGD3/RA0	41	CN27/USBID/RB5
20	CVREF-/VREF-/AN1/C3IND/RP6/ASCL1 ⁽¹⁾ /CTED2/CN3/PGC3/RA1	42	PMD6/CN24/VBus/RB6
21	AN2/CTCMP/C2INB/ RP0 /CN4/PGD1/HLVDIN/PMD0/RB0	43	RP7/CTED3/INT0/CN23/PMD5/RB7
22	AN3/C2INA/RP1/CTED12/CN5/PMD1/PGC1/RB1	44	RP8/SCL1/CTED10/PMD4/CN22/USBOEN/RB8

 $\textbf{Legend:} \ \ \textbf{RPn} \ \text{represents remappable peripheral pins}.$

Note 1: Alternative multiplexing for SDA1 and SCL1 when the I2C1SEL Configuration bit is set.

Table of Contents

1.0	Device Overview	9
2.0	Guidelines for Getting Started with 16-Bit Microcontrollers	23
3.0	CPU	29
4.0	Memory Organization	
5.0	Direct Memory Access Controller (DMA)	71
6.0	Flash Program Memory	
7.0	Resets	85
8.0	Interrupt Controller	
9.0	Oscillator Configuration	147
10.0	Power-Saving Features	161
11.0	I/O Ports	173
	Timer1	
	Timer2/3 and Timer4/5	
	Input Capture with Dedicated Timers	
	Output Compare with Dedicated Timers	
	Serial Peripheral Interface (SPI)	
	Inter-Integrated Circuit™ (I ² C™)	
	Universal Asynchronous Receiver Transmitter (UART)	
	Universal Serial Bus with On-The-Go Support (USB OTG)	
	Data Signal Modulator (DSM)	
	Enhanced Parallel Master Port (EPMP)	
	Real-Time Clock and Calendar (RTCC)	
23.0	Cryptographic Engine	
	32-Bit Programmable Cyclic Redundancy Check (CRC) Generator	
	12-Bit A/D Converter with Threshold Detect	
	Triple Comparator Module	
	Comparator Voltage Reference	
	Charge Time Measurement Unit (CTMU)	
	High/Low-Voltage Detect (HLVD)	
	Special Features	
	Development Support	
	Instruction Set Summary	
	Electrical Characteristics	
	Packaging Information	
	endix A: Revision History	
	X	
	Microchip Web Site	
	omer Change Notification Service	
	tomer Support	4/5

TO OUR VALUED CUSTOMERS

It is our intention to provide our valued customers with the best documentation possible to ensure successful use of your Microchip products. To this end, we will continue to improve our publications to better suit your needs. Our publications will be refined and enhanced as new volumes and updates are introduced.

If you have any questions or comments regarding this publication, please contact the Marketing Communications Department via E-mail at **docerrors@microchip.com**. We welcome your feedback.

Most Current Data Sheet

To obtain the most up-to-date version of this data sheet, please register at our Worldwide Web site at:

http://www.microchip.com

You can determine the version of a data sheet by examining its literature number found on the bottom outside corner of any page. The last character of the literature number is the version number, (e.g., DS30000000A is version A of document DS30000000).

Errata

An errata sheet, describing minor operational differences from the data sheet and recommended workarounds, may exist for current devices. As device/documentation issues become known to us, we will publish an errata sheet. The errata will specify the revision of silicon and revision of document to which it applies.

To determine if an errata sheet exists for a particular device, please check with one of the following:

- Microchip's Worldwide Web site; http://www.microchip.com
- Your local Microchip sales office (see last page)

When contacting a sales office, please specify which device, revision of silicon and data sheet (include literature number) you are using.

Customer Notification System

Register on our web site at www.microchip.com to receive the most current information on all of our products.

NOTES:

1.0 DEVICE OVERVIEW

This document contains device-specific information for the following devices:

- PIC24FJ64GB202
- PIC24FJ128GB202
- PIC24FJ64GB204
- PIC24FJ128GB204

The PIC24FJ128GB204 family expands the capabilities of the PIC24F family by adding a complete selection of Cryptographic Engines, ISO 7816 support and $\rm I^2S$ support to its existing features. This combination, along with its ultra low-power features, Direct Memory Access (DMA) for peripherals and USB On-The-Go, make this family the new standard for mixed-signal $\rm PIC^{\scriptsize (8)}$ microcontrollers in one economical and power-saving package.

1.1 Core Features

1.1.1 16-BIT ARCHITECTURE

Central to all PIC24F devices is the 16-bit modified Harvard architecture, first introduced with Microchip's dsPIC[®] Digital Signal Controllers (DSCs). The PIC24F CPU core offers a wide range of enhancements, such as:

- 16-bit data and 24-bit address paths with the ability to move information between data and memory spaces
- Linear addressing of up to 12 Mbytes (program space) and 32 Kbytes (data)
- A 16-element Working register array with built-in software stack support
- A 17 x 17 hardware multiplier with support for integer math
- Hardware support for 32 by 16-bit division
- An instruction set that supports multiple addressing modes and is optimized for high-level languages, such as 'C'
- · Operational performance up to 16 MIPS

1.1.2 XLP POWER-SAVING TECHNOLOGY

The PIC24FJ128GB204 family of devices introduces a greatly expanded range of power-saving operating modes for the ultimate in power conservation. The new modes include:

- Retention Sleep with essential circuits being powered from a separate low-voltage regulator
- Deep Sleep without RTCC for the lowest possible power consumption under software control
- VBAT mode (with or without RTCC) to continue limited operation from a backup battery when VDD is removed

Many of these new low-power modes also support the continuous operation of the low-power, on-chip Real-Time Clock/Calendar (RTCC), making it possible for an application to keep time while the device is otherwise asleep.

Aside from these new features, PIC24FJ128GB204 family devices also include all of the legacy power-saving features of previous PIC24F microcontrollers, such as:

- On-the-Fly Clock Switching, allowing the selection of a lower power clock during run time
- Doze Mode Operation, for maintaining peripheral clock speed while slowing the CPU clock
- Instruction-Based Power-Saving Modes, for quick invocation of Idle and the many Sleep modes

1.1.3 OSCILLATOR OPTIONS AND FEATURES

All of the devices in the PIC24FJ128GB204 family offer five different oscillator options, allowing users a range of choices in developing application hardware. These include:

- · Two Crystal modes
- Two External Clock (EC) modes
- A Phase-Locked Loop (PLL) frequency multiplier, which allows clock speeds of up to 32 MHz
- A Fast Internal Oscillator (FRC) Nominal 8 MHz output with multiple frequency divider options and automatic frequency self-calibration during run time
- A separate Low-Power Internal RC Oscillator (LPRC) – 31 kHz nominal for low-power, timing-insensitive applications.

The internal oscillator block also provides a stable reference source for the Fail-Safe Clock Monitor (FSCM). This option constantly monitors the main clock source against a reference signal provided by the internal oscillator and enables the controller to switch to the internal oscillator, allowing for continued low-speed operation or a safe application shutdown.

1.1.4 EASY MIGRATION

Regardless of the memory size, all devices share the same rich set of peripherals, allowing for a smooth migration path as applications grow and evolve. This extends the ability of applications to grow from the relatively simple, to the powerful and complex, yet still selecting a Microchip device.

1.2 DMA Controller

PIC24FJ128GB204 family devices also add a Direct Memory Access (DMA) Controller to the existing PIC24F architecture. The DMA acts in concert with the CPU, allowing data to move between data memory and peripherals without the intervention of the CPU, increasing data throughput and decreasing execution time overhead. Six independently programmable channels make it possible to service multiple peripherals at virtually the same time, with each channel peripheral performing a different operation. Many types of data transfer operations are supported.

1.3 USB On-The-Go (OTG)

USB On-The-Go provides on-chip functionality as a target device compatible with the USB 2.0 standard, as well as limited stand-alone functionality as a USB embedded host. By implementing USB Host Negotiation Protocol (HNP), the module can also dynamically switch between device and host operation, allowing for a much wider range of versatile USB-enabled applications on a microcontroller platform.

PIC24FJ128GB204 family devices also incorporate an integrated USB transceiver and precision oscillator, minimizing the required complexity of implementing a complete USB device, embedded host, dual role or On-The-Go application.

1.4 Cryptographic Engine

The Cryptographic Engine provides a new set of data security options. Using its own free-standing state machines, the engine can independently perform NIST standard encryption and decryption of data, independently of the CPU.

Support for True Random Number Generation (TRNG) and Pseudorandom Number Generation (PRNG); NIST SP800-90 compliant.

1.5 Other Special Features

- Peripheral Pin Select (PPS): The Peripheral Pin Select feature allows most digital peripherals to be mapped over a fixed set of digital I/O pins. Users may independently map the input and/or output of any one of the many digital peripherals to any one of the I/O pins.
- Communications: The PIC24FJ128GB204 family incorporates a range of serial communication peripherals to handle a range of application requirements. There are two independent I²C™ modules that support both Master and Slave modes of operation. Devices also have, through the PPS feature, four independent UARTs with built-in IrDA[®] encoders/decoders, ISO 7816 Smart Card support (UART1 and UART2 only) and three SPI modules with I²S and variable data width support.
- Analog Features: All members of the PIC24FJ128GB204 family include a 12-bit A/D Converter module and a triple comparator module. The A/D module incorporates a range of new features that allows the converter to assess and make decisions on incoming data, reducing CPU overhead for routine A/D conversions. The comparator module includes three analog comparators that are configurable for a wide range of operations.
- CTMU Interface: In addition to their other analog features, members of the PIC24FJ128GB204 family include the CTMU interface module. This provides a convenient method for precision time measurement and pulse generation, and can serve as an interface for capacitive sensors.
- Enhanced Parallel Master/Parallel Slave Port:
 This module allows rapid and transparent access to the microcontroller data bus, and enables the CPU to directly address external data memory. The parallel port can function in Master or Slave mode, accommodating data widths of 4, 8 or 16 bits, and address widths of up to 23 bits in Master modes.
- Real-Time Clock and Calendar (RTCC): This
 module implements a full-featured clock and
 calendar with alarm functions in hardware, freeing
 up timer resources and program memory space
 for use by the core application.
- Data Signal Modulator (DSM): The Data Signal Modulator (DSM) allows the user to mix a digital data stream (the "modulator signal") with a carrier signal to produce a modulated output.

1.6 Details on Individual Family Members

Devices in the PIC24FJ128GB204 family are available in 28-pin and 44-pin packages. The general block diagram for all devices is shown in Figure 1-1.

The devices are differentiated from each other in six ways:

- Flash program memory (64 Kbytes for PIC24FJ64GB2XX devices and 128 Kbytes for PIC24FJ128GB2XX devices).
- Available I/O pins and ports (21 pins on two ports for 28-pin devices, 35 pins on three ports for 44-pin devices).
- Available Input Change Notification (ICN) inputs (20 on 28-pin devices and 34 on 44-pin devices).
- 4. Available remappable pins (14 pins on 28-pin devices and 24 pins on 44-pin devices).
- Analog input channels for the A/D Converter (12 channels for 44-pin devices and 9 channels for 28-pin devices).

All other features for devices in this family are identical. These are summarized in Table 1-1 and Table 1-2.

A list of the pin features available on the PIC24FJ128GB204 family devices, sorted by function, is shown in Table 1-3. Note that this table shows the pin location of individual peripheral features and not how they are multiplexed on the same pin. This information is provided in the pinout diagrams in the beginning of the data sheet. Multiplexed features are sorted by the priority given to a feature, with the highest priority peripheral being listed first.

TABLE 1-1: DEVICE FEATURES FOR THE PIC24FJ128GB204 FAMILY: 44-PIN DEVICES

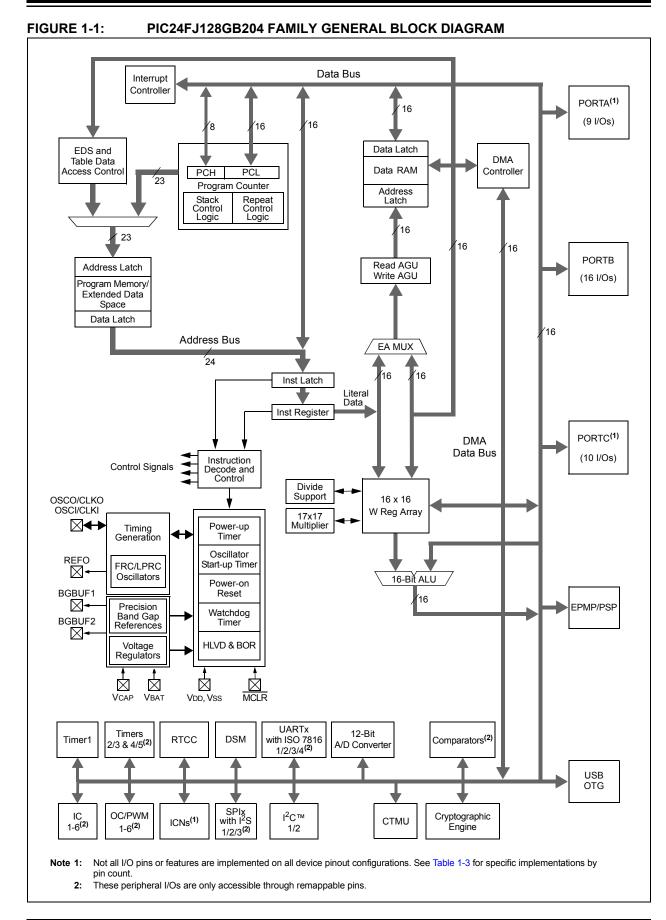
Features	PIC24FJ64GB204	PIC24FJ128GB204						
Operating Frequency	DC – 32	! MHz						
Program Memory (bytes)	64K	128K						
Program Memory (instructions)	22,016	44,032						
Data Memory (bytes)	8K							
Interrupt Sources (soft vectors/ NMI traps)	72 (68	3/4)						
I/O Ports	Ports A,	B, C						
Total I/O Pins	34							
Remappable Pins	24 (23 I/Os, 1	Input only)						
Timers: Total Number (16-bit)	5(1))						
32-Bit (from paired 16-bit timers)	2							
Input Capture w/Timer Channels	6(1)							
Output Compare/PWM Channels	6(1))						
Input Change Notification Interrupts	34							
Serial Communications:								
UART	4(1)							
SPI (3-wire/4-wire)	3(1))						
I ² C™	2							
Digital Signal Modulator (DSM)	Yes	8						
Parallel Communications (EPMP/PSP)	Yes	3						
JTAG Boundary Scan	Yes	3						
12-Bit SAR Analog-to-Digital (A/D) Converter (input channels)	12							
Analog Comparators	3							
CTMU Interface	12 Chai	nnels						
Resets (and Delays)	Core POR, VDD POR, VBAT POR, BOR, RESET Instruction, MCLR, WDT; Illegal Opcode, REPEAT Instruction, Hardware Traps, Configuration Word Mismatch (OST, PLL Lock)							
Instruction Set	76 Base Instructions, Multiple	Addressing Mode Variations						
Packages	44-Pin TQFF	and QFN						
Cryptographic Engine	Supports AES with 128, 192 and 256-Bit Key, DES and TDES, True Random and Pseudorandom Number Generator, On-Chip OTP Storage							
USB	USB Full-Speed and Low-Speed Compatible, On-The-Go (OTG) USB							
RTCC	Yes	3						

Note 1: Peripherals are accessible through remappable pins.

TABLE 1-2: DEVICE FEATURES FOR THE PIC24FJ128GB204 FAMILY: 28-PIN DEVICES

Features	PIC24FJ64GB202	PIC24FJ128GB202
Operating Frequency	DC - 32	2 MHz
Program Memory (bytes)	64K	128K
Program Memory (instructions)	22,016	44,032
Data Memory (bytes)	8K	
Interrupt Sources (soft vectors/ NMI traps)	72 (68	3/4)
I/O Ports	Ports A	A, B
Total I/O Pins	20	
Remappable Pins	15 (14 I/Os, 1	Input only)
Timers:		
Total Number (16-bit)	5 ⁽¹)
32-Bit (from paired 16-bit timers)	2	
Input Capture w/Timer Channels	6 ⁽¹)
Output Compare/PWM Channels	6 ⁽¹)
Input Change Notification Interrupts	20	
Serial Communications:		
UART	4(1)
SPI (3-wire/4-wire)	3 ⁽¹)
I ² C™	2	
Digital Signal Modulator (DSM)	Yes	6
JTAG Boundary Scan	Yes	6
12-Bit SAR Analog-to-Digital (A/D) Converter (input channels)	9	
Analog Comparators	3	
CTMU Interface	9 Char	nnels
Resets (and Delays)	Core POR, VDD POR, VBAT POMCLR, WDT; Illegal Opco Hardware Traps, Configu (OST, PLI	de, REPEAT Instruction, ration Word Mismatch
Instruction Set	76 Base Instructions, Multiple	Addressing Mode Variations
Packages	28-Pin SPDIP, SSOP	, SOIC and QFN-S
Cryptographic Engine	Supports AES with 128, 192 and True Random and Pseudora On-Chip OT	indom Number Generator,
USB	USB Full-Speed and Low-Speed Co	mpatible, On-The-Go (OTG) USB
RTCC	Yes	5

Note 1: Peripherals are accessible through remappable pins.



DS30005009C-page 14

TABLE 1-3: PIC24FJ128GB204 FAMILY PINOUT DESCRIPTION

	Pin Number/Grid Locator					
Pin Function	28-Pin SPDIP/SOIC/ SSOP	28-Pin QFN-S	44-Pin TQFP/QFN	I/O	O Input Buffer	Description
AN0	2	27	19	ı	ANA	12-Bit SAR A/D Converter Inputs.
AN1	3	28	20	ı	ANA	
AN2	4	1	21	ı	ANA	
AN3	5	2	22	I	ANA	
AN4	6	3	23	ı	ANA	
AN5	7	4	24	ı	ANA	
AN6	25	22	14	I	ANA	
AN7	24	21	11	ı	ANA	
AN9	26	23	15	I	ANA	
AN10	_		25	I	ANA	
AN11	_	_	26	ı	ANA	
AN12	_	_	27	I	ANA	
ASCL1	3	28	20	_	_	
ASDA1	2	27	19	_	_	
AVDD	_	_	17	Р	ANA	Positive Supply for Analog modules.
AVss	_	24	16	Р	ANA	Ground Reference for Analog modules.
C1INA	7	4	24	ı	ANA	Comparator 1 Input A.
C1INB	6	3	23	I	ANA	Comparator 1 Input B.
C1INC	24	15	1	I	ANA	Comparator 1 Input C.
C1IND	9	6	30	ı	ANA	Comparator 1 Input D.
C2INA	5	2	22	I	ANA	Comparator 2 Input A.
C2INB	4	1	21	ı	ANA	Comparator 2 Input B.
C2INC	18	15	1	ı	ANA	Comparator 2 Input C.
C2IND	10	7	31	ı	ANA	Comparator 2 Input D.
C3INA	26	23	15	I	ANA	Comparator 3 Input A.
C3INB	25	22	14	I	ANA	Comparator 3 Input B.
C3INC	2	15	1	I	ANA	Comparator 3 Input C.
C3IND	3	28	20	I	ANA	Comparator 3 Input D.
CLKI	9	6	30	I	ANA	Main Clock Input Connection.
CLKO	10	7	31	0	_	System Clock Output.

Legend: ST = Schmitt Trigger input

ANA = Analog input

 $I^2C = ST \text{ with } I^2C^{TM} \text{ or SMBus levels}$

TTL = TTL compatible input

I = Input

O = Output

TABLE 1-3: PIC24FJ128GB204 FAMILY PINOUT DESCRIPTION (CONTINUED)

	Pin Number/Grid Locator						
Pin Function	28-Pin SPDIP/SOIC/ SSOP	28-Pin QFN-S	44-Pin TQFP/QFN	I/O	I/O Input Buffer	Description	
CN0	12	9	34	_		Interrupt-on-Change Inputs.	
CN1	11	8	33	_	_		
CN2	2	27	19				
CN3	3	28	20	I			
CN4	4	1	21		_		
CN5	5	2	22	_	_		
CN6	6	3	23	_	_		
CN7	7	4	24	_	_		
CN8	_	_	25	_	_		
CN9	_	_	26	_	_		
CN10	_	_	27	_	_		
CN11	26	23	15	_	_		
CN12	25	22	14	_	_		
CN13	24	21	11	_	_		
CN15	22	19	9	_	_		
CN16	21	18	8	ı			
CN17	_	_	3	_	_		
CN18	_	_	2	_	_		
CN19	_	_	5	I			
CN20	_	_	4				
CN21	18	15	1				
CN22	17	14	44	I			
CN23	16	13	43		_		
CN24	15	12	42				
CN25	_	_	37	I			
CN26			38	_	_		
CN27	14	11	41				
CN28	_		36	_	_		
CN29	10	7	31	_	_		
CN30	9	6	30	_			
CN33	_	_	13	_	_		
CN34	_	_	32				
CN35	_	_	35	_			
CN36	_	_	12		_		
CTCMP	4	1	21	TTI	ANA	CTMU Comparator 2 Input (Pulse mode).	

Legend: ST = Schmitt Trigger input

ANA = Analog input I^2C = ST with I^2C^{TM} or SMBus levels

TTL = TTL compatible input

O = Output

I = Input

TABLE 1-3: PIC24FJ128GB204 FAMILY PINOUT DESCRIPTION (CONTINUED)

	Pin Number/Grid Locator		Locator			
Pin Function	28-Pin SPDIP/SOIC/ SSOP	28-Pin QFN-S	44-Pin TQFP/QFN	I/O	/O Input Buffer	Description
CTED1	2	27	19	ı	ANA	CTMU External Edge Inputs.
CTED2	3	28	20	I	ANA	
CTED3	16	13	43	I	ANA	
CTED4	18	15	1	I	ANA	
CTED5	25	22	14	I	ANA	
CTED6	26	23	15	I	ANA	
CTED7		_	5	I	ANA	
CTED8	7	4	24	I	ANA	
CTED9	22	19	9	I	ANA	
CTED10	17	14	44	I	ANA	
CTED11	21	18	8	I	ANA	
CTED12	5	2	22	I	ANA	
CTED13	6	3	23	I	ANA	
CTPLS	24	21	11	0		CTMU Pulse Output.
CVREF	25	22	14	0	ANA	Comparator Voltage Reference Output.
CVREF+	2	27	19	I	ANA	Comparator Voltage Reference (high) Input.
CVREF-	3	28	20	I	ANA	Comparator Voltage Reference (low) Input.
D+	21	18	8	I/O		USB Differential Plus Line (internal transceiver).
D-	22	19	9	I/O		USB Differential Minus Line (internal transceiver).
INT0	16	13	43	I	ST	External Interrupt Input 0.
HLVDIN	4	1	21	ı	ANA	High/Low-Voltage Detect Input.
MCLR	1	26	18	I	ST	Master Clear (device Reset) Input. This line is brought low to cause a Reset.
OSCI	9	6	30	I	ANA	Main Oscillator Input Connection.
OSCO	10	7	31	0	_	Main Oscillator Output Connection.
PGC1	5	2	22	I/O	ST	In-Circuit Debugger/Emulator/ICSP™
PGC2	22	19	9	I/O	ST	Programming Clock.
PGC3	3	28	20	I/O	ST	
PGD1	4	1	21	I/O	ST	
PGD2	21	18	8	I/O	ST	
PGD3	2	27	19	I/O	ST	

Legend: ST = Schmitt Trigger input

ANA = Analog input

 $I^2C = ST \text{ with } I^2C^{TM} \text{ or SMBus levels}$

TTL = TTL compatible input

O = Output

I = Input

TABLE 1-3: PIC24FJ128GB204 FAMILY PINOUT DESCRIPTION (CONTINUED)

	Pin Numl	oer/Grid	Locator			
Pin Function	28-Pin SPDIP/SOIC/ SSOP	28-Pin QFN-S	44-Pin TQFP/QFN	I/O	Input Buffer	Description
PMA0/PMALL	_	_	3	0	_	Parallel Master Port Address.
PMA1/PMALH	_	_	2	0	_	
PMA14/PMCS/ PMCS1	_	_	15	0	_	
PMA2/PMALU	_	_	12	0	_	
PMA3	_	_	38	0	_	
PMA4	_	_	37	0	_	
PMA5	_	_	4	0	_	
PMA6	_	_	5	0	_	
PMA7	_	_	13	0	_	
PMA8	_	_	32	0	_	
PMA9	_	_	35	0	_	
PMACK1	_	_	27	I	ST/TTL	Parallel Master Port Acknowledge Input 1.
PMBE0	_	_	36	0	_	Parallel Master Port Byte Enable 0 Strobe.
PMBE1	_	_	25	0	_	Parallel Master Port Byte Enable 1 Strobe.
PMCS1	_	_	30	I/O	ST/TTL	Parallel Master Port Chip Select 1 Strobe.
PMD0	_	_	21	I/O	ST/TTL	Parallel Master Port Data (Demultiplexed
PMD1	_	_	22	I/O	ST/TTL	Master mode) or Address/Data (Multiplexed
PMD2	_		23	I/O	ST/TTL	Master modes).
PMD3	_	_	1	I/O	ST/TTL	
PMD4	_	_	44	I/O	ST/TTL	
PMD5	_	_	43	I/O	ST/TTL	
PMD6	_	_	20	I/O	ST/TTL	
PMD7	_	_	19	I/O	ST/TTL	
PMRD	_	_	11	0	_	Parallel Master Port Read Strobe.
PMWR	_	_	24	0	_	Parallel Master Port Write Strobe.
RA0	2	27	19	I/O	ST	PORTA Digital I/Os.
RA1	3	28	20	I/O	ST	
RA2	9	6	30	I/O	ST	
RA3	10	7	31	I/O	ST	
RA4	12	9	34	ı	ST	
RA7	_	_	13	I/O	ST	
RA8	_	_	32	I/O	ST	
RA9	_	_	35	I/O	ST	
RA10	_	_	12	I/O	ST	

Legend: ST = Schmitt Trigger input

ANA = Analog input I^2C = ST with I^2C^{TM} or SMBus levels

TTL = TTL compatible input

O = Output

I = Input

TABLE 1-3: PIC24FJ128GB204 FAMILY PINOUT DESCRIPTION (CONTINUED)

	Pin Number/Grid Locator					
Pin Function	28-Pin SPDIP/SOIC/ SSOP	28-Pin QFN-S	44-Pin TQFP/QFN	I/O	Input Buffer	Description
RB0	4	1	21	I/O	ST	PORTB Digital I/Os.
RB1	5	2	22	I/O	ST	
RB2	6	3	23	I/O	ST	
RB3	7	4	24	I/O	ST	
RB4	11	8	33	-	ST	
RB5	14	11	41	I/O	ST	
RB6	15	12	42	I/O	ST	
RB7	16	13	43	I/O	ST	
RB8	17	14	44	I/O	ST	
RB9	18	15	1	I/O	ST	
RB10	21	18	8	I/O	ST	
RB11	22	19	9	I/O	ST	
RB13	24	21	11	I/O	ST	
RB14	25	22	14	I/O	ST	
RB15	26	23	15	I/O	ST	
RC0	_		25	1/0	ST	PORTC Digital I/Os.
RC1	_	_	26	I/O	ST	
RC2	_	_	27	I/O	ST	
RC3	_		36	1/0	ST	
RC4	_	_	37	I/O	ST	
RC5			38	1/0	ST	
RC6	_	_	2	I/O	ST	
RC7	_		3	I/O	ST	
RC8	_		4	I/O	ST	
RC9	_	1	5	I/O	ST	
REFI	22	19	9	_	_	Reference Clock Input.
REFO	24	21	11	_	_	Reference Clock Output.

Legend: ST = Schmitt Trigger input

ANA = Analog input

 $I^2C = ST \text{ with } I^2C^{TM} \text{ or SMBus levels}$

TTL = TTL compatible input

O = Output

I = Input

TABLE 1-3: PIC24FJ128GB204 FAMILY PINOUT DESCRIPTION (CONTINUED)

	Pin Number/Grid Locator					
Pin Function	28-Pin SPDIP/SOIC/ SSOP	28-Pin QFN-S	44-Pin TQFP/QFN	I/O	Input Buffer	Description
RP0	4	1	21	I/O	ST	Remappable Peripheral (input or output).
RP1	5	2	22	I/O	ST	
RP2	6	3	23	I/O	ST	
RP3	7	4	24	I/O	ST	
RP5	2	27	19	I/O	ST	
RP6	3,15	28	20	I/O	ST	
RP7	16	13	43	I/O	ST	
RP8	17	14	44	I/O	ST	
RP9	18	15	1	I/O	ST	
RP10	21	18	8	I/O	ST	
RP11	22	19	9	I/O	ST	
RP13	24	21	11	I/O	ST	
RP14	25	22	14	I/O	ST	
RP15	26	23	15	I/O	ST	
RP16	_	_	25	I/O	ST	
RP17	_	_	26	I/O	ST	
RP18	_	_	27	I/O	ST	
RP19	_	_	36	I/O	ST	
RP20	_	_	37	I/O	ST	
RP21	_	_	38	I/O	ST	
RP22	_	_	2	I/O	ST	
RP23	_	_	3	I/O	ST	
RP24	_	_	4	I/O	ST	
RP25	_	_	5	I/O	ST	
RPI4	11	8	33	I	ST	Remappable Peripheral (input).
RTCC	25	22	14	0	_	Real-Time Clock Alarm/Seconds Pulse Output.
SCL1	17	14	44	I/O	I ² C	I2C1 Synchronous Serial Clock Input/Output.
SCL2	7	4	24	I/O	I ² C	I2C2 Synchronous Serial Clock Input/Output.
SCLKI	12	9	34	I	_	Secondary Oscillator Digital Clock Input.
SDA1	18	15	1	I/O	I ² C	I2C1 Data Input/Output.
SDA2	6	3	23	I/O	I ² C	I2C2 Data Input/Output.
SOSCI	11	8	33	I	ANA	Secondary Oscillator/Timer1 Clock Input.
SOSCO	12	9	34	0	ANA	Secondary Oscillator/Timer1 Clock Output.

Legend: ST = Schmitt Trigger input

ANA = Analog input

 $I^2C = ST \text{ with } I^2C^{TM} \text{ or SMBus levels}$

TTL = TTL compatible input

O = Output

I = Input

PIC24FJ128GB204 FAMILY PINOUT DESCRIPTION (CONTINUED) **TABLE 1-3:**

	Pin Numl	per/Grid	Locator					
Pin Function	28-Pin SPDIP/SOIC/ SSOP	28-Pin QFN-S	44-Pin TQFP/QFN	I/O	Input Buffer	Description		
T1CK	18	15	1	ı	ST	Timer1 Clock.		
T2CK	26	23	15	- 1	ST	Timer2 Clock.		
T3CK	26	23	15	I	ST	Timer3 Clock.		
T4CK	6	3	23	I	ST	Timer4 Clock.		
T5CK	6	3	23	- 1	ST	Timer5 Clock.		
TCK	17	14	13	I	ST	JTAG Test Clock/Programming Clock Input.		
TDI	16	13	35	I	ST	JTAG Test Data/Programming Data Input.		
TDO	18	15	32	0	_	JTAG Test Data Output.		
TMS	14	11	12	I	_	JTAG Test Mode Select Input.		
USBID	14	11	41	I	ST	USB OTG ID (OTG mode only).		
USBOEN	17	14	44	0	_	USB Output Enable Control (for external transceiver).		
VBAT	19	16	6	Р	_	Backup Battery (B+) Input (1.2V nominal).		
VBUS	15	12	42	Р	_	USB Voltage, Host mode (5V).		
VCAP	20	17	7	Р	_	External Filter Capacitor Connection.		
VDD	13,28	25	28,40	Р	_	Positive Supply for Peripheral Digital Logic and I/O Pins.		
VDDCORE	20	17	7	_	_	Microcontroller Core Supply Voltage.		
VREF+	2	27	19	I	ANA	A/D Reference Voltage Input (+).		
VREF-	3	28	20	I	ANA	A/D Reference Voltage Input (-).		
Vss	8,27	5,24	29,39	Р	_	Ground Reference for Logic and I/O Pins.		
VUSB3V3	23	20	10	Р	_	USB Transceiver Power Input Voltage (3.3V nominal).		

Legend: ST = Schmitt Trigger input

ANA = Analog input I^2C = ST with I^2C^{TM} or SMBus levels

TTL = TTL compatible input

O = Output

I = Input P = Power

NOTES:

2.0 GUIDELINES FOR GETTING STARTED WITH 16-BIT MICROCONTROLLERS

2.1 Basic Connection Requirements

Getting started with the PIC24FJ128GB204 family of 16-bit microcontrollers requires attention to a minimal set of device pin connections before proceeding with development.

The following pins must always be connected:

- All VDD and Vss pins (see Section 2.2 "Power Supply Pins")
- All AVDD and AVSs pins, regardless of whether or not the analog device features are used (see Section 2.2 "Power Supply Pins")
- MCLR pin (see Section 2.3 "Master Clear (MCLR) Pin")
- ENVREG/DISVREG and VCAP/VDDCORE pins (see Section 2.4 "Voltage Regulator Pins (ENVREG/DISVREG and VCAP/VDDCORE)")

These pins must also be connected if they are being used in the end application:

- PGECx/PGEDx pins used for In-Circuit Serial Programming™ (ICSP™) and debugging purposes (see Section 2.5 "ICSP Pins")
- OSCI and OSCO pins when an external oscillator source is used (see Section 2.6 "External Oscillator Pins")

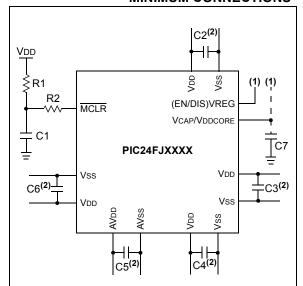
Additionally, the following pins may be required:

 VREF+/VREF- pins used when external voltage reference for analog modules is implemented

Note: The AVDD and AVSS pins must always be connected, regardless of whether any of the analog modules are being used.

The minimum mandatory connections are shown in Figure 2-1.

FIGURE 2-1: RECOMMENDED MINIMUM CONNECTIONS



Key (all values are recommendations):

C1 through C6: 0.1 µF, 20V ceramic

C7: 10 µF, 6.3V or greater, tantalum or ceramic

R1: 10 kΩ

R2: 100Ω to 470Ω

- Note 1: See Section 2.4 "Voltage Regulator Pins (ENVREG/DISVREG and VCAP/VDDCORE)" for the explanation of the ENVREG/DISVREG pin connections.
 - 2: The example shown is for a PIC24F device with five VDD/Vss and AVDD/AVss pairs. Other devices may have more or less pairs; adjust the number of decoupling capacitors appropriately.

2.2 Power Supply Pins

2.2.1 DECOUPLING CAPACITORS

The use of decoupling capacitors on every pair of power supply pins, such as VDD, VSS, AVDD and AVSs is required.

Consider the following criteria when using decoupling capacitors:

- Value and type of capacitor: A 0.1 μ F (100 nF), 10-20V capacitor is recommended. The capacitor should be a low-ESR device with a resonance frequency in the range of 200 MHz and higher. Ceramic capacitors are recommended.
- Placement on the printed circuit board: The
 decoupling capacitors should be placed as close
 to the pins as possible. It is recommended to
 place the capacitors 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 no greater
 than 0.25 inch (6 mm).
- 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 each 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 (e.g., 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 trace inductance.

2.2.2 TANK CAPACITORS

On boards with power traces running longer than six inches in length, it is suggested to use a tank capacitor for integrated circuits including microcontrollers to supply a local power source. The value of the tank capacitor should be determined based on the trace resistance that connects the power supply source to the device, and the maximum current drawn by the device in the application. In other words, select the tank capacitor so that it meets the acceptable voltage sag at the device. Typical values range from 4.7 μF to 47 μF .

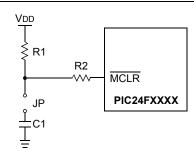
2.3 Master Clear (MCLR) Pin

The MCLR pin provides two specific device functions: device Reset, and device programming and debugging. If programming and debugging are not required in the end application, a direct connection to VDD may be all that is required. The addition of other components, to help increase the application's resistance to spurious Resets from voltage sags, may be beneficial. A typical configuration is shown in Figure 2-1. Other circuit designs may be implemented, depending on the application's requirements.

During 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 R1 and C1 will need to be adjusted based on the application and PCB requirements. For example, it is recommended that the capacitor, C1, be isolated from the MCLR pin during programming and debugging operations by using a jumper (Figure 2-2). The jumper is replaced for normal run-time operations.

Any components associated with the $\overline{\text{MCLR}}$ pin should be placed within 0.25 inch (6 mm) of the pin.

FIGURE 2-2: EXAMPLE OF MCLR PIN CONNECTIONS



- Note 1: R1 \leq 10 k Ω is recommended. A suggested starting value is 10 k Ω . Ensure that the MCLR pin VIH and VIL specifications are met.
 - 2: $R2 \le 470\Omega$ will limit any current flowing into \overline{MCLR} from the external capacitor, C, in the event of \overline{MCLR} pin breakdown due to Electrostatic Discharge (ESD) or Electrical Overstress (EOS). Ensure that the \overline{MCLR} pin VIH and VIL specifications are met.

2.4 Voltage Regulator Pins (ENVREG/DISVREG and VCAP/VDDCORE)

Note: This section applies only to PIC24FJ devices with an on-chip voltage regulator.

The on-chip voltage regulator enable/disable pin (ENVREG or DISVREG, depending on the device family) must always be connected directly to either a supply voltage or to ground. The particular connection is determined by whether or not the regulator is to be used:

- For ENVREG, tie to VDD to enable the regulator, or to ground to disable the regulator
- For DISVREG, tie to ground to enable the regulator or to VDD to disable the regulator

Refer to Section 30.2 "On-Chip Voltage Regulator" for details on connecting and using the on-chip regulator.

When the regulator is enabled, a low-ESR ($< 5\Omega$) capacitor is required on the VCAP/VDDCORE pin to stabilize the voltage regulator output voltage. The VCAP/VDDCORE pin must not be connected to VDD and must use a capacitor of 10 μ F connected to ground. The type can be ceramic or tantalum. Suitable examples of capacitors are shown in Table 2-1. Capacitors with equivalent specification can be used.

Designers may use Figure 2-3 to evaluate ESR equivalence of candidate devices.

The placement of this capacitor should be close to VCAP/VDDCORE. It is recommended that the trace length not exceed 0.25 inch (6 mm). Refer to Section 33.0 "Electrical Characteristics" for additional information.

When the regulator is disabled, the VCAP/VDDCORE pin must be tied to a voltage supply at the VDDCORE level. Refer to Section 33.0 "Electrical Characteristics" for information on VDD and VDDCORE.

FIGURE 2-3: FREQUENCY vs. ESR
PERFORMANCE FOR
SUGGESTED VCAP

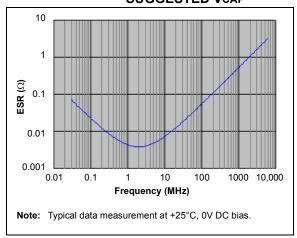


TABLE 2-1: SUITABLE CAPACITOR EQUIVALENTS

Make	Part #	Nominal Capacitance	Base Tolerance	Rated Voltage	Temp. Range
TDK	C3216X7R1C106K	10 μF	±10%	16V	-55 to +125°C
TDK	C3216X5R1C106K	10 μF	±10%	16V	-55 to +85°C
Panasonic	ECJ-3YX1C106K	10 μF	±10%	16V	-55 to +125°C
Panasonic	ECJ-4YB1C106K	10 μF	±10%	16V	-55 to +85°C
Murata	GRM32DR71C106KA01L	10 μF	±10%	16V	-55 to +125°C
Murata	GRM31CR61C106KC31L	10 μF	±10%	16V	-55 to +85°C

2.4.1 CONSIDERATIONS FOR CERAMIC CAPACITORS

In recent years, large value, low-voltage, surface-mount ceramic capacitors have become very cost effective in sizes up to a few tens of microfarad. The low-ESR, small physical size and other properties make ceramic capacitors very attractive in many types of applications.

Ceramic capacitors are suitable for use with the internal voltage regulator of this microcontroller. However, some care is needed in selecting the capacitor to ensure that it maintains sufficient capacitance over the intended operating range of the application.

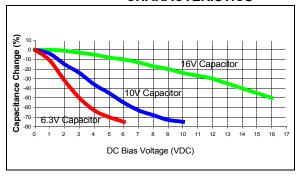
Typical low-cost, 10 μ F ceramic capacitors are available in X5R, X7R and Y5V dielectric ratings (other types are also available, but are less common). The initial tolerance specifications for these types of capacitors are often specified as $\pm 10\%$ to $\pm 20\%$ (X5R and X7R), or -20%/+80% (Y5V). However, the effective capacitance that these capacitors provide in an application circuit will also vary based on additional factors, such as the applied DC bias voltage and the temperature. The total in-circuit tolerance is, therefore, much wider than the initial tolerance specification.

The X5R and X7R capacitors typically exhibit satisfactory temperature stability (ex: $\pm 15\%$ over a wide temperature range, but consult the manufacturer's data sheets for exact specifications). However, Y5V capacitors typically have extreme temperature tolerance specifications of +22%/-82%. Due to the extreme temperature tolerance, a 10 μF nominal rated Y5V type capacitor may not deliver enough total capacitance to meet minimum internal voltage regulator stability and transient response requirements. Therefore, Y5V capacitors are not recommended for use with the internal regulator if the application must operate over a wide temperature range.

In addition to temperature tolerance, the effective capacitance of large value ceramic capacitors can vary substantially, based on the amount of DC voltage applied to the capacitor. This effect can be very significant, but is often overlooked or is not always documented.

Typical DC bias voltage vs. capacitance graph for X7R type capacitors is shown in Figure 2-4.

FIGURE 2-4: DC BIAS VOLTAGE vs. CAPACITANCE CHARACTERISTICS



When selecting a ceramic capacitor to be used with the internal voltage regulator, it is suggested to select a high-voltage rating, so that the operating voltage is a small percentage of the maximum rated capacitor voltage. For example, choose a ceramic capacitor rated at 16V for the 2.5V or 1.8V core voltage. Suggested capacitors are shown in Table 2-1.

2.5 ICSP Pins

The PGECx and PGEDx pins are used for In-Circuit Serial Programming (ICSP) and debugging purposes. It is recommended to keep the trace length between the ICSP connector and the ICSP pins on the device as short as possible. If the ICSP connector is expected to experience an ESD event, a series resistor is recommended, with the value in the range of a few tens of ohms, not to exceed 100Ω .

Pull-up resistors, series diodes and capacitors on the PGECx and PGEDx pins are not recommended as they will interfere with the programmer/debugger communications to the device. If such discrete components are an application requirement, they should be removed from the circuit during programming and debugging. Alternatively, refer to the AC/DC characteristics and timing requirements information in the respective device Flash programming specification for information on capacitive loading limits and pin input voltage high (VIH) and input low (VIL) requirements.

For device emulation, ensure that the "Communication Channel Select" (i.e., PGECx/PGEDx pins), programmed into the device, matches the physical connections for the ICSP to the Microchip debugger/emulator tool.

For more information on available Microchip development tools connection requirements, refer to **Section 31.0 "Development Support"**.

2.6 External Oscillator Pins

Many microcontrollers have options for at least two oscillators: a high-frequency Primary Oscillator and a low-frequency Secondary Oscillator (refer to Section 9.0 "Oscillator Configuration" for details).

The oscillator circuit should be placed on the same side of the board as the device. Place the oscillator circuit close to the respective oscillator pins with no more than 0.5 inch (12 mm) between the circuit components and the pins. The load capacitors should be placed next to the oscillator itself, on the same side of the board.

Use a grounded copper pour around the oscillator circuit to isolate it from surrounding circuits. The grounded copper pour should be routed directly to the MCU ground. Do not run any signal traces or power traces inside the ground pour. Also, if using a two-sided board, avoid any traces on the other side of the board where the crystal is placed.

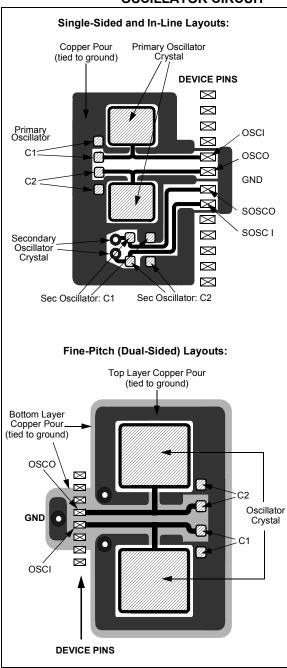
Layout suggestions are shown in Figure 2-5. In-line packages may be handled with a single-sided layout that completely encompasses the oscillator pins. With fine-pitch packages, it is not always possible to completely surround the pins and components. A suitable solution is to tie the broken guard sections to a mirrored ground layer. In all cases, the guard trace(s) must be returned to ground.

In planning the application's routing and I/O assignments, ensure that adjacent port pins, and other signals in close proximity to the oscillator, are benign (i.e., free of high frequencies, short rise and fall times and other similar noise).

For additional information and design guidance on oscillator circuits, please refer to these Microchip Application Notes, available at the corporate web site (www.microchip.com):

- AN826, "Crystal Oscillator Basics and Crystal Selection for rfPIC™ and PICmicro[®] Devices"
- AN849, "Basic PICmicro[®] Oscillator Design"
- AN943, "Practical PICmicro[®] Oscillator Analysis and Design"
- · AN949, "Making Your Oscillator Work"

FIGURE 2-5: SUGGESTED PLACEMENT OF THE OSCILLATOR CIRCUIT



2.7 Configuration of Analog and Digital Pins During ICSP Operations

If an ICSP compliant emulator is selected as a debugger, it automatically initializes all of the A/D input pins (ANx) as "digital" pins. Depending on the particular device, this is done by setting all bits in the ADxPCFG register(s) or clearing all bits in the ANSx registers.

All PIC24F devices will have either one or more ADxPCFG registers, or several ANSx registers (one for each port); no device will have both. Refer to Section 11.2 "Configuring Analog Port Pins (ANSx)" for more specific information.

The bits in these registers that correspond to the A/D pins that initialized the emulator must not be changed by the user application firmware; otherwise, communication errors will result between the debugger and the device.

If your application needs to use certain A/D pins as analog input pins during the debug session, the user application must modify the appropriate bits during initialization of the A/D module, as follows:

- For devices with an ADxPCFG register, clear the bits corresponding to the pin(s) to be configured as analog. Do not change any other bits, particularly those corresponding to the PGECx/PGEDx pair, at any time.
- For devices with ANSx registers, set the bits corresponding to the pin(s) to be configured as analog. Do not change any other bits, particularly those corresponding to the PGECx/PGEDx pair, at any time.

When a Microchip debugger/emulator is used as a programmer, the user application firmware must correctly configure the ADxPCFG or ANSx registers. Automatic initialization of this register is only done during debugger operation. Failure to correctly configure the register(s) will result in all A/D pins being recognized as analog input pins, resulting in the port value being read as a logic '0', which may affect user application functionality.

2.8 Unused I/Os

Unused I/O pins should be configured as outputs and driven to a logic low state. Alternatively, connect a 1 k Ω to 10 k Ω resistor to Vss on unused pins and drive the output to logic low.

3.0 CPU

Note:

This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information on the CPU, refer to the "dsPIC33/PIC24 Family Reference Manual", "CPU with Extended Data Space (EDS)" (DS39732). The information in this data sheet supersedes the information in the FRM.

The PIC24F CPU has a 16-bit (data) modified Harvard architecture with an enhanced instruction set and a 24-bit instruction word with a variable length opcode field. The Program Counter (PC) is 23 bits wide and addresses up to 4M instructions of user program memory space. A single-cycle instruction prefetch mechanism is used to help maintain throughput and provides predictable execution. All instructions execute in a single cycle, with the exception of instructions that change the program flow, the double-word move (MOV.D) instruction and the table instructions. Overhead-free program loop constructs are supported using the REPEAT instructions, which are interruptible at any point.

PIC24F devices have sixteen, 16-bit Working registers in the programmer's model. Each of the Working registers can act as a Data, Address or Address Offset register. The 16th Working register (W15) operates as a Software Stack Pointer (SSP) for interrupts and calls.

The lower 32 Kbytes of the Data Space (DS) can be accessed linearly. The upper 32 Kbytes of the Data Space are referred to as Extended Data Space to which the extended data RAM, EPMP memory space or program memory can be mapped.

The Instruction Set Architecture (ISA) has been significantly enhanced beyond that of the PIC18, but maintains an acceptable level of backward compatibility. All PIC18 instructions and addressing modes are supported, either directly, or through simple macros. Many of the ISA enhancements have been driven by compiler efficiency needs.

The core supports Inherent (no operand), Relative, Literal and Memory Direct Addressing modes, along with three groups of addressing modes. All modes support Register Direct and various Register Indirect modes. Each group offers up to seven addressing modes. Instructions are associated with predefined addressing modes depending upon their functional requirements.

For most instructions, the core is capable of executing a data (or program data) memory read, a Working register (data) read, a data memory write and a program (instruction) memory read per instruction cycle. As a result, three parameter instructions can be supported, allowing trinary operations (that is, A + B = C) to be executed in a single cycle.

A high-speed, 17-bit x 17-bit multiplier has been included to significantly enhance the core arithmetic capability and throughput. The multiplier supports Signed, Unsigned and Mixed mode, 16-bit x 16-bit or 8-bit x 8-bit, integer multiplication. All multiply instructions execute in a single cycle.

The 16-bit ALU has been enhanced with integer divide assist hardware that supports an iterative non-restoring divide algorithm. It operates in conjunction with the REPEAT instruction looping mechanism and a selection of iterative divide instructions to support 32-bit (or 16-bit), divided by 16-bit, integer signed and unsigned division. All divide operations require 19 cycles to complete but are interruptible at any cycle boundary.

The PIC24F has a vectored exception scheme with up to 8 sources of non-maskable traps and up to 118 interrupt sources. Each interrupt source can be assigned to one of seven priority levels.

A block diagram of the CPU is shown in Figure 3-1.

3.1 Programmer's Model

The programmer's model for the PIC24F is shown in Figure 3-2. All registers in the programmer's model are memory-mapped and can be manipulated directly by instructions.

A description of each register is provided in Table 3-1. All registers associated with the programmer's model are memory-mapped.

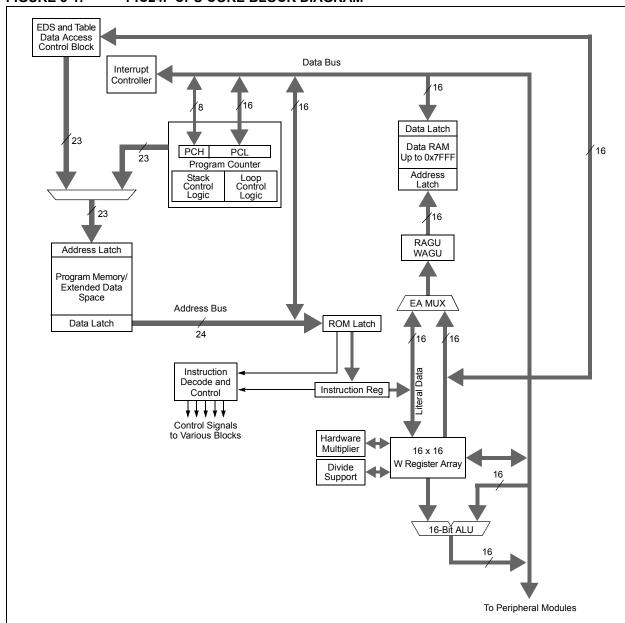


FIGURE 3-1: PIC24F CPU CORE BLOCK DIAGRAM

TABLE 3-1: CPU CORE REGISTERS

Register(s) Name	Description	
W0 through W15	Working Register Array	
PC	23-Bit Program Counter	
SR	ALU STATUS Register	
SPLIM	Stack Pointer Limit Value Register	
TBLPAG	Table Memory Page Address Register	
RCOUNT	REPEAT Loop Counter Register	
CORCON	CPU Control Register	
DISICNT	Disable Interrupt Count Register	
DSRPAG	Data Space Read Page Register	
DSWPAG	Data Space Write Page Register	

FIGURE 3-2: PROGRAMMER'S MODEL 0 W0 (WREG) **Divider Working Registers** W1 W2 Multiplier Registers W3 W4 W5 W6 W7 Working/Address Registers W8 W9 W10 W11 W12 W13 Frame Pointer W14 W15 Stack Pointer Stack Pointer Limit Value Register SPLIM 0 РС 0 **Program Counter** Table Memory Page Address Register **TBLPAG** DSRPAG Data Space Read Page Register DSWPAG Data Space Write Page Register 15 REPEAT Loop Counter RCOUNT Register SRH SRL 0 15 RA N OV Z ALU STATUS Register (SR) CPU Control Register (CORCON) DISICNT Disable Interrupt Count Register Registers or bits are shadowed for PUSH.S and POP.S instructions.

3.2 CPU Control Registers

REGISTER 3-1: SR: ALU STATUS REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
_	_	_	_	_	_	_	DC
bit 15							bit 8

R/W-0 ⁽¹⁾	R/W-0 ⁽¹⁾	R/W-0 ⁽¹⁾	R-0	R/W-0	R/W-0	R/W-0	R/W-0
IPL2 ⁽²⁾	IPL1 ⁽²⁾	IPL0 ⁽²⁾	RA	N	OV	Z	С
bit 7							bit 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 15-9 **Unimplemented:** Read as '0'

bit 8 DC: ALU Half Carry/Borrow bit

1 = A carry out from the 4th low-order bit (for byte-sized data) or 8th low-order bit (for word-sized data) of the result occurred

0 = No carry out from the 4th or 8th low-order bit of the result has occurred

bit 7-5 **IPL<2:0>:** CPU Interrupt Priority Level Status bits^(1,2)

111 = CPU Interrupt Priority Level is 7 (15); user interrupts are disabled

110 = CPU Interrupt Priority Level is 6 (14)

101 = CPU Interrupt Priority Level is 5 (13)

100 = CPU Interrupt Priority Level is 4 (12)

011 = CPU Interrupt Priority Level is 3 (11)

010 = CPU Interrupt Priority Level is 2 (10)

001 = CPU Interrupt Priority Level is 1 (9)

000 = CPU Interrupt Priority Level is 0 (8)

bit 4 RA: REPEAT Loop Active bit

1 = REPEAT loop is in progress

0 = REPEAT loop is not in progress

bit 3 N: ALU Negative bit

1 = Result was negative

0 = Result was not negative (zero or positive)

bit 2 **OV:** ALU Overflow bit

1 = Overflow occurred for signed (2's complement) arithmetic in this arithmetic operation

0 = No overflow has occurred

bit 1 **Z**: ALU Zero bit

1 = An operation, which affects the Z bit, has set it at some time in the past

0 = The most recent operation, which affects the Z bit, has cleared it (i.e., a non-zero result)

bit 0 C: ALU Carry/Borrow bit

1 = A carry out from the Most Significant bit (MSb) of the result occurred

0 = No carry out from the Most Significant bit of the result occurred

Note 1: The IPLx Status bits are read-only when NSTDIS (INTCON1<15>) = 1.

2: The IPLx Status bits are concatenated with the IPL3 Status (CORCON<3>) bit to form the CPU Interrupt Priority Level (IPL). The value in parentheses indicates the IPL when IPL3 = 1.

REGISTER 3-2: CORCON: CPU CORE CONTROL REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

U-0	U-0	U-0	U-0	R/C-0	r-1	U-0	U-0
_	_	_	_	IPL3 ⁽¹⁾	_	_	_
bit 7							bit 0

Legend: C = Clearable bit		r = Reserved bit	r = Reserved bit		
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 15-4 Unimplemented: Read as '0'

bit 3 IPL3: CPU Interrupt Priority Level Status bit(1)

1 = CPU Interrupt Priority Level is greater than 7

0 = CPU Interrupt Priority Level is 7 or less

bit 2 Reserved: Read as '1'

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

Note 1: The IPL3 bit is concatenated with the IPL<2:0> bits (SR<7:5>) to form the CPU Interrupt Priority Level; see Register 3-1 for bit description.

3.3 Arithmetic Logic Unit (ALU)

The PIC24F ALU is 16 bits wide and is capable of addition, subtraction, bit shifts and logic operations. Unless otherwise mentioned, arithmetic operations are 2's complement in nature. Depending on the operation, the ALU may affect the values of the Carry (C), Zero (Z), Negative (N), Overflow (OV) and Digit Carry (DC) Status bits in the SR register. The C and DC Status bits operate as $\overline{\text{Borrow}}$ and $\overline{\text{Digit}}$ Borrow bits, respectively, for subtraction operations.

The ALU can perform 8-bit or 16-bit operations, depending on the mode of the instruction that is used. Data for the ALU operation can come from the W register array, or data memory, depending on the addressing mode of the instruction. Likewise, output data from the ALU can be written to the W register array or a data memory location.

The PIC24F CPU incorporates hardware support for both multiplication and division. This includes a dedicated hardware multiplier and support hardware for 16-bit divisor division.

3.3.1 MULTIPLIER

The ALU contains a high-speed, 17-bit x 17-bit multiplier. It supports unsigned, signed or mixed sign operation in several multiplication modes:

- 16-bit x 16-bit signed
- 16-bit x 16-bit unsigned
- 16-bit signed x 5-bit (literal) unsigned
- 16-bit unsigned x 16-bit unsigned
- 16-bit unsigned x 5-bit (literal) unsigned
- · 16-bit unsigned x 16-bit signed
- · 8-bit unsigned x 8-bit unsigned

3.3.2 DIVIDER

The divide block supports 32-bit/16-bit and 16-bit/16-bit signed and unsigned integer divide operations with the following data sizes:

- 1. 32-bit signed/16-bit signed divide
- 2. 32-bit unsigned/16-bit unsigned divide
- 3. 16-bit signed/16-bit signed divide
- 4. 16-bit unsigned/16-bit unsigned divide

The quotient for all divide instructions ends up in W0 and the remainder in W1. The 16-bit signed and unsigned DIV instructions can specify any W register for both the 16-bit divisor (Wn), and any W register (aligned) pair (W(m + 1):Wm) for the 32-bit dividend. The divide algorithm takes one cycle per bit of divisor, so both 32-bit/16-bit and 16-bit/16-bit instructions take the same number of cycles to execute.

3.3.3 MULTI-BIT SHIFT SUPPORT

The PIC24F ALU supports both single bit and single-cycle, multi-bit arithmetic and logic shifts. Multi-bit shifts are implemented using a shifter block, capable of performing up to a 15-bit arithmetic right shift, or up to a 15-bit left shift, in a single cycle. All multi-bit shift instructions only support Register Direct Addressing for both the operand source and result destination.

A full summary of instructions that use the shift operation is provided in Table 3-2.

TABLE 3-2: INSTRUCTIONS THAT USE THE SINGLE BIT AND MULTI-BIT SHIFT OPERATION

Instruction	Description
ASR	Arithmetic Shift Right Source register by one or more bits.
SL	Shift Left Source register by one or more bits.
LSR	Logical Shift Right Source register by one or more bits.

4.0 MEMORY ORGANIZATION

As Harvard architecture devices, PIC24F microcontrollers feature separate program and data memory spaces and buses. This architecture also allows direct access of program memory from the Data Space (DS) during code execution.

4.1 Program Memory Space

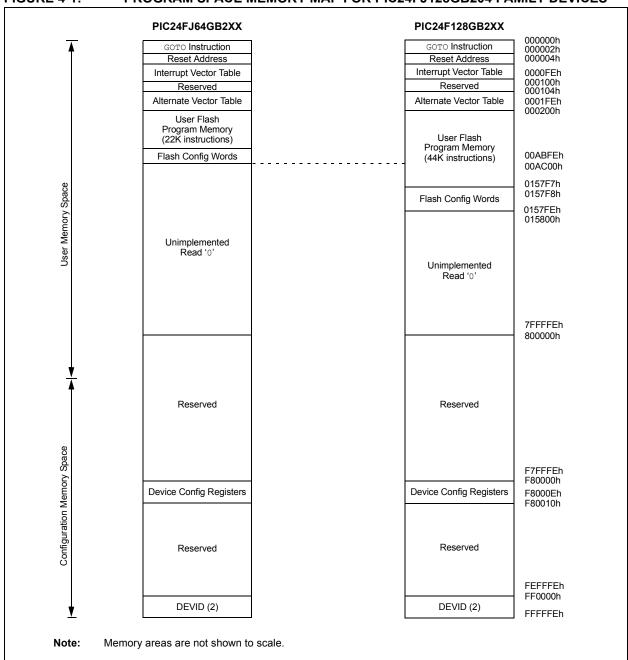
The program address memory space of the PIC24FJ128GB204 family devices is 4M instructions. The space is addressable by a 24-bit value derived

from either the 23-bit Program Counter (PC) during program execution, or from table operation or Data Space remapping, as described in **Section 4.3 "Interfacing Program and Data Memory Spaces"**.

User access to the program memory space is restricted to the lower half of the address range (000000h to 7FFFFFh). The exception is the use of ${\tt TBLRD/TBLWT}$ operations, which use TBLPAG<7> to permit access to the Configuration bits and Device ID sections of the configuration memory space.

Memory maps for the PIC24FJ128GB204 family of devices are shown in Figure 4-1.

FIGURE 4-1: PROGRAM SPACE MEMORY MAP FOR PIC24FJ128GB204 FAMILY DEVICES



4.1.1 PROGRAM MEMORY ORGANIZATION

The program memory space is organized in word-addressable blocks. Although it is treated as 24 bits wide, it is more appropriate to think of each address of the program memory as a lower and upper word, with the upper byte of the upper word being unimplemented. The lower word always has an even address, while the upper word has an odd address (Figure 4-2).

Program memory addresses are always word-aligned on the lower word and addresses are incremented or decremented by two during code execution. This arrangement also provides compatibility with data memory space addressing and makes it possible to access data in the program memory space.

4.1.2 HARD MEMORY VECTORS

All PIC24F devices reserve the addresses between 000000h and 000200h for hard-coded program execution vectors. A hardware Reset vector is provided to redirect code execution from the default value of the PC on device Reset to the actual start of code. A GOTO instruction is programmed by the user at 000000h with the actual address for the start of code at 000002h.

PIC24F devices also have two Interrupt Vector Tables (IVTs), located from 000004h to 0000FFh and 000100h to 0001FFh. These vector tables allow each of the many device interrupt sources to be handled by separate ISRs. A more detailed discussion of the Interrupt Vector Tables is provided in Section 8.1 "Interrupt Vector Table".

4.1.3 FLASH CONFIGURATION WORDS

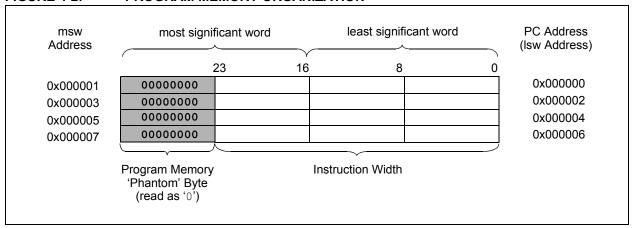
In PIC24FJ128GB204 family devices, the top four words of on-chip program memory are reserved for configuration information. On device Reset, the configuration information is copied into the appropriate Configuration register. The addresses of the Flash Configuration Word for devices in the PIC24FJ128GB204 family are shown in Table 4-1. Their location in the memory map is shown with the other memory vectors in Figure 4-1.

The Configuration Words in program memory are a compact format. The actual Configuration bits are mapped in several different registers in the configuration memory space. Their order in the Flash Configuration Words does not reflect a corresponding arrangement in the configuration space. Additional details on the device Configuration Words are provided in Section 30.0 "Special Features".

TABLE 4-1: FLASH CONFIGURATION WORDS FOR PIC24FJ128GB204 FAMILY DEVICES

Device	Program Memory (Words)	Configuration Word Addresses
PIC24FJ64GB2XX	22,016	00ABF8h:00ABFEh
PIC24FJ128GB2XX	44,032	0157F8h:0157FEh

FIGURE 4-2: PROGRAM MEMORY ORGANIZATION



4.2 Data Memory Space

Note:

This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the "dsPIC33/PIC24 Family Reference Manual", "Data Memory with Extended Data Space (EDS)" (DS39733). The information in this data sheet supersedes the information in the FRM.

The PIC24F core has a 16-bit wide data memory space, addressable as a single linear range. The Data Space (DS) is accessed using two Address Generation Units (AGUs), one each for read and write operations. The Data Space memory map is shown in Figure 4-3.

The 16-bit wide data addresses in the data memory space point to bytes within the Data Space. This gives a DS address range of 64 Kbytes or 32K words. The lower half (0000h to 7FFFh) is used for implemented (on-chip) memory addresses.

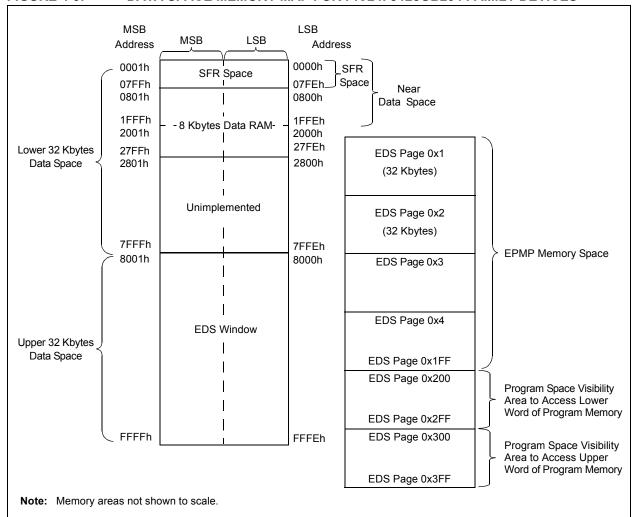
The upper half of data memory address space (8000h to FFFFh) is used as a window into the Extended Data Space (EDS). This allows the microcontroller to directly access a greater range of data beyond the standard 16-bit address range. EDS is discussed in detail in Section 4.2.5 "Extended Data Space (EDS)".

The lower half of DS is compatible with previous PIC24F microcontrollers without EDS. All PIC24FJ128GB204 family devices implement 8 Kbytes of data RAM in the lower half of DS, from 0800h to 27FFh.

4.2.1 DATA SPACE WIDTH

The data memory space is organized in byte-addressable, 16-bit wide blocks. Data is aligned in data memory and registers as 16-bit words, but all Data Space Effective Addresses (EAs) resolve to bytes. The Least Significant Bytes (LSBs) of each word have even addresses, while the Most Significant Bytes (MSBs) have odd addresses.

FIGURE 4-3: DATA SPACE MEMORY MAP FOR PIC24FJ128GB204 FAMILY DEVICES



4.2.2 DATA MEMORY ORGANIZATION AND ALIGNMENT

To maintain backward compatibility with PIC® MCUs and improve Data Space memory usage efficiency, the PIC24F instruction set supports both word and byte operations. As a consequence of byte accessibility, all Effective Address (EA) calculations are internally scaled to step through word-aligned memory. For example, the core recognizes that Post-Modified Register Indirect Addressing mode [Ws++] will result in a value of Ws + 1 for byte operations and Ws + 2 for word operations.

Data byte reads will read the complete word, which contains the byte, using the LSB of any EA to determine which byte to select. The selected byte is placed onto the LSB of the data path. That is, data memory and registers are organized as two parallel, byte-wide entities with shared (word) address decode but separate write lines. Data byte writes only write to the corresponding side of the array or register which matches the byte address.

All word accesses must be aligned to an even address. Misaligned word data fetches are not supported, so care must be taken when mixing byte and word operations or translating from 8-bit MCU code. If a misaligned read or write is attempted, an address error trap will be generated. If the error occurred on a read, the instruction underway is completed; if it occurred on a write, the instruction will be executed but the write will not occur. In either case, a trap is then executed, allowing the system and/or user to examine the machine state prior to execution of the address Fault.

All byte loads into any W register are loaded into the LSB. The Most Significant Byte (MSB) is not modified.

A Sign-Extend (SE) instruction is provided to allow users to translate 8-bit signed data to 16-bit signed values. Alternatively, for 16-bit unsigned data, users can clear the MSB of any W register by executing a Zero-Extend (ZE) instruction on the appropriate address.

Although most instructions are capable of operating on word or byte data sizes, it should be noted that some instructions operate only on words.

4.2.3 NEAR DATA SPACE

The 8-Kbyte area, between 0000h and 1FFFh, is referred to as the Near Data Space. Locations in this space are directly addressable via a 13-bit absolute address field within all memory direct instructions. The remainder of the Data Space is addressable indirectly. Additionally, the whole Data Space is addressable using ${\tt MOV}$ instructions, which support Memory Direct Addressing with a 16-bit address field.

4.2.4 SPECIAL FUNCTION REGISTER (SFR) SPACE

The first 2 Kbytes of the Near Data Space, from 0000h to 07FFh, are primarily occupied with Special Function Registers (SFRs). These are used by the PIC24F core and peripheral modules for controlling the operation of the device.

SFRs are distributed among the modules that they control and are generally grouped together by the module. Much of the SFR space contains unused addresses; these are read as '0'. A diagram of the SFR space, showing where the SFRs are actually implemented, is shown in Table 4-2. Each implemented area indicates a 32-byte region where at least one address is implemented as an SFR. A complete list of implemented SFRs, including their addresses, is shown in Table 4-3 through Table 4-33.

TABLE 4-2: IMPLEMENTED REGIONS OF SFR DATA SPACE

					S	FR Space	e Add	ress						
	xx(00	XX	20	хх	40	xx	60	х	x80	xxA) ;	cxC0	xxE0
000h			Cor	е			IC	N			In	terrupts	;	
100h	System	NVM/F	RTCC	Р	MP	CR	С	PM	1D	I/O		C	rypto	
200h		A/D/CT	MU		CMP	TM	R		C	C		IC	l ² C	C™/DSM
300h			SPI							PF	PS			
400h			USB							D۱	ΛA			
500h		UAR	T.							_				
600h							_	-						
700h							_							

Legend: — = No implemented SFRs in this block

TABLE 4-3: CPU CORE REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
WREG0	0000								Working F	Register 0								0000
WREG1	0002								Working F	Register 1								0000
WREG2	0004								Working F	Register 2								0000
WREG3	0006								Working F	Register 3								0000
WREG4	8000								Working F	Register 4								0000
WREG5	000A								Working F	Register 5								0000
WREG6	000C								Working F	Register 6								0000
WREG7	000E				Working Register 7 Working Register 8													
WREG8	0010																	
WREG9	0012				Working Register 9													
WREG10	0014																	
WREG11	0016								Working F	Register 11								0000
WREG12	0018								Working F	Register 12								0000
WREG13	001A								Working R	Register 13								0000
WREG14	001C								Working R	Register 14								0000
WREG15	001E								Working F	Register 15								0800
SPLIM	0020							Stack	Pointer Lin	nit Value Re	egister							xxxx
PCL	002E							Progra	m Counter I	Low Word F	Register							0000
PCH	0030	_	_	_	_	_	_	_	_			Progra	m Counter I	High Word	Register			0000
DSRPAG	0032	_	_	_	_	_	_			Ext	ended Data	Space Re	ad Page Ac	ldress Regi	ster			0001
DSWPAG	0034	_	_	_	_	_	_	_			Extended	d Data Spa	ce Write Pa	ge Address	Register			0001
RCOUNT	0036							REP:	EAT Loop C	Counter Reg	gister							xxxx
SR	0042	_	_	_	_	_	_	_	DC	IPL2	IPL1	IPL0	RA	N	OV	Z	С	0000
CORCON	0044	_	_	_														0004
DISICNT	0052	_	_						Disabl	e Interrupts	Counter R	egister						xxxx
TBLPAG	0054	_	_	_	_	_	_	_	_			Table M	lemory Pag	e Address	Register			0000

Legend: — = unimplemented, read as '0'; r = reserved bit, do not modify; x = unknown value on Reset. Reset values are shown in hexadecimal.

TABLE 4-4: ICN REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
CNPD1	0056	CN15PDE	_	CN13PDE	CN12PDE	CN11PDE	CN10PDE ⁽¹⁾	CN9PDE(1)	CN8PDE(1)	CN7PDE	CN6PDE	CN5PDE	CN4PDE	CN3PDE	CN2PDE	CN1PDE	CN0PDE	0000
CNPD2	0058	-	CN30PDE	CN29PDE	CN28PDE(1)	CN27PDE	CN26PDE(1)	CN25PDE ⁽¹⁾	CN24PDE	CN23PDE	CN22PDE	CN21PDE	CN20PDE(1)	CN19PDE(1)	CN18PDE(1)	CN17PDE(1)	CN16PDE	0000
CNPD3	005A	-	-	-	-	_	-	_	_	_	_	_	CN36PDE(1)	CN35PDE(1)	CN34PDE(1)	CN33PDE(1)	_	0000
CNEN1	0062	CN15IE	-	CN13IE	CN12IE	CN11IE	CN10IE ⁽¹⁾	CN9IE(1)	CN8IE ⁽¹⁾	CN7IE	CN6IE	CN5IE	CN4IE	CN3IE	CN2IE	CN1IE	CN0IE	0000
CNEN2	0064	-	CN30IE	CN29IE	CN28IE ⁽¹⁾	CN27IE	CN26IE ⁽¹⁾	CN25IE ⁽¹⁾	CN24IE	CN23IE	CN22IE	CN21IE	CN20IE ⁽¹⁾	CN19IE ⁽¹⁾	CN18IE ⁽¹⁾	CN17IE ⁽¹⁾	CN16IE	0000
CNEN3	0066	-	-	-	-	_	-	_	_	_	_	_	CN36IE ⁽¹⁾	CN35IE ⁽¹⁾	CN34IE ⁽¹⁾	CN33IE ⁽¹⁾	_	0000
CNPU1	006E	CN15PUE	-	CN13PUE	CN12PUE	CN11PUE	CN10PUE ⁽¹⁾	CN9PUE(1)	CN8PUE ⁽¹⁾	CN7PUE	CN6PUE	CN5PUE	CN4PUE	CN3PUE	CN2PUE	CN1PUE	CN0PUE	0000
CNPU2	0070	_	CN30PUE	CN29PUE	CN28PUE ⁽¹⁾	CN27PUE	CN26PUE ⁽¹⁾	CN25PUE ⁽¹⁾	CN24PUE	CN23PUE	CN22PUE	CN21PUE	CN20PUE(1)	CN19PUE(1)	CN18PUE(1)	CN17PUE(1)	CN16PUE	0000
CNPU3	0072	_	_	_	_	_	_	_	_	_	_	_	CN36PUE ⁽¹⁾	CN35PUE ⁽¹⁾	CN34PUE ⁽¹⁾	CN33PUE ⁽¹⁾	_	0000

Legend: — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

Note 1: These bits are unimplemented in 28-pin devices, read as '0'.

TABLE 4-5: INTERRUPT CONTROLLER REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
INTCON1	0800	NSTDIS	_	_	_	_	_	_	-	_	_	_	MATHERR	ADDRERR	STKERR	OSCFAIL	_	0000
INTCON2	0082	ALTIVT	DISI	_	_	_	_	_	_	_	_	_	INT4EP	INT3EP	INT2EP	INT1EP	INT0EP	0000
IFS0	0084	_	DMA1IF	AD1IF	U1TXIF	U1RXIF	SPI1TXIF	SPI1IF	T3IF	T2IF	OC2IF	IC2IF	DMA0IF	T1IF	OC1IF	IC1IF	INT0IF	0000
IFS1	0086	U2TXIF	U2RXIF	INT2IF	T5IF	T4IF	OC4IF	OC3IF	DMA2IF	1	_	1	INT1IF	CNIF	CMIF	MI2C1IF	SI2C1IF	0000
IFS2	0088	-	DMA4IF	PMPIF	ı	_	OC6IF	OC5IF	IC6IF	IC5IF	IC4IF	IC3IF	DMA3IF	CRYROLLIF	CRYFREEIF	SPI2TXIF	SPI2IF	0000
IFS3	008A	_	RTCIF	DMA5IF	SPI3RXIF	SPI2RXIF	SPI1RXIF	_	KEYSTRIF	CRYDNIF	INT4IF	INT3IF	_	-	MI2C2IF	SI2C2IF	_	0000
IFS4	008C	_	_	CTMUIF	_	_	_	_	HLVDIF	_	_	_	_	CRCIF	U2ERIF	U1ERIF	_	0000
IFS5	008E	-	ı	_	ı	SPI3TXIF	SPI3IF	U4TXIF	U4RXIF	U4ERIF	USB1IF	I2C2BCIF	I2C1BCIF	U3TXIF	U3RXIF	U3ERIF	ı	0000
IFS6	0090	_	1	_	ı	_	FSTIF	ı	1	1	_	1	_	ı	_	ı	1	0000
IFS7	0092	_	_	_	_	_	_	_	_	_	_	JTAGIF	_	-	_	_	_	0000
IEC0	0094	_	DMA1IE	AD1IE	U1TXIE	U1RXIE	SPI1TXIE	SPI1IE	T3IE	T2IE	OC2IE	IC2IE	DMA0IE	T1IE	OC1IE	IC1IE	INT0IE	0000
IEC1	0096	U2TXIE	U2RXIE	INT2IE	T5IE	T4IE	OC4IE	OC3IE	DMA2IE	1	_	1	INT1IE	CNIE	CMIE	MI2C1IE	SI2C1IE	0000
IEC2	0098	_	DMA4IE	PMPIE	ı	_	OC6IE	OC5IE	IC6IE	IC5IE	IC4IE	IC3IE	DMA3IE	CRYROLLIE	CRYFREEIE	SPI2TXIE	SPI2IE	0000
IEC3	009A	_	RTCIE	DMA5IE	SPI3RXIE	SPI2RXIE	SPI1RXIE	ı	KEYSTRIE	CRYDNIE	INT4IE	INT3IE	_	ı	MI2C2IE	SI2C2IE	1	0000
IEC4	009C	_	1	CTMUIE	ı	_	-	ı	HLVDIE	1	_	1	_	CRCIE	U2ERIE	U1ERIE	1	0000
IEC5	009E	_	1	_	ı	SPI3TXIE	SPI3IE	U4TXIE	U4RXIE	U4ERIE	USB1IE	I2C2BCIE	I2C1BCIE	U3TXIE	U3RXIE	U3ERIE	1	0000
IEC6	00A0	_	1	_	ı	_	FSTIE	ı	1	1	_	1	_	ı	_	ı	1	0000
IEC7	00A2	_	1	_	ı	_	-	ı	1	1	_	JTAGIE	_	ı	_	ı	1	0000
IPC0	00A4	_	T1IP2	T1IP1	T1IP0	_	OC1IP2	OC1IP1	OC1IP0	1	IC1IP2	IC1IP1	IC1IP0	ı	INT0IP2	INT0IP1	INT0IP0	4444
IPC1	00A6	_	T2IP2	T2IP1	T2IP0	_	OC2IP2	OC2IP1	OC2IP0	1	IC2IP2	IC2IP1	IC2IP0	ı	DMA0IP2	DMA0IP1	DMA0IP0	4444
IPC2	00A8	_	U1RXIP2	U1RXIP1	U1RXIP0	_	SPI1TXIP2	SPI1TXIP1	SPI1TXIP0	1	SPI1IP2	SPI1IP1	SPI1IP0	ı	T3IP2	T3IP1	T3IP0	4444
IPC3	00AA	_	1	_	ı	_	DMA1IP2	DMA1IP1	DMA1IP0	1	AD1IP2	AD1IP1	AD1IP0	ı	U1TXIP2	U1TXIP1	U1TXIP0	0444
IPC4	00AC	_	CNIP2	CNIP1	CNIP0	_	CMIP2	CMIP1	CMIP0	1	MI2C1IP2	MI2C1IP1	MI2C1IP0	ı	SI2C1IP2	SI2C1IP1	SI2C1IP0	4444
IPC5	00AE	_	1	_	ı	_	-	ı	1	1	_	1	_	ı		INT1IP<2:0>		0004
IPC6	00B0	_	T4IP2	T4IP1	T4IP0	_	OC4IP2	OC4IP1	OC4IP0	1	OC3IP2	OC3IP1	OC3IP0	ı	DMA2IP2	DMA2IP1	DMA2IP0	4444
IPC7	00B2	_	U2TXIP2	U2TXIP1	U2TXIP0	_	U2RXIP2	U2RXIP1	U2RXIP0	1	INT2IP2	INT2IP1	INT2IP0	ı	T5IP2	T5IP1	T5IP0	4444
IPC8	00B4	_	CRYROLLIP2	CRYROLLIP1	CRYROLLIP0	_	CRYFREEIP2	CRYFREEIP1	CRYFREEIP0	1	SPI2TXIP2	SPI2TXIP1	SPI2TXIP0	ı	SPI2IP2	SPI2IP1	SPI2IP0	4444
IPC9	00B6	_	IC5IP2	IC5IP1	IC5IP0	_	IC4IP2	IC4IP1	IC4IP0	1	IC3IP2	IC3IP1	IC3IP0	ı	DMA3IP2	DMA3IP1	DMA3IP0	4444
IPC10	00B8	_	_	_		_	OC6IP2	OC6IP1	OC6IP0		OC5IP2	OC5IP1	OC5IP0		IC6IP2	IC6IP1	IC6IP0	0444
IPC11	00BA	_	_			_	DMA4IP2	DMA4IP1	DMA4IP0		PMPIP2	PMPIP1	PMPIP0	_				0440
IPC12	00BC	1	_	_	-	_	MI2C2IP2	MI2C2IP1	MI2C2IP0	_	SI2C2IP2	SI2C2IP1	SI2C2IP0	_	_	_	_	0440
IPC13	00BE	-	CRYDNIP2	CRYDNIP1	CRYDNIP0	_	INT4IP2	INT4IP1	INT4IP0	_	INT3IP2	INT3IP1	INT3IP0	_	_	_	_	4440
IPC14	00CO	_	SPI2RXIP2	SPI2RXIP1	SPI2RXIP0	_	SPI1RXIP2	SPI1RXIP1	SPI1RXIP0	_	_	_	_	-	KEYSTRIP2	KEYSTRIP1	KEYSTRIP0	4404
IPC15	00C2	-	-	_	_	_	RTCIP2	RTCIP1	RTCIP0	-	DMA5IP2	DMA5IP1	DMA5IP0	1	SPI3RXIP2	SPI3RXIP1	SPI3RXIP0	0444

Legend: — = unimplemented, read as '0'; r = reserved bit, maintain as '0'. Reset values are shown in hexadecimal.

TABLE 4-5: INTERRUPT CONTROLLER REGISTER MAP (CONTINUED)

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
IPC16	00C4	_	CRCIP2	CRCIP1	CRCIP0	_	U2ERIP2	U2ERIP1	U2ERIP0	_	U1ERIP2	U1ERIP1	U1ERIP0	_	_	_	_	4440
IPC18	00C8	_	-	-	-	-	_	_	_	-	_	_	_	_		HLVDIP<2:0>		0004
IPC19	00CA	_	-	-	-	-	_	_	_	-	(CTMUIP<2:0	>	_	_	_	_	0040
IPC20	00CC	_	U3TXIP2	U3TXIP1	U3TXIP0	-	U3RXIP2	U3RXIP1	U3RXIP0	-	U3ERIP2	U3ERIP1	U3ERIP0	_	_	_	_	4440
IPC21	00CE	_	U4ERIP2	U4ERIP1	U4ERIP0	-	USB1IP2	USB1IP1	USB1IP0	-	I2C2BCIP2	I2C2BCIP1	I2C2BCIP0	_	I2C1BCIP2	I2C1BCIP1	I2C1BCIP0	4444
IPC22	00D0	_	SPI3TXIP2	SPI3TXIP1	SPI3TXIP0	_	SPI3IP2	SPI3IP1	SPI3IP0	-	U4TXIP2	U4TXIP1	U4TXIP0	ı	U4RXIP2	U4RXIP1	U4RXIP0	4444
IPC26	00D8	_	-	-	-	-		FSTIP<2:0>		-	_	_	_	_	_	_	_	0400
IPC29	00DE	_	-	ı	-	_	-	_	-	-		JTAGIP<2:0	>	ı	_	_	_	0040
INTTREG	00E0	CPUIRQ	r	VHOLD	_	ILR3	ILR2	ILR1	ILR0	VECNUM7	VECNUM6	VECNUM5	VECNUM4	VECNUM3	VECNUM2	VECNUM1	VECNUM0	0000

Legend: — = unimplemented, read as '0'; r = reserved bit, maintain as '0'. Reset values are shown in hexadecimal.

TABLE 4-6: TIMER REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
TMR1	024C								Timer1 F	Register								0000
PR1	024E								Timer1 Peri	od Register								FFFF
T1CON	0250	TON	_	TSIDL	_	_	_	TECS1	TECS0	_	TGATE	TCKPS1	TCKPS0	_	TSYNC	TCS	_	0000
TMR2	0252								Timer2 F	Register								0000
TMR3HLD	0254		Timer3 Holding Register (for 32-bit timer operations only) Timer3 Register															0000
TMR3	0256		Timer3 Register															0000
PR2	0258																	FFFF
PR3	025A								Timer3 Peri	od Register								FFFF
T2CON	025C	TON	_	TSIDL	_	_	_	TECS1	TECS0	_	TGATE	TCKPS1	TCKPS0	T32	_	TCS	_	0000
T3CON	025E	TON	_	TSIDL	_	_	_	TECS1	TECS0	_	TGATE	TCKPS1	TCKPS0	_	_	TCS	_	0000
TMR4	0260								Timer4 F	Register								0000
TMR5HLD	0262						Tim	er5 Holdin	g Register (1	for 32-bit op	perations or	nly)						0000
TMR5	0264								Timer5 F	Register								0000
PR4	0266								Timer4 Peri	od Register								FFFF
PR5	0268								Timer5 Peri	od Register								FFFF
T4CON	026A	TON	_	TSIDL	_	_	_	TECS1	TECS0	_	TGATE	TCKPS1	TCKPS0	T45	_	TCS	_	0000
T5CON	026C	TON	_	TSIDL	_	_	_	TECS1	TECS0	_	TGATE	TCKPS1	TCKPS0	_	_	TCS	_	0000

TABLE 4-7: INPUT CAPTURE REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
IC1CON1	02AA	_	-	ICSIDL	ICTSEL2	ICTSEL1	ICTSEL0	_	_	_	ICI1	ICI0	ICOV	ICBNE	ICM2	ICM1	ICM0	0000
IC1CON2	02AC	_	_	_	_	_	_	_	IC32	ICTRIG	TRIGSTAT	_	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000D
IC1BUF	02AE							In	put Capture	1 Buffer Re	gister							0000
IC1TMR	02B0							Input	t Capture Ti	mer Value 1	Register							xxxx
IC2CON1	02B2	_	ı	ICSIDL	ICTSEL2	ICTSEL1	ICTSEL0	_	-	ı	ICI1	ICI0	ICOV	ICBNE	ICM2	ICM1	ICM0	0000
IC2CON2	02B4	-	_	-	_	_	_	-	IC32	ICTRIG	TRIGSTAT	-	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000D
IC2BUF	02B6		Input Capture 2 Buffer Register Input Capture 7 Engister Input Capture 7 Engister Input Capture 7 Engister Input Capture 7 Engister Input Capture 9 Engister Input Capture 9 Engister															0000
IC2TMR	02B8		Input Capture Timer Value 2 Register															xxxx
IC3CON1	02BA	_	- ICSIDL ICTSEL2 ICTSEL1 ICTSEL0 ICI1 ICI0 ICOV ICBNE ICM2 ICM1 ICM0															0000
IC3CON2	02BC	-																000D
IC3BUF	02BE							In	put Capture	3 Buffer Re	gister							0000
IC3TMR	02C0							Input	t Capture Ti	mer Value 3	Register							xxxx
IC4CON1	02C2	_	_	ICSIDL	ICTSEL2	ICTSEL1	ICTSEL0	_	_	_	ICI1	ICI0	ICOV	ICBNE	ICM2	ICM1	ICM0	0000
IC4CON2	02C4	_	-			-	_	_	IC32	ICTRIG	TRIGSTAT	_	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000D
IC4BUF	02C6							In	put Capture	4 Buffer Re	gister							0000
IC4TMR	02C8							Input	t Capture Ti	mer Value 4	Register							xxxx
IC5CON1	02CA	_	_	ICSIDL	ICTSEL2	ICTSEL1	ICTSEL0	_	_	_	ICI1	ICI0	ICOV	ICBNE	ICM2	ICM1	ICM0	0000
IC5CON2	02CC	_	-			-	_	_	IC32	ICTRIG	TRIGSTAT	_	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000D
IC5BUF	02CE							In	put Capture	5 Buffer Re	gister							0000
IC5TMR	02D0							Input	t Capture Ti	mer Value 5	Register							xxxx
IC6CON1	02D2	_	_	ICSIDL	ICTSEL2	ICTSEL1	ICTSEL0	_	_	-	ICI1	ICI0	ICOV	ICBNE	ICM2	ICM1	ICM0	0000
IC6CON2	02D4	_	_	_	_	-	_	_	IC32	ICTRIG	TRIGSTAT	_	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000D
IC6BUF	02D6							In	put Capture	6 Buffer Re	gister							0000
IC6TMR	02D8							Input	t Capture Ti	mer Value 6	Register							xxxx

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

TABLE 4-8: OUTPUT COMPARE REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
OC1CON1	026E	_	_	OCSIDL	OCTSEL2	OCTSEL1	OCTSEL0	ENFLT2	ENFLT1	ENFLT0	OCFLT2	OCFLT1	OCFLT0	TRIGMODE	OCM2	OCM1	OCM0	0000
OC1CON2	0270	FLTMD	FLTOUT	FLTTRIEN	OCINV		DCB1	DCB0	OC32	OCTRIG	TRIGSTAT	OCTRIS	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000C
OC1RS	0272							Ou	ıtput Compa	re 1 Second	ary Register							0000
OC1R	0274								Output C	ompare 1 Re	egister							0000
OC1TMR	0276							Out	tput Compar	e Timer Valu	ıe 1 Register							xxxx
OC2CON1	0278	_	_	OCSIDL	OCTSEL2	OCTSEL1	OCTSEL0	ENFLT2	ENFLT1	ENFLT0	OCFLT2	OCFLT1	OCFLT0	TRIGMODE	OCM2	OCM1	OCM0	0000
OC2CON2	027A	FLTMD	FLTOUT	FLTTRIEN	OCINV	_	DCB1	DCB0	OC32	OCTRIG	TRIGSTAT	OCTRIS	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000C
OC2RS	027C							Οι	ıtput Compa	re 2 Second	ary Register							0000
OC2R	027E		Output Compare 2 Register Output Compare Timer Value 2 Register OCSIDL OCTSEL2 OCTSEL1 OCTSEL0 ENFLT2 ENFLT1 ENFLT0 OCFLT2 OCFLT1 OCFLT0 TRIGMODE OCM2 OCM1 OCFLT0 OCFLT															0000
OC2TMR	0280		Output Compare Timer Value 2 Register — OCSIDL OCTSEL2 OCTSEL1 OCTSEL0 ENFLT2 ENFLT1 ENFLT0 OCFLT2 OCFLT1 OCFLT0 TRIGMODE OCM2 OCM1 O															xxxx
OC3CON1	0282	_	Output Compare Timer Value 2 Register OCSIDL OCTSEL2 OCTSEL1 OCTSEL0 ENFLT2 ENFLT1 ENFLT0 OCFLT2 OCFLT1 OCFLT0 TRIGMODE OCM2 OCM1 OCM0															0000
OC3CON2	0284	FLTMD	- OCSIDL OCTSEL2 OCTSEL1 OCTSEL0 ENFLT2 ENFLT1 ENFLT0 OCFLT2 OCFLT1 OCFLT0 TRIGMODE OCM2 OCM1 OCM0 MD FLTOUT FLTTRIEN OCINV — DCB1 DCB0 OC32 OCTRIG TRIGSTAT OCTRIS SYNCSEL4 SYNCSEL3 SYNCSEL2 SYNCSEL1 SYNCSEL1															000C
OC3RS	0286							Οι	ıtput Compa	re 3 Second	ary Register							0000
OC3R	0288								Output C	ompare 3 Re	egister							0000
OC3TMR	028A								tput Compar	e Timer Valu	ie 3 Register							xxxx
OC4CON1	028C	_	_	OCSIDL	OCTSEL2	OCTSEL1	OCTSEL0	ENFLT2	ENFLT1	ENFLT0	OCFLT2	OCFLT1	OCFLT0	TRIGMODE	OCM2	OCM1	OCM0	0000
OC4CON2	028E	FLTMD	FLTOUT	FLTTRIEN	OCINV	_	DCB1	DCB0	OC32	OCTRIG	TRIGSTAT	OCTRIS	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000C
OC4RS	0290							Οι	<u> </u>	re 4 Second	, ,							0000
OC4R	0292									ompare 4 Re	-							0000
OC4TMR	0294			r	,	,	-	Out	tput Compar	e Timer Valu	ie 4 Register						1	XXXX
OC5CON1	0296	_	_	OCSIDL	OCTSEL2	OCTSEL1	OCTSEL0	ENFLT2	ENFLT1	ENFLT0	OCFLT1	OCFLT1	OCFLT0	TRIGMODE	OCM2	OCM1	OCM0	0000
OC5CON2	0298	FLTMD	FLTOUT	FLTTRIEN	OCINV	_	DCB1	DCB0	OC32	OCTRIG	TRIGSTAT	OCTRIS	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000C
OC5RS	029A							Οι	<u> </u>	re 5 Second	, ,							0000
OC5R	029C								Output C	ompare 5 Re	egister							0000
OC5TMR	029E							Out	tput Compar	e Timer Valu	ie 5 Register							xxxx
OC6CON1	02A0	_	_	OCSIDL	OCTSEL2	OCTSEL1	OCTSEL0	ENFLT2	ENFLT1	ENFLT0	OCFLT2	OCFLT1	OCFLT0	TRIGMODE	OCM2	OCM1	OCM0	0000
OC6CON2	02A2	FLTMD	FLTOUT	FLTTRIEN	OCINV	_	DCB1	DCB0	OC32	OCTRIG	TRIGSTAT	OCTRIS	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000C
OC6RS	02A4							Ou	tput Compa	re 6 Second	ary Register							0000
OC6R	02A6								Output C	ompare 6 Re	egister							0000
OC6TMR	02A8							Out	tput Compar	e Timer Valu	ie 6 Register							xxxx

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

TABLE 4-9: I²C™ REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
I2C1RCV	02DA	_	_	_	_	_	_	_	_				2C1 Receiv	ve Register				0000
I2C1TRN	02DC	_	_	_	_	_	_	_	_			I	2C1 Transr	nit Register				OOFF
I2C1BRG	02DE	_	_	_	_					I2C1 B	aud Rate G	Senerator R	egister					0000
I2C1CONL	02E0	I2CEN	_	I2CSIDL	SCLREL	STRICT	A10M	DISSLW	SMEN	GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	1000
I2C1CONH	02E2	_	_	_	_	_	PCIE SCIE BOEN SDAHT SBCDE AHEN DHEN 00										0000	
I2C1STAT	02E4	ACKSTAT	TRSTAT	ACKTIM	_	_	BCL											0000
I2C1ADD	02E6	_	_	_	_	_	_				I	I2C1 Addre	ss Register					0000
I2C1MSK	02E8	_	_	_	_	_	_				120	1 Address	Mask Regis	ster				0000
I2C2RCV	02EA	_	_	_	_	_	_	_	_				2C2 Recei	ve Register				0000
I2C2TRN	02EC	_	_	_	_	_	_	_	_			I	2C2 Transr	nit Register				00FF
I2C2BRG	02EE	_	_	_	_					12C2 B	aud Rate G	Senerator R	egister					0000
I2C2CONL	02F0	I2CEN	_	I2CSIDL	SCLREL	STRICT	A10M	DISSLW	SMEN	GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	1000
I2C2CONH	02F2	_	_	_	_	_	_	PCIE SCIE BOEN SDAHT SBCDE AHEN DHEN 000										0000
I2C2STAT	02F4	ACKSTAT	TRSTAT	ACKTIM	_	_	BCL	GCSTAT ADD10 IWCOL I2COV D/A P S R/W RBF TBF 000										0000
I2C2ADD	02F6	_	-	_	_	_	_	I2C2 Address Register 00										
I2C2MSK	02F8	_	-	_	_	_		- I2C2 Address Mask Register 0										0000

TABLE 4-10: UART REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets			
U1MODE	0500	UARTEN	_	USIDL	IREN	RTSMD	_	UEN1	UEN0	WAKE	LPBACK	ABAUD	URXINV	BRGH	PDSEL1	PDSEL0	STSEL	0000			
U1STA	0502	UTXISEL1	UTXINV	UTXISEL0	URXEN	UTXBRK	UTXEN	UTXBF	TRMT	URXISEL1	URXISEL0	ADDEN	RIDLE	PERR	FERR	OERR	URXDA	0110			
U1TXREG	0504	LAST	_	_	_	-	_	_				U	1TXREG<8:0	>				xxxx			
U1RXREG	0506	_	_	1	_	-	_	_				U	1RXREG<8:0	>				0000			
U1BRG	0508									U1BRG<1	5:0>							0000			
U1ADMD	050A				ADMMAS	K<7:0>							ADMADE	OR<7:0>				0000			
U1SCCON	050C	_	_	ı	_	ı	_	_	-	_	_	TXRPT1	TXRPT0	CONV	T0PD	PTRCL	SCEN	0000			
U1SCINT	050E	_	_	RXRPTIF	TXRPTIF	ı	_	WTCIF	GTCIF	_	PARIE	RXRPTIE	TXRPTIE	_	_	WTCIE	GTCIE	0000			
U1GTC	0510	_	_	ı	_	ı	_	_					GTC<8:0>					0000			
U1WTCL	0512									WTC<15	i:0>							0000			
U1WTCH	0514	_	_	-	_		_	_	-				WTC<2	23:16>				0000			
U2MODE	0516	UARTEN	_	USIDL	IREN	RTSMD	_	UEN1	UEN0	WAKE	LPBACK	ABAUD	URXINV	BRGH	PDSEL1	PDSEL0	STSEL	0000			
U2STA	0518	UTXISEL1	UTXINV	UTXISEL0	URXEN	UTXBRK	UTXEN	UTXBF	TRMT	URXISEL1	URXISEL0	ADDEN	RIDLE	PERR	FERR	OERR	URXDA	0110			
U2TXREG	051A	LAST	_	_	_	_	_	_	TRMT URXISEL1 URXISEL0 ADDEN RIDLE PERR FERR OERR URXDA U2TXREG<8:0>												
U2RXREG	051C	_	_	_	_	_	_	_				U	2RXREG<8:0	>				0000			
U2BRG	051E									U2BRG<1	5:0>							0000			
U2ADMD	0520				ADMMAS	K<7:0>							ADMADE	OR<7:0>				0000			
U2SCCON	0522		_	_	_	_	_	_		_	_	TXRPT1	TXRPT0	CONV	T0PD	PTRCL	SCEN	0000			
U2SCINT	0524		_	RXRPTIF	TXRPTIF	_	_	WTCIF	GTCIF	_	PARIE	RXRPTIE	TXROTIE		_	WTCIE	GTCIE	0000			
U2GTC	0526		_	_	_	_	_	_					GTC<8:0>					0000			
U2WTCL	0528									WTC<15	5:0>							0000			
U2WTCH	052A		_	_	_	_	_	_					WTC<2	23:16>				0000			
U3MODE	052C	UARTEN	_	USIDL	IREN	RTSMD	_	UEN1	UEN0	WAKE	LPBACK	ABAUD	URXINV	BRGH	PDSEL1	PDSEL0	STSEL	0000			
U3STA	052E	UTXISEL1	UTXINV	UTXISEL0	URXEN	UTXBRK	UTXEN	UTXBF	TRMT	URXISEL1	URXISEL0	ADDEN	RIDLE	PERR	FERR	OERR	URXDA	0110			
U3TXREG	0530	LAST	_	_	_	_	_	_				U	3TXREG<8:0	>				xxxx			
U3RXREG	0532	_	_	_	_	_	_	_				U	3RXREG<8:0	>				0000			
U3BRG	0534									U3BRG<1	5:0>							0000			
U3ADMD	0536				ADMMAS	K<7:0>							ADMADE	OR<7:0>				0000			
U4MODE	0538	UARTEN	_	USIDL	IREN	RTSMD	_	UEN1	UEN0	NO WAKE LPBACK ABAUD URXINV BRGH PDSEL1 PDSEL0 STSEL											
U4STA	053A	UTXISEL1	UTXINV	UTXISEL0	URXEN	UTXBRK	UTXEN	UTXBF	TRMT	RMT URXISEL1 URXISEL0 ADDEN RIDLE PERR FERR OERR URXDA (
U4TXREG	053C	LAST	_	_	_	_	_	_				U	4TXREG<8:0	>				xxxx			
U4RXREG	053E	_	_	_	_	_	_	_				U	4RXREG<8:0	>				0000			
U4BRG	0540									U4BRG<1	5:0>							0000			
U4ADMD	0542				ADMMAS	K<7:0>							ADMADE	OR<7:0>				0000			

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

TABLE 4-11: SPI1 REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
SPI1CON1L	0300	SPIEN	_	SPISIDL	DISSDO	MODE32	MODE16	SMP	CKE	SSEN	CKP	MSTEN	DISSDI	DISSCK	MCLKEN	SPIFE	ENHBUF	0000
SPI1CON1H	0302	AUDEN	SPISGNEXT	IGNROV	IGNTUR	AUDMONO	URDTEN	AUDMOD1	AUDMOD0	FRMEN	FRMSYNC	FRMPOL	MSSEN	FRMSYPW	FRMCNT2	FRMCNT1	FRMCNT0	0000
SPI1CON2L	0304	-	ı	-	ı	_	-	1	_	ı	_	ı		V	VLENGTH<4:0)>		0000
SPI1STATL	0308	-	ı	-	FRMERR	SPIBUSY	-	1	SPITUR	SRMT	SPIROV	SPIRBE	ı	SPITBE	_	SPITBF	SPIRBF	0028
SPI1STATH	030A	-	-	RXELM5	RXELM4	LM4 RXELM3 RXELM2 RXELM1 RXELM0 — — TXELM5 TXELM4 TXELM3 TXELM2 TXELM1 TXELM0												
SPI1BUFL	030C								SPI1BL	JFL<15:0>								0000
SPI1BUFH	030E								SPI1BU	FH<31:16>								0000
SPI1BRGL	0310	1	1	-						5	SPI1BRG<12:0)>						0000
SPI1IMSKL	0314	-	ı	-	FRMERREN	BUSYEN	-	1	SPITUREN	SRMTEN	SPIROVEN	SPIRBEN	ı	SPITBEN	_	SPITBFEN	SPIRBFEN	0000
SPI1IMSKH	0316	RXWIEN	-	RXMSK5	RXMSK4	RXMSK3	RXMSK2	RXMSK1	RXMSK0	TXWIEN	_	TXMSK5	TXMSK4	TXMSK3	TXMSK2	TXMSK1	TXMSK0	0000
SPI1URDTL	0318								SPI1UR	DTL<15:0>								0000
SPI1URDTH	031A				•	•		•	SPI1URE	OTH<31:16>	•	•	•		•	•		0000

Legend: — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

TABLE 4-12: SPI2 REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets	
SPI2CON1L	031C	SPIEN	_	SPISIDL	DISSDO	MODE32	MODE16	SMP	CKE	SSEN	CKP	MSTEN	DISSDI	DISSCK	MCLKEN	SPIFE	ENHBUF	0000	
SPI2CON1H	031E	AUDEN	SPISGNEXT	IGNROV	IGNTUR	AUDMONO	URDTEN	AUDMOD1	AUDMOD0	FRMEN	FRMSYNC	FRMPOL	MSSEN	FRMSYPW	FRMCNT2	FRMCNT1	FRMCNT0	0000	
SPI2CON2L	0320	_	_	_	_	The second secon													
SPI2STATL	0324	_	_	_	FRMERR	SPIBUSY	_	_	SPITUR	SRMT	SPIROV	SPIRBE	_	SPITBE	_	SPITBF	SPIRBF	0028	
SPI2STATH	0326	_	_	RXELM5	RXELM4	M4 RXELM3 RXELM2 RXELM1 RXELM0 — TXELM5 TXELM4 TXELM3 TXELM2 TXELM1 TXELM0													
SPI2BUFL	0328								SPI2BUFL	<15:0>								0000	
SPI2BUFH	032A								SPI2BUFH	<31:16>								0000	
SPI2BRGL	032C	_	_	_						SF	PI2BRG<12:0>							0000	
SPI2IMSKL	0330	_	_	_	FRMERREN	BUSYEN	_	_	SPITUREN	SRMTEN	SPIROVEN	SPIRBEN	_	SPITBEN	_	SPITBFEN	SPIRBFEN	0000	
SPI2IMSKH	0332	RXWIEN	_	RXMSK5	RXMSK4	RXMSK3	RXMSK2	RXMSK1	RXMSK0	TXWIEN	_	TXMSK5	TXMSK4	TXMSK3	TXMSK2	TXMSK1	TXMSK0	0000	
SPI2URDTL	0334								SPI2URDT	L<15:0>								0000	
SPI2URDTH	0336								SPI2URDTH	1<31:16>								0000	

TABLE 4-13: SPI3 REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
SPI3CON1L	0338	SPIEN	_	SPISIDL	DISSDO	MODE32	MODE16	SMP	CKE	SSEN	CKP	MSTEN	DISSDI	DISSCK	MCLKEN	SPIFE	ENHBUF	0000
SPI3CON1H	033A	AUDEN	SPISGNEXT	IGNROV	IGNTUR	AUDMONO	URDTEN	AUDMOD1	AUDMOD0	FRMEN	FRMSYNC	FRMPOL	MSSEN	FRMSYPW	FRMCNT2	FRMCNT1	FRMCNT0	0000
SPI3CON2L	033C	_	_	_	_	_	_	_	_	_	_	_		V	VLENGTH<4:0)>		0000
SPI3STATL	0340	_	_	_	FRMERR	SPIBUSY	-	_	SPITUR	SRMT	SPIROV	SPIRBE	_	SPITBE	_	SPITBF	SPIRBF	0028
SPI3STATH	0342	_	_	RXELM5	RXELM4	RXELM3	RXELM2	RXELM1	RXELM0	_	_	TXELM5	TXELM4	TXELM3	TXELM2	TXELM1	TXELM0	0000
SPI3BUFL	0344																	0000
SPI3BUFH	0346								SPI3BUFF	l<31:16>								0000
SPI3BRGL	0348	-	_	_						S	PI3BRG<12:0)>						0000
SPI3IMSKL	034C	_	_	_	FRMERREN	BUSYEN	_	_	SPITUREN	SRMTEN	SPIROVEN	SPIRBEN	_	SPITBEN	_	SPITBFEN	SPIRBFEN	0000
SPI3IMSKH	034E	RXWIEN	_	RXMSK5	RXMSK4	RXMSK3	RXMSK2	RXMSK1	RXMSK0	TXWIEN	_	TXMSK5	TXMSK4	TXMSK3	TXMSK2	TXMSK1	TXMSK0	0000
SPI3URDTL	0350								SPI3URD1	TL<15:0>						•		0000
SPI3URDTH	0352								SPI3URDT	H<31:16>								0000

TABLE 4-14: PORTA REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
TRISA	0180	_	_	_	_	_		TRISA	<10:7>		_	_	_		TRISA	A<3:0>		078F
PORTA	0182	_	_	_	_	_		RA<	10:7>		_	_			RA<4:0>			xxxx
LATA	0184	_	_	_	_	_		LATA	<10:7>		_	_	_		LATA	<3:0>		xxxx
ODCA	0186	-	_	_	_	_		ODA<	:10:7>		_	_	_		ODA	<3:0>		0000

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

TABLE 4-15: PORTB REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
TRISB	018A	Т	RISB<15:13	3>	-	- TRISB<11:5> - TRISB<3:0> EFF									EFEF			
PORTB	018C		RB<15:13>		_	RB<11:0> xxx									xxxx			
LATB	018E	L	ATB<15:13	>	_	LATB<11:5> — LATB<3:0> xxx									xxxx			
ODCB	0190	(DDB<15:13	>	_	ODB<11:5> — ODB<3:0> 00									0000			

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

TABLE 4-16: PORTC REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9 ⁽¹⁾	Bit 8 ⁽¹⁾	Bit 7 ⁽¹⁾	Bit 6 ⁽¹⁾	Bit 5 ⁽¹⁾	Bit 4 ⁽¹⁾	Bit 3 ⁽¹⁾	Bit 2 ⁽¹⁾	Bit 1 ⁽¹⁾	Bit 0 ⁽¹⁾	All Resets
TRISC	0194	_	-	_	_	-	_										03FF ⁽²⁾	
PORTC	0196	_	_	_	_	_	_					RC<	9:0>					XXXX ⁽²⁾
LATC	0198	_	_	_	_	_	_					LATC	<9:0>					XXXX ⁽²⁾
ODCC	019A	_	_	_	_	_	_					ODC-	<9:0>					0000(2)

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

Note 1: These bits are not available on 28-pin devices; read as '0'.

2: The Reset value for 44-pin devices is shown.

TABLE 4-17: PAD CONFIGURATION REGISTER MAP (PADCFG1)

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
PADCFG1	01A0	_	-	_	_	_	_		1	1		_			-	-	PMPTTL	0000

DS30005009C-page 51

TABLE 4-18: A/D CONVERTER REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
ADC1BUF0	0200						A/D [Data Buffer	0/Thresho	ld for Chanr	nel 0							xxxx
ADC1BUF1	0202						A/D [Data Buffer	1/Thresho	ld for Chanr	nel 1							xxxx
ADC1BUF2	0204						A/D [Data Buffer	2/Thresho	ld for Chanr	nel 2							xxxx
ADC1BUF3	0206						A/D [Data Buffer	3/Thresho	ld for Chanr	nel 3							xxxx
ADC1BUF4	0208						A/D [Data Buffer	4/Thresho	ld for Chanr	nel 4							xxxx
ADC1BUF5	020A						A/D [Data Buffer	5/Thresho	ld for Chanr	nel 5							xxxx
ADC1BUF6	020C						A/D [Data Buffer	6/Thresho	ld for Chanr	nel 6							xxxx
ADC1BUF7	020E						A/D [Data Buffer	7/Thresho	ld for Chanr	nel 7							xxxx
ADC1BUF8	0210				A/D	Data Buffer 8/	Threshold for	Channel 8/	Threshold	for Channel	0 in Windo	wed Comp	are mode					xxxx
ADC1BUF9	0212				A/D	Data Buffer 9/	Threshold for	Channel 9/	Threshold	for Channel	1 in Windo	wed Comp	are mode					xxxx
ADC1BUF10	0214				A/D Da	ata Buffer 10/T	hreshold for C	channel 10/	Threshold	for Channel	2 in Windo	wed Comp	are mode ⁽¹	1)				xxxx
ADC1BUF11	0216		A/D Data Buffer 11/Threshold for Channel 11/Threshold for Channel 3 in Windowed Compare mode ⁽¹⁾															xxxx
ADC1BUF12	0218		A/D Data Buffer 11/Threshold for Channel 11/Threshold for Channel 3 in Windowed Compare mode ⁽¹⁾ A/D Data Buffer 12/Threshold for Channel 12/Threshold for Channel 4 in Windowed Compare mode ⁽¹⁾															xxxx
ADC1BUF13	021A							A/D	Data Buffe	r 13								xxxx
ADC1BUF14	021C							A/D	Data Buffe	r 14								xxxx
ADC1BUF15	021E							A/D	Data Buffe	r 15								xxxx
AD1CON1	0220	ADON	-	ADSIDL	DMABM	DMAEN	MODE12	FORM1	FORM0	SSRC3	SSRC2	SSRC1	SSRC0	_	ASAM	SAMP	DONE	0000
AD1CON2	0222	PVCFG1	PVCFG0	NVCFG0	OFFCAL	BUFREGEN	CSCNA	_	_	BUFS	SMPI4	SMPI3	SMPI2	SMPI1	SMPI0	BUFM	ALTS	0000
AD1CON3	0224	ADRC	EXTSAM	PUMPEN	SAMC4	SAMC3	SAMC2	SAMC1	SAMC0	ADCS7	ADCS6	ADCS5	ADCS4	ADCS3	ADCS2	ADCS1	ADCS0	0000
AD1CHS	0228	CH0NB2	CH0NB1	CH0NB0	CH0SB4	CH0SB3	CH0SB2	CH0SB1	CH0SB0	CH0NA2	CH0NA1	CH0NA0	CH0SA4	CH0SA3	CH0SA2	CH0SA1	CH0SA0	0000
AD1CSSH	022A			CSS<31:2			-	_	_	_	_	_	_	_	_	_	_	0000
AD1CSSL	022C				CSS<	:14:9> ⁽²⁾			_				CSS<	7:0>				0000
AD1CON4	022E		-	_	_	_	ı	_	_	_	-	-	ı	_		DMABL<2:0)>	0000
AD1CON5	0230	ASEN	LPEN	CTMREQ	BGREQ	_	ı	ASINT1	ASINT0	_	-	-	ı	WM1	WM0	CM1	CM0	0000
AD1CHITL	0234	_	_	_		CHH<12	2:9> ⁽²⁾		_				CHH<	7:0>				0000
AD1CTMENL	0238	_	_	_		CTMEN<	12:9> ⁽²⁾		_				CTMEN	I<7:0>				0000
AD1DMBUF	023A		•		•	•	A/D Conv	ersion Data	Buffer (Ex	tended Buff	fer mode)				•	•		xxxx

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

Note 1: These bits are unimplemented in 28-pin devices, read as '0'.

2: The CSS<12:10>, CHH<12:10> and CTMEN<12:10> bits are unimplemented in 28-pin devices, read as '0'.

TABLE 4-19: CTMU REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
CTMUCON1	023C	CTMUEN	-	CTMUSIDL	TGEN	EDGEN	EDGSEQEN	IDISSEN	CTTRIG	_	_	_	_	_	_	-	_	0000
CTMUCON2	023E	EDG1MOD	EDG1POL	EDG1SEL3	EDG1SEL2	EDG1SEL1	EDG1SEL0	EDG2STAT	EDG1STAT	EDG2MOD	EDG2POL	EDG2SEL3	EDG2SEL2	EDG2SEL1	EDG2SEL0	_	_	0000
CTMUICON	0240	ITRIM5	ITRIM4	ITRIM3	ITRIM2	ITRIM1	ITRIM0	IRNG1	IRNG0	_	_	ı	1	_	1	_	_	0000

Legend: — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

TABLE 4-20: ANALOG CONFIGURATION REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
ANCFG	019E	_	_	_	_	_	_	_	_	_	_	_	_	_	_	VBG2EN	VBGEN	0000
ANSA	0188	_	_	_	-	_	_	_	_	_	_	_	_	ANSA<3:0>				000F
ANSB	0192	А	NSB<15:13	>	-	_	_	ANSB9	_	_	ANSB6	_	_	ANSB<3:0>				E24F
ANSC	019C	-	_	_		-	-	-	-	-	ı	-	_	ı	P	ANSC<2:0> ⁽	1)	0007

Legend: — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

Note 1: These bits are unimplemented in 28-pin devices, read as '0'.

TARI F 1-21.	DMA REGISTER MAP
IADLE 4-21.	DIVIA REGISTER WAF

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
DMACON	0450	DMAEN	_	_	_	_	_	_	_	_	_	_	_	_	_	_	PRSSEL	0000
DMABUF	0452								DMA Trans	fer Data Buf	fer							0000
DMAL	0454							DM	A High Add	ress Limit Re	egister							0000
DMAH	0456							DM	1A Low Add	ress Limit Re	gister							0000
DMACH0	0458	_		_	r	_	NULLW	RELOAD	CHREQ	SAMODE1	SAMODE0	DAMODE1	DAMODE0	TRMODE1	TRMODE0	SIZE	CHEN	0000
DMAINT0	045A	DBUFWF	_	CHSEL5	CHSEL4	CHSEL3	CHSEL2	CHSEL1	CHSEL0	HIGHIF	LOWIF	DONEIF	HALFIF	OVRUNIF	_	_	HALFEN	0000
DMASRC0	045C							DMA C	hannel 0 Sc	urce Addres	s Register							0000
DMADST0	045E							DMA Cha	nnel 0 Dest	ination Addr	ess Register							0000
DMACNT0	0460							DMA Ch	annel 0 Tra	nsaction Cou	ınt Register							0001
DMACH1	0462	_	_	_	r	1	NULLW	RELOAD	CHREQ	SAMODE1	SAMODE0	DAMODE1	DAMODE0	TRMODE1	TRMODE0	SIZE	CHEN	0000
DMAINT1	0464	DBUFWF	-WF — CHSEL5 CHSEL4 CHSEL3 CHSEL2 CHSEL1 CHSEL0 HIGHIF LOWIF DONEIF HALFIF OVRUNIF — HALFIF OVRUNIF — HALFIF DMA Channel 1 Source Address Register DMA Channel 1 Destination Address Register															0000
DMASRC1	0466		DMA Channel 1 Source Address Register DMA Channel 1 Destination Address Register															0000
DMADST1	0468		DMA Channel 1 Destination Address Register DMA Channel 1 Transaction Count Register															0000
DMACNT1	046A		DMA Channel 1 Transaction Count Register															0001
DMACH2	046C	_	DMA Channel 1 Transaction Count Register — r — NULLW RELOAD CHREQ SAMODE1 SAMODE0 DAMODE0 TRMODE1 TRMODE0 SIZE CH															0000
DMAINT2	046E	DBUFWF	r _ NULLW RELOAD CHREQ SAMODE1 SAMODE0 DAMODE1 DAMODE0 TRMODE1 TRMODE0 SIZE CH															0000
DMASRC2	0470							DMA C	hannel 2 Sc	ource Addres	s Register							0000
DMADST2	0472							DMA Cha	nnel 2 Dest	ination Addre	ess Register							0000
DMACNT2	0474							DMA Ch	annel 2 Tra	nsaction Cou	ınt Register							0001
DMACH3	0476	_	_	_	r	_	NULLW	RELOAD	CHREQ	SAMODE1	SAMODE0	DAMODE1	DAMODE0	TRMODE1	TRMODE0	SIZE	CHEN	0000
DMAINT3	0478	DBUFWF	_	CHSEL5	CHSEL4	CHSEL3	CHSEL2	CHSEL1	CHSEL0	HIGHIF	LOWIF	DONEIF	HALFIF	OVRUNIF	_	_	HALFEN	0000
DMASRC3	047A							DMA C	hannel 3 Sc	ource Addres	s Register							0000
DMADST3	047C							DMA Cha	nnel 3 Dest	ination Addr	ess Register							0000
DMACNT3	047E							DMA Ch	annel 3 Tra	nsaction Cou	ınt Register							0001
DMACH4	0480	_		_	r	_	NULLW	RELOAD	CHREQ	SAMODE1	SAMODE0	DAMODE1	DAMODE0	TRMODE1	TRMODE0	SIZE	CHEN	0000
DMAINT4	0482	DBUFWF	_	CHSEL5	CHSEL4	CHSEL3	CHSEL2	CHSEL1	CHSEL0	HIGHIF	LOWIF	DONEIF	HALFIF	OVRUNIF	_	_	HALFEN	0000
DMASRC4	0484							DMA C	hannel 4 Sc	ource Addres	s Register							0000
DMADST4	0486							DMA Cha	nnel 4 Dest	ination Addr	ess Register							0000
DMACNT4	0488				_		-	DMA Ch	annel 4 Tra	nsaction Cou	ınt Register							0001
DMACH5	048A	_	_	_	r	1	NULLW	RELOAD	CHREQ	SAMODE1	SAMODE0	DAMODE1	DAMODE0	TRMODE1	TRMODE0	SIZE	CHEN	0000
DMAINT5	048C	DBUFWF		CHSEL5	CHSEL4	CHSEL3	CHSEL2	CHSEL1	CHSEL0	HIGHIF	LOWIF	DONEIF	HALFIF	OVRUNIF	_	_	HALFEN	0000
DMASRC5	048E							DMA C	hannel 5 Sc	ource Addres	s Register							0000
DMADST5	0490							DMA Cha	nnel 5 Dest	ination Addr	ess Register							0000
DMACNT5	0492							DMA Ch	annel 5 Tra	nsaction Cou	ınt Register							0001

Legend: — = unimplemented, read as '0'; r = reserved bit. Reset values are shown in hexadecimal.

TABLE 4-22: USB OTG REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
U10TGIR	0400	_	_	_	_	_	_	_	_	IDIF	T1MSECIF	LSTATEIF	ACTVIF	SESVDIF	SESENDIF	_	VBUSVDIF	0000
U10TGIE	0402	_	_	_	_	1	_	I	_	IDIE	T1MSECIE	LSTATEIE	ACTVIE	SESVDIE	SESENDIE	_	VBUSVDIE	0000
U1OTGSTAT	0404	_	_	_		1	_	1	_	ID	_	LSTATE	_	SESVD	SESEND	_	VBUSVD	0000
U10TGCON	0406	_	_	_	_	1	_	1	_	DPPULUP	DMPULUP	DPPULDWN	DMPULDWN	VBUSON	OTGEN	VBUSCHG	VBUSDIS	0000
U1PWRC	0408	-	_	_		-	_	-	_	UACTPND	_	ı	USLPGRD	-	_	USUSPND	USBPWR	00x0
U1IR	040A ⁽¹⁾	_	_	_		1	_	1	_	STALLIF	_	RESUMEIF	IDLEIF	TRNIF	SOFIF	UERRIF	URSTIF	0000
		_	_	_	_	1	_	1	_	STALLIF	ATTACHIF ⁽¹⁾	RESUMEIF	IDLEIF	TRNIF	SOFIF	UERRIF	DETACHIF ⁽¹⁾	0000
U1IE	040C ⁽¹⁾	_	_	_	_	1	_	I	_	STALLIE	_	RESUMEIE	IDLEIE	TRNIE	SOFIE	UERRIE	URSTIE	0000
		_	_	_	1	-	_	-	_	STALLIE	ATTACHIE(1)	RESUMEIE	IDLEIE	TRNIE	SOFIE	UERRIE	DETACHIE(1)	0000
U1EIR	040E ⁽¹⁾	_	_	_	_	_	_	_	_	BTSEF	_	DMAEF	BTOEF	DFN8EF	CRC16EF	CRC5EF	PIDEF	0000
		_	_	_		_	_	_	_	BTSEF	_	DMAEF	BTOEF	DFN8EF	CRC16EF	EOFEF ⁽¹⁾	PIDEF	0000
U1EIE	0410 ⁽¹⁾	_	_	_	-	_	_	_	_	BTSEE	_	DMAEE	BTOEE	DFN8EE	CRC16EE	CRC5EE	PIDEE	0000
		_	_	_	_	_	_	_	_	BTSEE	_	DMAEE	BTOEE	DFN8EE	CRC16EE	EOFEE(1)	PIDEE	0000
U1STAT	0412	_	_	_		_	_	_	_	ENDPT3	ENDPT2	ENDPT1	ENDPT0	DIR	PPBI	_	_	0000
U1CON	0414 ⁽¹⁾	_	_	_	-	_	_	_	_	_	SE0	PKTDIS	_	HOSTEN	RESUME	PPBRST	USBEN	0000
		_	_	_	_	_	_	_	_	JSTATE ⁽¹⁾	SE0	TOKBUSY	USBRST	HOSTEN	RESUME	PPBRST	SOFEN ⁽¹⁾	0000
U1ADDR	0416	_	_	_	_	1	_	I	_	LSPDEN ⁽¹⁾	ADDR6	ADDR5	ADDR4	ADDR3	ADDR2	ADDR1	ADDR0	0000
U1BDTP1	0418	_	_	_	1	-	_	-	_		USB E	Buffer Descript	tor Table Base	Address Re	gister		_	0000
U1FRML	041A	_	_	_		1	_	1	_			USB	Frame Count F	Register Low	Byte			0000
U1FRMH	041C	-	_	_	1	I	_	I	_			USB I	Frame Count F	Register High	n Byte			0000
U1TOK ⁽²⁾	041E	_	_	_	-	_	_	_	_	PID3	PID2	PID1	PID0	EP3	EP2	EP1	EP0	0000
U1SOF ⁽²⁾	0420	_	_	_	_	_	_	_	_			USB Sta	art-of-Frame (S	OF) Count F	Register			0000
U1BDTP2	0422	_	_	_		_	_	_	_		USB I	Buffer Descript	tor Table Base	Address Re	gister		_	0000
U1BDTP3	0424	_	_	_		_	_	_	_		USB I	Buffer Descript	tor Table Base	Address Re	gister		_	0000
U1CNFG1	0426	_	_	_	_	_	_	_	_	UTEYE	UOEMON	_	USBSIDL	_	_	PPB1	PPB0	0000
U1CNFG2	0428	_	_	_				_			_	I	PUVBUS	EXTI2CEN	UVBUSDIS		UTRDIS	0000
U1EP0	042A	_	_	_	_	_	_	_	_	LSPD ⁽¹⁾	RETRYDIS(1)	_	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
U1EP1	042C	_	_	_	_	_	_		_	_	_		EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
U1EP2	042E	_	_	_	_	_	_		_	_	_	1	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
U1EP3	0430	-	_		_	_	_	_	_	_	_	ı	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000

Legend: — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

Note 1: Alternate register or bit definitions when the module is operating in Host mode.

2: These registers are available in Host mode only.

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
U1EP4	0432	_	_	_	_	_	_	_	_	_	_	_	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
U1EP5	0434	_	_	_	_	_	_		_	_	_	_	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
U1EP6	0436	_	_	_	_	_	_		_	_	_	_	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
U1EP7	0438	_	_	_	_	_	_		_	_	_	_	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
U1EP8	043A	_	_	_	_	_	_		_	_	_	_	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
U1EP9	043C	_	_	_	_	_	_	_	_	_	_	_	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
U1EP10	043E	_	_	_	_	_	_		_	_	_	_	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
U1EP11	0440	_	_	_	_	_	_		_	_	_	_	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
U1EP12	0442	_	_	_	_	_	_		_	_	_	_	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
U1EP13	0444	_	_	_	_	_	_	_	_	_	_	_	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
U1EP14	0446	_	_	_	_	_	_	_	_	_	_	_	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
U1EP15	0448	_	_	_	_	_	_	_	_	_	_	_	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000

Legend: — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

Note 1: Alternate register or bit definitions when the module is operating in Host mode.

2: These registers are available in Host mode only.

TABLE 4-23: ENHANCED PARALLEL MASTER/SLAVE PORT REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
PMCON1	0128	PMPEN	_	PSIDL	ADRMUX1	ADRMUX0	_	MODE1	MODE0	CSF1	CSF0	ALP	ALMODE	_	BUSKEEP	IRQM1	IRQM0	0000
PMCON2	012A	PMPBUSY	_	ERROR	TIMEOUT	_	_	_	_	RADDR23	RADDR22	RADDR21	RADDR20	RADDR19	RADDR18	RADDR17	RADDR16	0000
PMCON3	012C	PTWREN	PTRDEN	PTBE1EN	PTBE0EN	_	AWAITM1	AWAITM0	AWAITE	_	_	_	_	_	_	_	_	0000
PMCON4	012E	-	PTEN14	_	_	_	_					PTEN	I<9:0>					0000
PMCS1CF	0130	CSDIS	CSP	CSPTEN	BEP	_	WRSP	RDSP	SM	ACKP	PTSZ1	PTSZ0	_	_	_	_	_	0000
PMCS1BS	0132				В	BASE<23:15>	•		BASE11									
PMCS1MD	0134	ACKM1	ACKM0	AMWAIT2	AMWAIT1	AMWAIT0	_	_	_	DWAITB1	DWAITB0	DWAITM3	DWAITM2	DWAITM1	DWAITM0	DWAITE1	DWAITE0	0000
PMCS2CF	0136	CSDIS	CSP	CSPTEN	BEP	_	WRSP	RDSP	SM	ACKP	PTSZ1	PTSZ0	_	_	_	_	_	0000
PMCS2BS	0138				В	BASE<23:15>	•				_	_	_	BASE11	_	_	_	0600
PMCS2MD	013A	ACKM1	ACKM0	AMWAIT2	AMWAIT1	AMWAIT0	_	_	_	DWAITB1	DWAITB0	DWAITM3	DWAITM2	DWAITM1	DWAITM0	DWAITE1	DWAITE0	0000
PMDOUT1	013C			EPM	IP Data Out I	Register 1<1	5:8>					EPN	/IP Data Out	Register 1<	7:0>			xxxx
PMDOUT2	013E			EPM	IP Data Out I	Register 2<1	5:8>					EPN	/IP Data Out	Register 2<	7:0>			xxxx
PMDIN1	0140			EPI	MP Data In R	Register 1<15	:8>					EP	MP Data In	Register 1<7	7:0>			xxxx
PMDIN2	0142			EPI	MP Data In R	Register 2<15	:8>			_		EP	MP Data In	Register 2<7	7:0>			xxxx
PMSTAT	0144	IBF	IBOV	ı	_	IB3F	IB2F	IB1F	IB0F	OBE	OBUF	_	_	OB3E	OB2E	OB1E	OB0E	008F

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

TABLE 4-24: REAL-TIME CLOCK AND CALENDAR (RTCC) REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
ALRMVAL	011E						Alarm Va	alue Register	Window Bas	ed on ALR	MPTR<1:0	>						xxxx
ALCFGRPT	0120	ALRMEN	CHIME	AMASK3	AMASK2	AMASK1	AMASK0	ALRMPTR1	ALRMPTR0	ARPT7	ARPT6	ARPT5	ARPT4	ARPT3	ARPT2	ARPT1	ARPT0	0000
RTCVAL	0122						RTCC \	/alue Registe	r Window Ba	sed on RT0	CPTR<1:0>	•						xxxx
RCFGCAL	0124	RTCEN	-	RTCWREN	RTCSYNC	HALFSEC	RTCOE	RTCPTR1	RTCPTR0	CAL7	CAL6	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	Note 1
RTCPWC	0126	PWCEN	PWCPOL	PWCPRE	PWSPRE	RTCLK1	RTCLK0	RTCOUT1	RTCOUT0	_	_	_	_	_	_	_	_	Note 1

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

Note 1: The status of the RCFGCAL and RTCPWC registers on POR is '0000' and on other Resets, it is unchanged.

TABLE 4-25: DATA SIGNAL MODULATOR (DSM) REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
MDCON	02FA	MDEN	1	MDSIDL	_	-	-	_	_	_	MDOE	MDSLR	MDOPOL	_	_	_	MDBIT	0020
MDSRC	02FC	_	_	_	_	_	_	_	_	SODIS	_	_	_	MS3	MS2	MS1	MS0	0000
MDCAR	02FE	CHODIS	CHPOL	CHSYNC	-	CH3	CH2	CH1	CH0	CLODIS	CLPOL	CLSYNC	_	CL3	CL2	CL1	CL0	0000

PIC24FJ128GB204 FAMILY

Legend: — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

TABLE 4-26: COMPARATOR REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
CMSTAT	0242	CMIDL	_	_	_		C3EVT	C2EVT	C1EVT	_	_	_	_	_	C3OUT	C2OUT	C1OUT	0000
CVRCON	0244	-	_	_	_	_	CVREFP	CVREFM1	CVREFM0	CVREN	CVROE	CVRSS	CVR4	CVR3	CVR2	CVR1	CVR0	0000
CM1CON	0246	CON	COE	CPOL	_	_	_	CEVT	COUT	EVPOL1	EVPOL0	_	CREF	_	-	CCH1	CCH0	0000
CM2CON	0248	CON	COE	CPOL	_	_	_	CEVT	COUT	EVPOL1	EVPOL0	_	CREF	_	-	CCH1	CCH0	0000
CM3CON	024A	CON	COE	CPOL	_	_	_	CEVT	COUT	EVPOL1	EVPOL0	_	CREF	_	1	CCH1	CCH0	0000

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
CRCCON1	0158	CRCEN	_	CSIDL	VWORD4	VWORD3	VWORD2	VWORD1	VWORD0	CRCFUL	CRCMPT	CRCISEL	CRCGO	LENDIAN	_	_	_	0040
CRCCON2	015A	_	-	ı	DWIDTH4	DWIDTH3	DWIDTH2	DWIDTH1	DWIDTH0	_	-	-	PLEN4	PLEN3	PLEN2	PLEN1	PLEN0	0000
CRCXORL	015C		X<15:1> —														0000	
CRCXORH	015E		X<15:1>															0000
CRCDATL	0160							CRC	Data Input	Register L	ow							xxxx
CRCDATH	0162							CRC	Data Input	Register H	igh							xxxx
CRCWDATL	0164							CF	RC Result R	egister Lov	v							xxxx
CRCWDATH	0166							CF	RC Result R	egister Higl	h							xxxx

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

TABLE 4-28: PERIPHERAL PIN SELECT REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
RPINR0	038C	_	_	INT1R5	INT1R4	INT1R3	INT1R2	INT1R1	INT1R0	_	_	OCTRIG1R5	OCTRIG1R4	OCTRIG1R3	OCTRIG1R2	OCTRIG1R1	OCTRIG1R0	3F3F
RPINR1	038E	-	_	INT3R5	INT3R4	INT3R3	INT3R2	INT3R1	INT3R0	_	-	INT2R5	INT2R4	INT2R3	INT2R2	INT2R1	INT2R0	3F3F
RPINR2	0390	_	_	OCTRIG2R5	OCTRIG2R4	OCTRIG2R3	OCTRIG2R2	OCTRIG2R1	OCTRIG2R0	_	_	INT4R5	INT4R4	INT4R3	INT4R2	INT4R1	INT4R0	3F3F
RPINR7	039A	_	_	IC2R5	IC2R4	IC2R3	IC2R2	IC2R1	IC2R0	_	_	IC1R5	IC1R4	IC1R3	IC1R2	IC1R1	IC1R0	3F3F
RPINR8	039C	_	_	IC4R5	IC4R4	IC4R3	IC4R2	IC4R1	IC4R0	_	_	IC3R5	IC3R4	IC3R3	IC3R2	IC3R1	IC3R0	3F3F
RPINR9	039E	_	_	IC6R5	IC6R4	IC6R3	IC6R2	IC6R1	IC6R0	_	_	IC5R5	IC5R4	IC5R3	IC5R2	IC5R1	IC5R0	3F3F
RPINR11	03A2	_	_	OCFBR5	OCFBR4	OCFBR3	OCFBR2	OCFBR1	OCFBR0	_	_	OCFAR5	OCFAR4	OCFAR3	OCFAR2	OCFAR1	OCFAR0	3F3F
RPINR17	03AE	_	_			U3RXF	R<5:0>			_	_	_	_	_	_	_	_	3F00
RPINR18	03B0	_	_	U1CTSR5	U1CTSR4	U1CTSR3	U1CTSR2	U1CTSR1	U1CTSR0	_	_	U1RXR5	U1RXR4	U1RXR3	U1RXR2	U1RXR1	U1RXR0	3F3F
RPINR19	03B2	_	_	U2CTSR5	U2CTSR4	U2CTSR3	U2CTSR2	U2CTSR1	U2CTSR0	_	_	U2RXR5	U2RXR4	U2RXR3	U2RXR2	U2RXR1	U2RXR0	3F3F
RPINR20	03B4	_	_	SCK1R5	SCK1R4	SCK1R3	SCK1R2	SCK1R1	SCK1R0	_	_	SDI1R5	SDI1R4	SDI1R3	SDI1R2	SDI1R1	SDI1R0	3F3F
RPINR21	03B6	_	_	U3CTSR5	U3CTSR4	U3CTSR3	U3CTSR2	U3CTSR1	U3CTSR0	_	_	SS1R5	SS1R4	SS1R3	SS1R2	SS1R1	SS1R0	3F3F
RPINR22	03B8	_	_	SCK2R5	SCK2R4	SCK2R3	SCK2R2	SCK2R1	SCK2R0	_	_	SDI2R5	SDI2R4	SDI2R3	SDI2R2	SDI2R1	SDI2R0	3F3F
RPINR23	03BA	_	_	TMRCKR5	TMRCKR4	TMRCKR3	TMRCKR2	TMRCKR1	TMRCKR0	_	_	SS2R5	SS2R4	SS2R3	SS2R2	SS2R1	SS2R0	3F3F
RPINR27	03C2	_	_	U4CTSR5	U4CTSR4	U4CTSR3	U4CTSR2	U4CTSR1	U4CTSR0	_	_	U4RXR5	U4RXR4	U4RXR3	U4RXR2	U4RXR1	U4RXR0	3F3F
RPINR28	03C4	_	_	SCK3R5	SCK3R4	SCK3R3	SCK3R2	SCK3R1	SCK3R0	_		SDI3R5	SDI3R4	SDI3R3	SDI3R2	SDI3R1	SDI3R0	3F3F
RPINR29	03C6	_	_	_	_	_	_	_	_	_	_			SS3R	<5:0>			003F
RPINR30	03C8	_	_	_	_	_	_	_	_	_				MDMIF	R<5:0>			003F
RPINR31	03CA	_	_	MDC2R5	MDC2R4	MDC2R3	MDC2R2	MDC2R1	MDC2R0	_	_	MDC1R5	MDC1R4	MDC1R3	MDC1R2	MDC1R1	MDC1R0	3F3F

TABLE 4-28: PERIPHERAL PIN SELECT REGISTER MAP (CONTINUED)

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
RPOR0	03D6	_	_	RP1R5	RP1R4	RP1R3	RP1R2	RP1R1	RP1R0	_	_	RP0R5	RP0R4	RP0R3	RP0R2	RP0R1	RP0R0	0000
RPOR1	03D8	_	_	RP3R5	RP3R4	RP3R3	RP3R2	RP3R1	RP3R0	_	-	RP2R5	RP2R4	RP2R3	RP2R2	RP2R1	RP2R0	0000
RPOR2	03DA	1	1			RP5R	<5:0>			-	1	1	ı	1	_	1	1	0000
RPOR3	03DC	1	1	RP7R5	RP7R4	RP7R3	RP7R2	RP7R1	RP7R0	1	1	RP6R5	RP6R4	RP6R3	RP6R2	RP6R1	RP6R0	0000
RPOR4	03DE	_	-	RP9R5	RP9R4	RP9R3	RP9R2	RP9R1	RP9R0	_	-	RP8R5	RP8R4	RP8R3	RP8R2	RP8R1	RP8R0	0000
RPOR5	03E0	1	1	RP11R5	RP11R4	RP11R3	RP11R2	RP11R1	RP11R0	-	1	RP10R5	RP10R4	RP10R3	RP10R2	RP10R1	RP10R0	0000
RPOR6	03E2	1	1			RP13F	R<5:0>			1	1	-	1	_	_	_	_	0000
RPOR7	03E4	_	-	RP15R5	RP15R4	RP15R3	RP15R2	RP15R1	RP15R0	_	-	RP14R5	RP14R4	RP14R3	RP14R2	RP14R1	RP14R0	0000
RPOR8	03E6	1		RP17R5	RP17R4	RP17R3	RP17R2	RP17R1	RP17R0	1	1	RP16R5	RP16R4	RP16R3	RP16R2	RP16R1	RP16R0	0000
RPOR9	03E8	1	1	RP19R5	RP19R4	RP19R3	RP19R2	RP19R1	RP19R0	1	1	RP18R5	RP18R4	RP18R3	RP18R2	RP18R1	RP18R0	0000
RPOR10	03EA	_		RP21R5	RP21R4	RP21R3	RP21R2	RP21R1	RP21R0	_		RP20R5	RP20R4	RP20R3	RP20R2	RP20R1	RP20R0	0000
RPOR11	03EC	_	_	RP23R5	RP23R4	RP23R3	RP23R2	RP23R1	RP23R0	_	_	RP22R5	RP22R4	RP22R3	RP22R2	RP22R1	RP22R0	0000
RPOR12	03EE	_	_	RP25R5	RP25R4	RP25R3	RP25R2	RP25R1	RP25R0	_	_	RP24R5	RP24R4	RP24R3	RP24R2	RP24R1	RP24R0	0000

Legend: — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

TABLE 4-29: SYSTEM CONTROL (CLOCK AND RESET) REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
RCON	0108	TRAPR	IOPUWR	_	RETEN	_	DPSLP	CM	VREGS	EXTR	SWR	SWDTEN	WDTO	SLEEP	IDLE	BOR	POR	Note 1
OSCCON	0100	_	COSC2	COSC1	COSC0	_	NOSC2	NOSC1	NOSC0	CLKLOCK	IOLOCK	LOCK	_	CF	POSCEN	SOSCEN	OSWEN	Note 2
CLKDIV	0102	ROI	DOZE2	DOZE1	DOZE0	DOZEN	RCDIV2	RCDIV1	RCDIV0	CPDIV1	CPDIV0	PLLEN	_	_	_	_	_	0100
OSCTUN	0106	STEN	_	STSIDL	STSRC	STLOCK	STLPOL	STOR	STORPOL	_	_	TUN5	TUN4	TUN3	TUN2	TUN1	TUN0	0000
REFOCONL	0168	ROEN	_	ROSIDL	ROOUT	ROSLP	_	ROSWEN	ROACTIVE	_	_	1	_	ROSEL3	ROSEL2	ROSEL1	ROSEL0	0000
REFOCONH	016A	_							RODIV	<14:0>								0000
REFOTRIML	016C				RC	OTRIM<15:7:	>				_	1	_	_	_	_	_	0000
HLVDCON	010C	HLVDEN	_	LSIDL	_	_	_	_	_	VDIR	BGVST	IRVST	_	HLVDL3	HLVDL2	HLVDL1	HLVDL0	0000
RCON2	010A	_	_	-	_	_	_	_	_	_	_	_	r	VDDBOR	VDDPOR	VBPOR	VBAT	Note 1

Legend: — = unimplemented, read as '0'; r = reserved bit. Reset values are shown in hexadecimal.

Note 1: The Reset value of the RCON register is dependent on the type of Reset event. For more information, refer to Section 7.0 "Resets".

2: The Reset value of the OSCCON register is dependent on both the type of Reset event and the device configuration. For more information, refer to Section 9.0 "Oscillator Configuration".

TABLE 4-30: DEEP SLEEP REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
DSCON	010E	DSEN	_	_	_	_	_	_	_	_	_	_	_	_	r	DSBOR	RELEASE	0000(1)
DSWAKE	0110	_	-	_	_	-	1	_	DSINT0	DSFLT	_	_	DSWDT	DSRTCC	DSMCLR	_	_	0000(1)
DSGPR0	0112							Deep SI	eep Semap	hore Data	0 Register							0000(1)
DSGPR1	0114							Deep SI	eep Semap	hore Data	1 Register							0000(1)

Legend: — = unimplemented, read as '0'; r = reserved bit. Reset values are shown in hexadecimal.

Note 1: These registers are only reset on a VDD POR event.

TABLE 4-31: CRYPTOGRAPHIC ENGINE REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
CRYCONL	01A4	CRYON	_	CRYSIDL	ROLLIE	DONEIE	FREEIE	_	CRYGO	OPMOD3	OPMOD2	OPMOD1	OPMOD0	CPHRSEL	CPHRMOD2	CPHRMOD1	CPHRMOD0	0000
CRYCONH	01A6	_	CTRSIZE6	CTRSIZE5	CTRSIZE4	CTRSIZE3	CTRSIZE2	CTRSIZE1	CTRSIZE0	SKEYSEL	KEYMOD1	KEYMOD0	-	KEYSRC3	KEYSRC2	KEYSRC1	KEYSRC0	0000
CRYSTAT	01A8	_	_	_	_	_	_	_	_	CRYBSY	TXTABSY	CRYABRT	ROLLOVR	_	MODFAIL	KEYFAIL	PGMFAIL	0000
CRYOTP	01AC	_	_	-	_	-	-	_	_	PGMTST	OTPIE	CRYREAD	KEYPG3	KEYPG2	KEYPG1	KEYPG0	CRYWR	0020
CRYTXTA	01B0							Crypt	tographic Te	kt Register A	(128 bits wi	de)						xxxx
CRYKEY	01C0							Cryptogra	aphic Key Re	egister (256 b	oits wide, writ	te-only)						xxxx
CRYTXTB	01E0		•			•	•	Crypt	tographic Te	kt Register B	(128 bits wi	de)	•					xxxx
CRYTXTC	01F0			·				Crypt	tographic Te	kt Register C	(128 bits wi	de)		·	<u>'</u>	<u>'</u>	<u>'</u>	xxxx

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

TABLE 4-32: NVM REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
NVMCON	0760	WR	WREN	WRERR	_	_	_	_	_	_	ERASE	_	_	NVMOP3	NVMOP2	NVMOP1	NVMOP0	0000(1)
NVMKEY	0766	_	_	_	_	_	_	_	-			١	NVMKEY R	egister<7:0	>			0000

Legend: — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

Note 1: The Reset value shown is for POR only. The value on other Reset states is dependent on the state of the memory write or erase operations at the time of Reset.

TABLE 4-33: PERIPHERAL MODULE DISABLE (PMD) REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
PMD1	0170	T5MD	T4MD	T3MD	T2MD	T1MD	_	_	_	I2C1MD	U2MD	U1MD	SPI2MD	SPI1MD	_	1	ADC1MD	0000
PMD2	0172	-	-	IC6MD	IC5MD	IC4MD	IC3MD	IC2MD	IC1MD	_	_	OC6MD	OC5MD	OC4MD	OC3MD	OC2MD	OC1MD	0000
PMD3	0174	-	-	_	_	DSMMD	CMPMD	RTCCMD	PMPMD	CRCMD	_	_	_	U3MD	_	I2C2MD	_	0000
PMD4	0176	_	_	_	ı	_	_	_	_	_	UPWMMD	U4MD	_	REFOMD	CTMUMD	HLVDMD	USB1MD	0000
PMD6	017A	_	_	_	ı	_	_	_	_	_	_	_	_	_	_	ı	SPI3MD	0000
PMD7	017C	_	_	_	ı	_	_	_	_	_	_	DMA1MD	DMA0MD	_	_	ı	-	0000
PMD8	017E		_	_		_	_	_	_	_	_	_	_	_	_	_	CRYMD	0000

4.2.5 EXTENDED DATA SPACE (EDS)

The Extended Data Space (EDS) allows PIC24F devices to address a much larger range of data than would otherwise be possible with a 16-bit address range. EDS includes any additional internal data memory not directly accessible by the lower 32-Kbyte data address space and any external memory through the Enhanced Parallel Master Port (EPMP).

In addition, EDS also allows read access to the program memory space. This feature is called Program Space Visibility (PSV) and is discussed in detail in Section 4.3.3 "Reading Data from Program Memory Using EDS".

Figure 4-4 displays the entire EDS space. The EDS is organized as pages, called EDS pages, with one page equal to the size of the EDS window (32 Kbytes). A particular EDS page is selected through the Data Space Read register (DSRPAG) or Data Space Write register (DSWPAG). For PSV, only the DSRPAG register is used. The combination of the DSRPAG register value and the 16-bit wide data address forms a 24-bit Effective Address (EA).

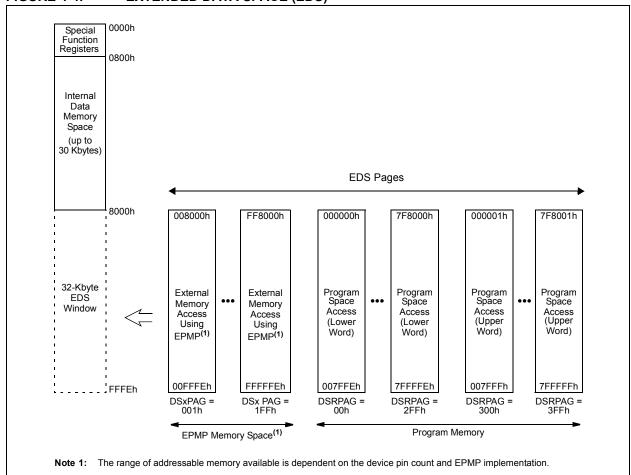
The data addressing range of PIC24FJ128GB204 family devices depends on the version of the Enhanced Parallel Master Port implemented on a particular device; this is, in turn, a function of device pin count. Table 4-34 lists the total memory accessible by each of the devices in this family. For more details on accessing external memory using EPMP, refer to the "dsPIC33/PIC24 Family Reference Manual", "Enhanced Parallel Master Port (EPMP)" (DS39730).

TABLE 4-34: TOTAL ACCESSIBLE DATA MEMORY

Family	Internal RAM	External RAM Access Using EPMP	
PIC24FJXXXGB204	8K	Up to 16 Mbytes	
PIC24FJXXXGB202	8K	Up to 64K	

Note: Accessing Page 0 in the EDS window will generate an address error trap as Page 0 is the base data memory (data locations, 0800h to 7FFFh, in the lower Data Space).





4.2.5.1 Data Read from EDS

In order to read the data from the EDS space first, an Address Pointer is set up by loading the required EDS page number into the DSRPAG register and assigning the offset address to one of the W registers. Once the above assignment is done, the EDS window is enabled by setting bit 15 of the Working register assigned with the offset address; then, the contents of the pointed EDS location can be read.

Figure 4-5 illustrates how the EDS space address is generated for read operations.

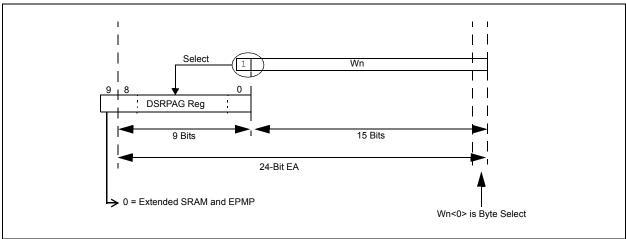
When the Most Significant bit of the EA is '1' and DSRPAG<9> = 0, the lower 9 bits of DSRPAG are concatenated to the lower 15 bits of the EA to form a 24-bit EDS space address for read operations.

Example 4-1 shows how to read a byte, word and double-word from EDS.

Note:

All read operations from EDS space have an overhead of one instruction cycle. Therefore, a minimum of two instruction cycles is required to complete an EDS read. EDS reads under the REPEAT instruction; the first two accesses take three cycles and the subsequent accesses take one cycle.

FIGURE 4-5: EDS ADDRESS GENERATION FOR READ OPERATIONS



EXAMPLE 4-1: EDS READ CODE IN ASSEMBLY

```
; Set the EDS page from where the data to be read
             #0x0002, w0
   mov
   mov
              w0, DSRPAG ;page 2 is selected for read
             \#0x0800, w1 ;select the location (0x800) to be read
   mov
             w1, #15
                          ; set the MSB of the base address, enable EDS mode
   bset
; Read a byte from the selected location
   mov.b
          [w1++], w2 ; read Low byte
             [w1++], w3
                            ;read High byte
   mov.b
; Read a word from the selected location
             [w1], w2
; Read Double - word from the selected location
   mov.d
             [w1], w2
                       ;two word read, stored in w2 and w3
```

4.2.5.2 Data Write into EDS

In order to write data to EDS space, such as in EDS reads, an Address Pointer is set up by loading the required EDS page number into the DSWPAG register and assigning the offset address to one of the W registers. Once the above assignment is done, then the EDS window is enabled by setting bit 15 of the Working register, assigned with the offset address, and the accessed location can be written.

Figure 4-2 illustrates how the EDS space address is generated for write operations.

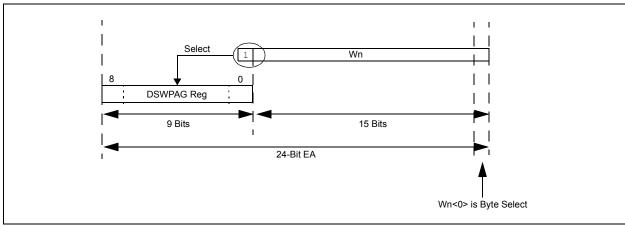
When the MSBs of EA are '1', the lower 9 bits of DSWPAG are concatenated to the lower 15 bits of the EA to form a 24-bit EDS address for write operations. Example 4-2 shows how to write a byte, word and double-word to EDS.

The Data Space Page registers (DSRPAG/DSWPAG) do not update automatically while crossing a page boundary when the rollover happens from 0xFFFF to

0x8000. While developing code in assembly, care must be taken to update the Data Space Page registers when an Address Pointer crosses the page boundary. The 'C' compiler keeps track of the addressing, and increments or decrements the Page registers accordingly, while accessing contiguous data memory locations.

- **Note 1:** All write operations to EDS are executed in a single cycle.
 - 2: Use of Read-Modify-Write operation on any EDS location under a REPEAT instruction is not supported. For example, BCLR, BSW, BTG, RLC f, RLNC f, RRC f, RRNC f, ADD f, SUB f, SUBR f, AND f, IOR f, XOR f, ASR f, ASL f.
 - **3:** Use the DSRPAG register while performing Read-Modify-Write operations.

FIGURE 4-6: EDS ADDRESS GENERATION FOR WRITE OPERATIONS



EXAMPLE 4-2: EDS WRITE CODE IN ASSEMBLY

```
; Set the EDS page where the data to be written
   mov
          #0x0002, w0
   mov
          w0, DSWPAG
                         ;page 2 is selected for write
          \#0x0800, w1 ;select the location (0x800) to be written
   mov
          w1, #15
                         ; set the MSB of the base address, enable EDS mode
   bset
;Write a byte to the selected location
   mov
        #0x00A5, w2
          #0x003C, w3
   mov.
   mov.b w2, [w1++]
                       ;write Low byte
   mov.b w3, [w1++]
                        ;write High byte
;Write a word to the selected location
         #0x1234, w2 ;
   mov.
   mov
          w2, [w1]
                         ;
;Write a Double - word to the selected location
        #0x1122, w2
   mov.
          #0x4455, w3
   mov
   mov.d w2, [w1]
                         ;2 EDS writes
```

TABLE 4-35: EDS MEMORY ADDRESS WITH DIFFERENT PAGES AND ADDRESSES

DSRPAG (Data Space Read Register)	DSWPAG (Data Space Write Register)	Source/Destination Address while Indirect Addressing	24-Bit EA Pointing to EDS	Comment
x ⁽¹⁾	x ⁽¹⁾	0000h to 1FFFh	000000h to 001FFFh	Near Data Space ⁽²⁾
X. 7	χ.,	2000h to 7FFFh	002000h to 007FFFh	
001h	001h		008000h to 00FFFEh	
002h	002h		010000h to 017FFEh	
003h	003h		018000h to 0187FEh	
•	•	8000h to FFFFh	•	EPMP Memory Space
•	•		•	
•	•		•	
•	•		•	
•	•		FF8000h to	
1FFh	1FFh		FFFFEh	
000h	000h		Invalid Address	Address Error Trap ⁽³⁾

- Note 1: If the source/destination address is below 8000h, the DSRPAG and DSWPAG registers are not considered.
 - 2: This Data Space can also be accessed by Direct Addressing.
 - **3:** When the source/destination address is above 8000h and DSRPAG/DSWPAG are '0', an address error trap will occur.

4.2.6 SOFTWARE STACK

Apart from its use as a Working register, the W15 register in PIC24F devices is also used as a Software Stack Pointer (SSP). The pointer always points to the first available free word and grows from lower to higher addresses. It predecrements for stack pops and post-increments for stack pushes, as shown in Figure 4-7. Note that for a PC push during any CALL instruction, the MSB of the PC is zero-extended before the push, ensuring that the MSB is always clear.

Note: A PC push during exception processing will concatenate the SRL Register to the MSB of the PC prior to the push.

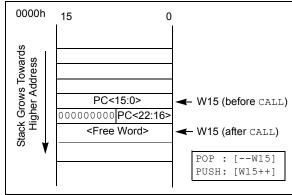
The Stack Pointer Limit Value (SPLIM) register, associated with the Stack Pointer, sets an upper address boundary for the stack. SPLIM is uninitialized at Reset. As is the case for the Stack Pointer, SPLIM<0> is forced to '0' as all stack operations must be word-aligned. Whenever an EA is generated using W15 as a Source or Destination Pointer, the resulting address is compared with the value in SPLIM. If the contents of the Stack Pointer (W15) and the SPLIM register are equal, and a push operation is performed, a stack error trap will not occur. The stack error trap will occur on a subsequent push operation. Thus, for

example, if it is desirable to cause a stack error trap when the stack grows beyond address, 2000h in RAM, initialize the SPLIM with the value, 1FFEh.

Similarly, a Stack Pointer underflow (stack error) trap is generated when the Stack Pointer address is found to be less than 0800h. This prevents the stack from interfering with the SFR space.

A write to the SPLIM register should not be immediately followed by an indirect read operation using W15.

FIGURE 4-7: CALL STACK FRAME



4.3 Interfacing Program and Data Memory Spaces

The PIC24F architecture uses a 24-bit wide program space and 16-bit wide Data Space. The architecture is also a modified Harvard scheme, meaning that data can also be present in the program space. To use this data successfully, it must be accessed in a way that preserves the alignment of information in both spaces.

Aside from normal execution, the PIC24F architecture provides two methods by which program space can be accessed during operation:

- Using table instructions to access individual bytes or words anywhere in the program space
- Remapping a portion of the program space into the Data Space (Program Space Visibility)

Table instructions allow an application to read or write to small areas of the program memory. This makes the method ideal for accessing data tables that need to be updated from time to time. It also allows access to all bytes of the program word. The remapping method allows an application to access a large block of data on a read-only basis, which is ideal for look-ups from a large table of static data. It can only access the least significant word of the program word.

4.3.1 ADDRESSING PROGRAM SPACE

Since the address ranges for the data and program spaces are 16 and 24 bits, respectively, a method is needed to create a 23-bit or 24-bit program address from 16-bit data registers. The solution depends on the interface method to be used.

For table operations, the 8-bit Table Memory Page Address register (TBLPAG) is used to define a 32K word region within the program space. This is concatenated with a 16-bit EA to arrive at a full 24-bit program space address. In this format, the MSbs of TBLPAG are used to determine if the operation occurs in the user memory (TBLPAG<7> = 0) or the configuration memory (TBLPAG<7> = 1).

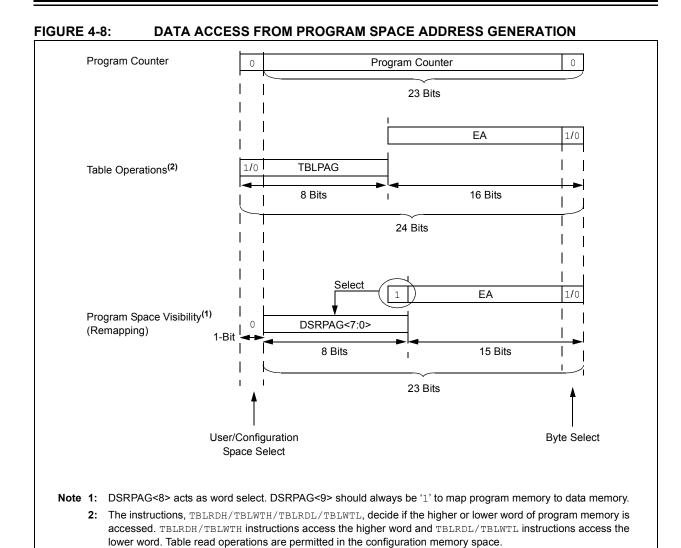
For remapping operations, the 10-bit Extended Data Space Read register (DSRPAG) is used to define a 16K word page in the program space. When the Most Significant bit (MSb) of the EA is '1', and the MSb (bit 9) of DSRPAG is '1', the lower 8 bits of DSRPAG are concatenated with the lower 15 bits of the EA to form a 23-bit program space address. The DSRPAG<8> bit decides whether the lower word (when the bit is '0') or the higher word (when the bit is '1') of program memory is mapped. Unlike table operations, this strictly limits remapping operations to the user memory area.

Table 4-36 and Figure 4-8 show how the program EA is created for table operations and remapping accesses from the data EA. Here, P<23:0> refer to a program space word, whereas D<15:0> refer to a Data Space word.

TABLE 4-36.	PROGRAM SPACE ADDRESS CONSTRUCTION

Access Time	Access	Program Space Address						
Access Type	Space	<23>	<22:16>	<15>	<14:1>	<0>		
Instruction Access	User	0	PC<22:1>			0		
(Code Execution)		0xx xxxx xxxx xxxx xxxx xxx0						
TBLRD/TBLWT	User	ТВ	LPAG<7:0>	Data EA<15:0>				
(Byte/Word Read/Write)		02	xxx xxxx	xxxx xxxx xxxx xxxx				
	Configuration	ТВ	LPAG<7:0>	Data EA<15:0>				
		1:	xxx xxxx	xxxx xxxx xxxx xxxx				
Program Space Visibility	User	0	DSRPAG<7:	0> ⁽²⁾ Data EA<14:0> ⁽¹⁾		:0> ⁽¹⁾		
(Block Remap/Read)		0	XXXX XXXX		XXX XXXX XXXX XXXX			

- **Note 1:** Data EA<15> is always '1' in this case, but is not used in calculating the program space address. Bit 15 of the address is DSRPAG<0>.
 - 2: DSRPAG<9> is always '1' in this case. DSRPAG<8> decides whether the lower word or higher word of program memory is read. When DSRPAG<8> is '0', the lower word is read and when it is '1', the higher word is read.



DS30005009C-page 66

4.3.2 DATA ACCESS FROM PROGRAM MEMORY USING TABLE INSTRUCTIONS

The TBLRDL and TBLWTL instructions offer a direct method of reading or writing the lower word of any address within the program space without going through Data Space. The TBLRDH and TBLWTH instructions are the only method to read or write the upper 8 bits of a program space word as data.

The PC is incremented by two for each successive 24-bit program word. This allows program memory addresses to directly map to Data Space addresses. Program memory can thus be regarded as two, 16-bit word-wide address spaces, residing side by side, each with the same address range. TBLRDL and TBLWTL access the space which contains the least significant data word, and TBLRDH and TBLWTH access the space which contains the upper data byte.

Two table instructions are provided to move byte or word-sized (16-bit) data to and from program space. Both function as either byte or word operations.

TBLRDL (Table Read Low): In Word mode, it
maps the lower word of the program space
location (P<15:0>) to a data address (D<15:0>).
In Byte mode, either the upper or lower byte of
the lower program word is mapped to the lower
byte of a data address. The upper byte is
selected when byte select is '1'; the lower byte
is selected when it is '0'.

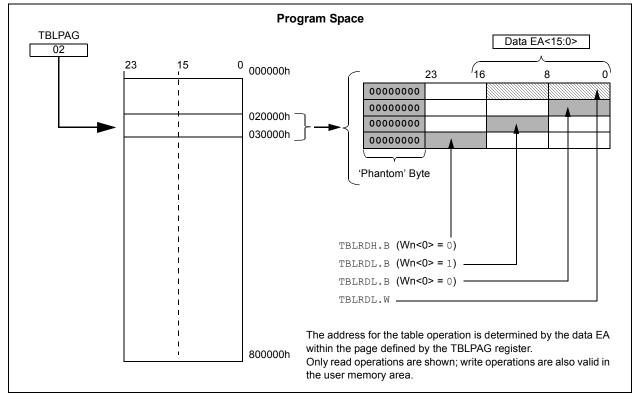
2. TBLRDH (Table Read High): In Word mode, it maps the entire upper word of a program address (P<23:16>) to a data address. Note that D<15:8>, the 'phantom' byte, will always be '0'. In Byte mode, it maps the upper or lower byte of the program word to D<7:0> of the data address, as above. Note that the data will always be '0' when the upper 'phantom' byte is selected (Byte Select = 1).

In a similar fashion, two table instructions, TBLWTH and TBLWTL, are used to write individual bytes or words to a program space address. The details of their operation are described in Section 6.0 "Flash Program Memory".

For all table operations, the area of program memory space to be accessed is determined by the Table Memory Page Address register (TBLPAG). TBLPAG covers the entire program memory space of the device, including user and configuration spaces. When TBLPAG<7> = 0, the table page is located in the user memory space. When TBLPAG<7> = 1, the page is located in configuration space.

Note: Only Table Read operations will execute in the configuration memory space where Device IDs are located; Table Write operations are not allowed.

FIGURE 4-9: ACCESSING PROGRAM MEMORY WITH TABLE INSTRUCTIONS



4.3.3 READING DATA FROM PROGRAM MEMORY USING EDS

The upper 32 Kbytes of Data Space may optionally be mapped into any 16K word page of the program space. This provides transparent access of stored constant data from the Data Space without the need to use special instructions (i.e., $\mathtt{TBLRDL/H}$).

Program space access through the Data Space occurs when the MSb of EA is '1' and the DSRPAG<9> bit is also '1'. The lower 8 bits of DSRPAG are concatenated to the Wn<14:0> bits to form a 23-bit EA to access program memory. The DSRPAG<8> bit decides which word should be addressed; when the bit is '0', the lower word and when '1', the upper word of the program memory is accessed.

The entire program memory is divided into 512 EDS pages, from 200h to 3FFh, each consisting of 16K words of data. Pages, 200h to 2FFh, correspond to the lower words of the program memory, while 300h to 3FFh correspond to the upper words of the program memory.

Using this EDS technique, the entire program memory can be accessed. Previously, the access to the upper word of the program memory was not supported.

Table 4-37 provides the corresponding 23-bit EDS address for program memory with EDS page and source addresses.

For operations that use PSV and are executed outside a REPEAT loop, the MOV and MOV. D instructions will require one instruction cycle in addition to the specified execution time. All other instructions will require two instruction cycles in addition to the specified execution time

For operations that use PSV, which are executed inside a REPEAT loop, there will be some instances that require two instruction cycles in addition to the specified execution time of the instruction:

- · Execution in the first iteration
- · Execution in the last iteration
- Execution prior to exiting the loop due to an interrupt
- Execution upon re-entering the loop after an interrupt is serviced

Any other iteration of the REPEAT loop will allow the instruction accessing data, using PSV, to execute in a single cycle.

TΔRI F 4-37·	FDS PROGRAM	ADDRESS WITH DIFFERENT F	PAGES AND ADDRESSES

DSRPAG (Data Space Read Register)	Source Address while Indirect Addressing	23-Bit EA Pointing to EDS	Comment
200h		000000h to 007FFEh	Lower words of 4M program
•		•	instructions (8 Mbytes); for
•		•	read operations only
•		•	
2FFh		7F8000h to 7FFFEh	
300h	8000h to FFFFh	000001h to 007FFFh	Upper words of 4M program
•		•	instructions (4 Mbytes remaining,
•		•	4 Mbytes are phantom bytes); for
•		•	read operations only
3FFh		7F8001h to 7FFFFFh	
000h		Invalid Address	Address error trap ⁽¹⁾

Note 1: When the source/destination address is above 8000h and DSRPAG/DSWPAG is '0', an address error trap will occur.

EXAMPLE 4-3: EDS READ CODE FROM PROGRAM MEMORY IN ASSEMBLY

```
; Set the EDS page from where the data to be read
          #0x0202, w0
   mov
          w0, DSRPAG
                                    ;page 0x202, consisting lower words, is selected for read
          #0x000A, w1
                                    ; select the location (0x0A) to be read
   mov.
                                    ; set the MSB of the base address, enable EDS mode
   bset. w1, #15
; Read a byte from the selected location
   mov.b [w1++], w2 ; read Low byte
   mov.b [w1++], w3
                                   ;read High byte
; Read a word from the selected location
         [w1], w2
; Read Double - word from the selected location
   mov.d [w1], w2
                                    ; two word read, stored in w2 and w3
```

FIGURE 4-10: PROGRAM SPACE VISIBILITY OPERATION TO ACCESS LOWER WORD

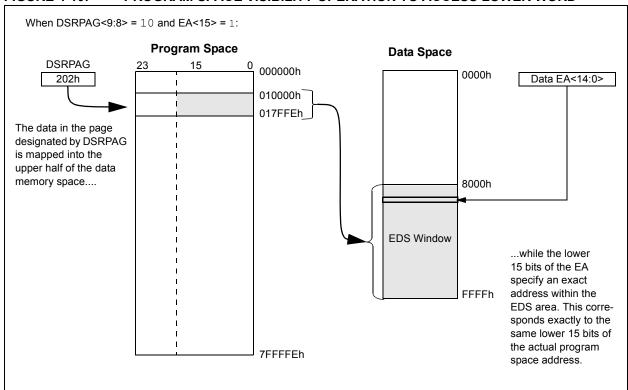
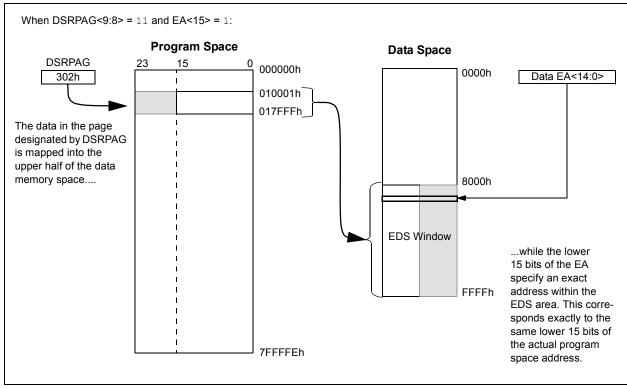


FIGURE 4-11: PROGRAM SPACE VISIBILITY OPERATION TO ACCESS UPPER WORD



NOTES:

5.0 DIRECT MEMORY ACCESS CONTROLLER (DMA)

Note:

This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the "dsPIC33/PIC24 Family Reference Manual", "Direct Memory Access Controller (DMA)" (DS39742). The information in this data sheet supersedes the information in the FRM.

The Direct Memory Access Controller (DMA) is designed to service high data throughput peripherals operating on the SFR bus, allowing them to access data memory directly and alleviating the need for CPU-intensive management. By allowing these data-intensive peripherals to share their own data path, the main data bus is also deloaded, resulting in additional power savings.

The DMA Controller functions both as a peripheral and a direct extension of the CPU. It is located on the microcontroller data bus between the CPU and DMA-enabled peripherals, with direct access to SRAM. This partitions the SFR bus into two buses, allowing the DMA Controller access to the DMA capable peripherals located on the new DMA SFR bus. The controller serves as a master device on the DMA SFR bus, controlling data flow from DMA capable peripherals.

The controller also monitors CPU instruction processing directly, allowing it to be aware of when the CPU requires access to peripherals on the DMA bus and automatically relinquishing control to the CPU as needed. This increases the effective bandwidth for handling data without DMA operations causing a processor stall. This makes the controller essentially transparent to the user.

The DMA Controller has these features:

- Six multiple independent and independently programmable channels
- Concurrent operation with the CPU (no DMA caused Wait states)
- · DMA bus arbitration
- · Five Programmable Address modes
- · Four Programmable Transfer modes
- · Four Flexible Internal Data Transfer modes
- · Byte or word support for data transfer
- 16-Bit Source and Destination Address register for each channel, dynamically updated and reloadable
- 16-Bit Transaction Count register, dynamically updated and reloadable
- Upper and Lower Address Limit registers
- · Counter half-full level interrupt
- · Software-triggered transfer
- · Null Write mode for symmetric buffer operations

A simplified block diagram of the DMA Controller is shown in Figure 5-1.

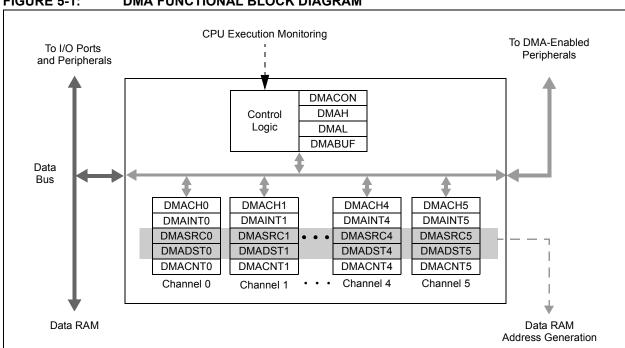


FIGURE 5-1: DMA FUNCTIONAL BLOCK DIAGRAM

5.1 Summary of DMA Operations

The DMA Controller is capable of moving data between addresses according to a number of different parameters. Each of these parameters can be independently configured for any transaction. In addition, any or all of the DMA channels can independently perform a different transaction at the same time. Transactions are classified by these parameters:

- · Source and destination (SFRs and data RAM)
- Data size (byte or word)
- · Trigger source
- Transfer mode (One-Shot, Repeated or Continuous)
- Addressing modes (Fixed Address or Address Blocks with or without Address Increment/Decrement)

In addition, the DMA Controller provides channel priority arbitration for all channels.

5.1.1 SOURCE AND DESTINATION

Using the DMA Controller, data may be moved between any two addresses in the Data Space. The SFR space (0000h to 07FFh), or the data RAM space (0800h to FFFFh) can serve as either the source or the destination. Data can be moved between these areas in either direction, or between addresses in either area. The four different combinations are shown in Figure 5-2.

If it is necessary to protect areas of data RAM, the DMA Controller allows the user to set upper and lower address boundaries for operations in the Data Space above the SFR space. The boundaries are set by the DMAH and DMAL High/Low Address Limit registers. If a DMA channel attempts an operation outside of the address boundaries, the transaction is terminated and an interrupt is generated.

5.1.2 DATA SIZE

The DMA Controller can handle both 8-bit and 16-bit transactions. Size is user-selectable using the SIZE bit (DMACHn<1>). By default, each channel is configured for word-size transactions. When byte-size transactions are chosen, the LSb of the source and/or destination address determines if the data represents the upper or lower byte of the data RAM location.

5.1.3 TRIGGER SOURCE

The DMA Controller can use 63 of the device's interrupt sources to initiate a transaction. The DMA trigger sources occur in reverse order than their natural interrupt priority and are shown in Table 5-1.

These sources cannot be used as DMA triggers:

- · Input Capture 8 and 9
- · Output Compare 7, 8 and 9
- USB

Since the source and destination addresses for any transaction can be programmed independently of the trigger source, the DMA Controller can use any trigger to perform an operation on any peripheral. This also allows DMA channels to be cascaded to perform more complex transfer operations.

5.1.4 TRANSFER MODE

The DMA Controller supports four types of data transfers, based on the volume of data to be moved for each trigger:

- One-Shot: A single transaction occurs for each trigger.
- Continuous: A series of back-to-back transactions occur for each trigger; the number of transactions is determined by the DMACNTn transaction counter.
- Repeated One-Shot: A single transaction is performed repeatedly, once per trigger, until the DMA channel is disabled.
- Repeated Continuous: A series of transactions are performed repeatedly, one cycle per trigger, until the DMA channel is disabled.

All Transfer modes allow the option to have the source and destination addresses, and counter value automatically reloaded after the completion of a transaction; Repeated mode transfers do this automatically.

5.1.5 ADDRESSING MODES

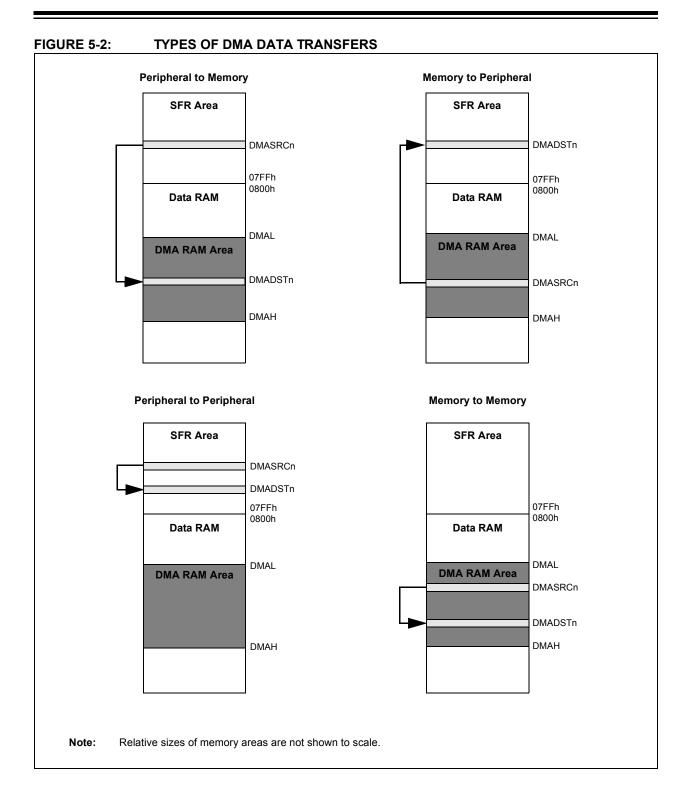
The DMA Controller also supports transfers between single addresses or address ranges. The four basic options are:

- Fixed-to-Fixed: Between two constant addresses
- Fixed-to-Block: From a constant source address to a range of destination addresses
- Block-to-Fixed: From a range of source addresses to a single, constant destination address
- Block-to-Block: From a range of source addresses to a range of destination addresses

The option to select auto-increment or auto-decrement of source and/or destination addresses is available for Block Addressing modes.

In addition to the four basic modes, the DMA Controller also supports Peripheral Indirect Addressing (PIA) mode, where the source or destination address is generated jointly by the DMA Controller and a PIA capable peripheral. When enabled, the DMA channel provides a base source and/or destination address, while the peripheral provides a fixed range offset address.

For PIC24FJ128GB204 family devices, the 12-bit A/D Converter module is the only PIA capable peripheral. Details for its use in PIA mode are provided in Section 25.0 "12-Bit A/D Converter with Threshold Detect".



5.1.6 CHANNEL PRIORITY

Each DMA channel functions independently of the others, but also competes with the others for access to the data and DMA buses. When access collisions occur, the DMA Controller arbitrates between the channels using a user-selectable priority scheme. Two schemes are available:

- Round Robin: When two or more channels collide, the lower numbered channel receives priority on the first collision. On subsequent collisions, the higher numbered channels each receive priority based on their channel number.
- Fixed Priority: When two or more channels collide, the lowest numbered channel always receives priority, regardless of past history.

5.2 Typical Setup

To set up a DMA channel for a basic data transfer:

- 1. Enable the DMA Controller (DMAEN = 1) and select an appropriate channel priority scheme by setting or clearing PRSSEL.
- Program DMAH and DMAL with appropriate upper and lower address boundaries for data RAM operations.
- Select the DMA channel to be used and disable its operation (CHEN = 0).
- 4. Program the appropriate source and destination addresses for the transaction into the channel's DMASRCn and DMADSTn registers. For PIA Addressing mode, use the base address value.
- Program the DMACNTn register for the number of triggers per transfer (One-Shot or Continuous modes) or the number of words (bytes) to be transferred (Repeated modes).
- 6. Set or clear the SIZE bit to select the data size.
- Program the TRMODEx bits to select the Data Transfer mode.
- Program the SAMODEx and DAMODEx bits to select the addressing mode.
- 9. Enable the DMA channel by setting CHEN.
- 10. Enable the trigger source interrupt.

5.3 Peripheral Module Disable

Unlike other peripheral modules, the channels of the DMA Controller cannot be individually powered down using the Peripheral Module Disable x (PMDx) registers. Instead, the channels are controlled as two groups. The DMA0MD bit (PMD7<4>) selectively controls DMACH0 through DMACH3. The DMA1MD bit (PMD7<5>) controls DMACH4 and DMACH5. Setting both bits effectively disables the DMA Controller.

5.4 Registers

The DMA Controller uses a number of registers to control its operation. The number of registers depends on the number of channels implemented for a particular device.

There are always four module-level registers (one control and three buffer/address):

- DMACON: DMA Control Register (Register 5-1)
- DMAH and DMAL: DMA High and Low Address Limit Registers
- · DMABUF: DMA Transfer Data Buffer

Each of the DMA channels implements five registers (two control and three buffer/address):

- DMACHn: DMA Channel n Control Register (Register 5-2)
- DMAINTn: DMA Channel n Interrupt Control Register (Register 5-3)
- DMASRCn: DMA Channel Source Address Pointer for Channel n Register
- DMADSTn: DMA Destination Address for Channel n Register
- DMACNTn: DMA Transaction Count for Channel n Register

For PIC24FJ128GB204 family devices, there are a total of 34 registers.

REGISTER 5-1: DMACON: DMA ENGINE CONTROL REGISTER

R/W-0	U-0						
DMAEN	_	_	_	_	_	_	_
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
_	_	_	_	_	_	_	PRSSEL
bit 7							bit 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 15 **DMAEN:** DMA Module Enable bit

1 = Enables module

0 = Disables module and terminates all active DMA operation(s)

bit 14-1 **Unimplemented:** Read as '0'

bit 0 PRSSEL: Channel Priority Scheme Selection bit

1 = Round robin scheme0 = Fixed priority scheme

REGISTER 5-2: DMACHn: DMA CHANNEL n CONTROL REGISTER

U-0	U-0	U-0	r-0	U-0	R/W-0	R/W-0	R/W-0
_	_	_	_	_	NULLW	RELOAD ⁽¹⁾	CHREQ ⁽³⁾
bit 15							bit 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
SAMODE1	SAMODE0	DAMODE1	DAMODE0	TRMODE1	TRMODE0	SIZE	CHEN
bit 7							bit 0

Legend:	r = Reserved bit		
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 15-13 Unimplemented: Read as '0'
bit 12 Reserved: Maintain as '0'
bit 11 Unimplemented: Read as '0'
bit 10 NULLW: Null Write Mode bit

1 = A dummy write is initiated to DMASRCn for every write to DMADSTn

0 = No dummy write is initiated

bit 9 **RELOAD:** Address and Count Reload bit⁽¹⁾

1 = DMASRCn, DMADSTn and DMACNTn registers are reloaded to their previous values upon the start of the next operation

0 = DMASRCn, DMADSTn and DMACNTn are not reloaded on the start of the next operation⁽²⁾

bit 8 CHREQ: DMA Channel Software Request bit (3)

1 = A DMA request is initiated by software; automatically cleared upon completion of a DMA transfer

0 = No DMA request is pending

bit 7-6 **SAMODE<1:0>:** Source Address Mode Selection bits

11 = DMASRCn is used in Peripheral Indirect Addressing and remains unchanged

10 = DMASRCn is decremented based on the SIZE bit after a transfer completion

01 = DMASRCn is incremented based on the SIZE bit after a transfer completion

00 = DMASRCn remains unchanged after a transfer completion

bit 5-4 DAMODE<1:0>: Destination Address Mode Selection bits

11 = DMADSTn is used in Peripheral Indirect Addressing and remains unchanged

10 = DMADSTn is decremented based on the SIZE bit after a transfer completion

01 = DMADSTn is incremented based on the SIZE bit after a transfer completion

00 = DMADSTn remains unchanged after a transfer completion

bit 3-2 TRMODE<1:0>: Transfer Mode Selection bits

11 = Repeated Continuous mode

10 = Continuous mode

01 = Repeated One-Shot mode

00 = One-Shot mode

bit 1 SIZE: Data Size Selection bit

1 = Byte (8-bit) 0 = Word (16-bit)

bit 0 CHEN: DMA Channel Enable bit

1 = The corresponding channel is enabled

0 = The corresponding channel is disabled

Note 1: Only the original DMACNTn is required to be stored to recover the original DMASRCn and DMADSTn values.

2: DMACNTn will always be reloaded in Repeated mode transfers, regardless of the state of the RELOAD bit.

3: The number of transfers executed while CHREQ is set depends on the configuration of TRMODE<1:0>.

REGISTER 5-3: DMAINTn: DMA CHANNEL n INTERRUPT REGISTER

R-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DBUFWF ⁽¹⁾	_	CHSEL5	CHSEL4	CHSEL3	CHSEL2	CHSEL1	CHSEL0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0
HIGHIF ^(1,2)	LOWIF ^(1,2)	DONEIF(1)	HALFIF ⁽¹⁾	OVRUNIF ⁽¹⁾	_	_	HALFEN
bit 7							bit 0

Legend:

R = Readable bit W = Writable bit U = Unimplemented bit, read as '0'

bit 15 **DBUFWF:** DMA Buffered Data Write Flag bit⁽¹⁾

1 = The content of the DMA buffer has not been written to the location specified in DMADSTn or DMASRCn in Null Write mode

0 = The content of the DMA buffer has been written to the location specified in DMADSTn or DMASRCn in Null Write mode

bit 14 Unimplemented: Read as '0'

bit 13-8 CHSEL<5:0>: DMA Channel Trigger Selection bits

See Table 5-1 for a complete list.

bit 7 HIGHIF: DMA High Address Limit Interrupt Flag bit (1,2)

1 = The DMA channel has attempted to access an address higher than DMAH or the upper limit of the data RAM space

0 = The DMA channel has not invoked the high address limit interrupt

bit 6 LOWIF: DMA Low Address Limit Interrupt Flag bit (1,2)

1 = The DMA channel has attempted to access a DMA SFR address lower than DMAL, but above the SFR range (07FFh)

0 = The DMA channel has not invoked the low address limit interrupt

bit 5 **DONEIF:** DMA Complete Operation Interrupt Flag bit⁽¹⁾

If CHEN = 1:

1 = The previous DMA session has ended with completion

0 = The current DMA session has not yet completed

If CHEN = 0:

1 = The previous DMA session has ended with completion

0 = The previous DMA session has ended without completion

bit 4 **HALFIF:** DMA 50% Watermark Level Interrupt Flag bit⁽¹⁾

1 = DMACNTn has reached the halfway point to 0000h

0 = DMACNTn has not reached the halfway point

bit 3 **OVRUNIF:** DMA Channel Overrun Flag bit⁽¹⁾

1 = The DMA channel is triggered while it is still completing the operation based on the previous trigger

0 = The overrun condition has not occurred

bit 2-1 **Unimplemented:** Read as '0'

bit 0 HALFEN: Halfway Completion Watermark bit

1 = Interrupts are invoked when DMACNTn has reached its halfway point and at completion

0 = An interrupt is invoked only at the completion of the transfer

Note 1: Setting these flags in software does not generate an interrupt.

2: Testing for address limit violations (DMASRCn or DMADSTn is either greater than DMAH or less than DMAL) is NOT done before the actual access.

TABLE 5-1: DMA CHANNEL TRIGGER SOURCES

CHSEL<5:0>	Trigger (Interrupt)	CHSEL<5:0>	Trigger (Interrupt)
000000	(Unimplemented)	100000	UART2 Transmit
000001	SPI3 General Event	100001	UART2 Receive
000010	I2C1 Slave Event	100010	External Interrupt 2
000011	UART4 Transmit	100011	Timer5
000100	UART4 Receive	100100	Timer4
000101	UART4 Error	100101	Output Compare 4
000110	UART3 Transmit	100110	Output Compare 3
000111	UART3 Receive	100111	DMA Channel 2
001000	UART3 Error	101000	I2C2 Slave Event
001001	CTMU Event	101001	External Interrupt 1
001010	HLVD	101010	Interrupt-on-Change
001011	CRC Done	101011	Comparators Event
001100	UART2 Error	101100	SPI3 Receive Event
001101	UART1 Error	101101	I2C1 Master Event
001110	RTCC	101110	DMA Channel 1
001111	DMA Channel 5	101111	A/D Converter
010000	External Interrupt 4	110000	UART1 Transmit
010001	External Interrupt 3	110001	UART1 Receive
010010	SPI2 Receive Event	110010	SPI1 Transmit Event
010011	I2C2 Master Event	110011	SPI1 General Event
010100	DMA Channel 4	110100	Timer3
010101	EPMP	110101	Timer2
010110	SPI1 Receive Event	110110	Output Compare 2
010111	Output Compare 6	110111	Input Capture 2
011000	Output Compare 5	111000	DMA Channel 0
011001	Input Capture 6	111001	Timer1
011010	Input Capture 5	111010	Output Compare 1
011011	Input Capture 4	111011	Input Capture 1
011100	Input Capture 3	111100	External Interrupt 0
011101	DMA Channel 3	111101	USB
011110	SPI2 Transmit Event	111110	SPI3 Transmit Event
011111	SPI2 General Event	111111	Crypto Done

6.0 FLASH PROGRAM MEMORY

Note:

This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the "dsPIC33/PIC24 Family Reference Manual", "Program Memory" (DS39715). The information in this data sheet supersedes the information in the FRM.

The PIC24FJ128GB204 family of devices contains internal Flash program memory for storing and executing application code. The program memory is readable, writable and erasable. The Flash memory can be programmed in four ways:

- In-Circuit Serial Programming™ (ICSP™)
- · Run-Time Self-Programming (RTSP)
- JTAG
- · Enhanced In-Circuit Serial Programming (Enhanced ICSP)

ICSP allows a PIC24FJ128GB204 family device to be serially programmed while in the end application circuit. This is simply done with two lines for the programming clock and programming data (named PGECx and PGEDx, respectively), and three other lines for power (VDD), ground (VSS) and Master Clear (MCLR). This allows customers to manufacture boards with unprogrammed devices and then program the

microcontroller just before shipping the product. This also allows the most recent firmware or a custom firmware to be programmed.

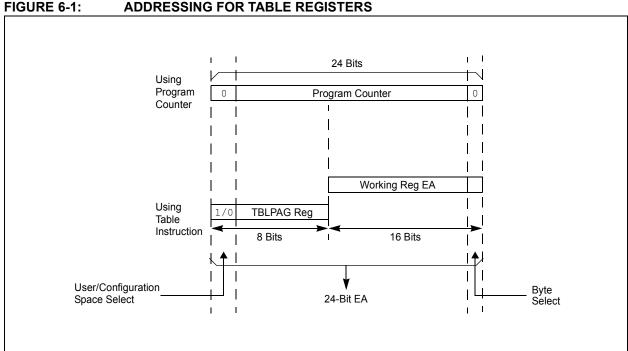
RTSP is accomplished using TBLRD (Table Read) and TBLWT (Table Write) instructions. With RTSP, the user may write program memory data in blocks of 64 instructions (192 bytes) at a time and erase program memory in blocks of 512 instructions (1536 bytes) at a time.

6.1 Table Instructions and Flash **Programming**

Regardless of the method used, all programming of Flash memory is done with the Table Read and Table Write instructions. These allow direct read and write access to the program memory space from the data memory while the device is in normal operating mode. The 24-bit target address in the program memory is formed using the TBLPAG<7:0> bits and the Effective Address (EA) from a W register, specified in the table instruction, as shown in Figure 6-1.

The TBLRDL and the TBLWTL instructions are used to read or write to bits<15:0> of program memory. TBLRDL and TBLWTL can access program memory in both Word and Byte modes.

The TBLRDH and TBLWTH instructions are used to read or write to bits<23:16> of program memory. TBLRDH and TBLWTH can also access program memory in Word or Byte mode.



6.2 RTSP Operation

The PIC24F Flash program memory array is organized into rows of 64 instructions or 192 bytes. RTSP allows the user to erase blocks of eight rows (512 instructions) at a time and to program one row at a time. It is also possible to program single words.

The 8-row erase blocks and single row write blocks are edge-aligned, from the beginning of program memory on boundaries of 1536 bytes and 192 bytes, respectively.

When data is written to program memory using TBLWT instructions, the data is not written directly to memory. Instead, data written using Table Writes is stored in holding latches until the programming sequence is executed.

Any number of TBLWT instructions can be executed and a write will be successfully performed. However, 64 TBLWT instructions are required to write the full row of memory.

To ensure that no data is corrupted during a write, any unused address should be programmed with FFFFFFh. This is because the holding latches reset to an unknown state, so if the addresses are left in the Reset state, they may overwrite the locations on rows which were not rewritten.

The basic sequence for RTSP programming is to set up a Table Pointer, then do a series of TBLWT instructions to load the buffers. Programming is performed by setting the control bits in the NVMCON register.

Data can be loaded in any order and the holding registers can be written to, multiple times, before performing a write operation. Subsequent writes, however, will wipe out any previous writes.

Note: Writing to a location multiple times, without erasing, is not recommended.

All of the Table Write operations are single-word writes (2 instruction cycles), because only the buffers are written. A programming cycle is required for programming each row.

6.3 JTAG Operation

The PIC24F family supports JTAG boundary scan. Boundary scan can improve the manufacturing process by verifying pin to PCB connectivity.

6.4 Enhanced In-Circuit Serial Programming

Enhanced In-Circuit Serial Programming uses an on-board bootloader, known as the Program Executive (PE), to manage the programming process. Using an SPI data frame format, the Program Executive can erase, program and verify program memory. For more information on Enhanced ICSP, see the device programming specification.

6.5 Control Registers

There are two SFRs used to read and write the program Flash memory: NVMCON and NVMKEY.

The NVMCON register (Register 6-1) controls which blocks are to be erased, which memory type is to be programmed and when the programming cycle starts.

NVMKEY is a write-only register that is used for write protection. To start a programming or erase sequence, the user must consecutively write 55h and AAh to the NVMKEY register. For more information, refer to Section 6.6 "Programming Operations".

6.6 Programming Operations

A complete programming sequence is necessary for programming or erasing the internal Flash in RTSP mode. During a programming or erase operation, the processor stalls (waits) until the operation is finished. Setting the WR bit (NVMCON<15>) starts the operation and the WR bit is automatically cleared when the operation is finished.

REGISTER 6-1: NVMCON: FLASH MEMORY CONTROL REGISTER

R/S-0, HC ⁽¹⁾	R/W-0 ⁽¹⁾	R-0, HSC ⁽¹⁾	U-0	U-0	U-0	U-0	U-0
WR	WREN	WRERR	_	_	_	_	_
bit 15							bit 8

U-0	R/W-0 ⁽¹⁾	U-0	U-0	R/W-0 ⁽¹⁾	R/W-0 ⁽¹⁾	R/W-0 ⁽¹⁾	R/W-0 ⁽¹⁾
_	ERASE	_	_	NVMOP3 ⁽²⁾	NVMOP2 ⁽²⁾	NVMOP1 ⁽²⁾	NVMOP0 ⁽²⁾
bit 7							bit 0

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

- bit 15 WR: Write Control bit⁽¹⁾
 - 1 = Initiates a Flash memory program or erase operation; the operation is self-timed and the bit is cleared by hardware once the operation is complete
 - 0 = Program or erase operation is complete and inactive
- bit 14 WREN: Write Enable bit⁽¹⁾
 - 1 = Enables Flash program/erase operations
 - 0 = Inhibits Flash program/erase operations
- bit 13 WRERR: Write Sequence Error Flag bit⁽¹⁾
 - 1 = An improper program or erase sequence attempt, or termination has occurred (bit is set automatically on any set attempt of the WR bit)
 - 0 = The program or erase operation completed normally
- bit 12-7 Unimplemented: Read as '0'
- bit 6 **ERASE**: Erase/Program Enable bit⁽¹⁾
 - 1 = Performs the erase operation specified by the NVMOP<3:0> bits on the next WR command
 - 0 = Performs the program operation specified by the NVMOP<3:0> bits on the next WR command
- bit 5-4 **Unimplemented:** Read as '0'
- bit 3-0 **NVMOP<3:0>:** NVM Operation Select bits^(1,2)
 - 1111 = Memory bulk erase operation (ERASE = 1) or no operation (ERASE = 0)(3)
 - 0011 = Memory word program operation (ERASE = 0) or no operation (ERASE = 1)
 - 0010 = Memory page erase operation (ERASE = 1) or no operation (ERASE = 0)
 - 0001 = Memory row program operation (ERASE = 0) or no operation (ERASE = 1)
- Note 1: These bits can only be reset on a Power-on Reset.
 - 2: All other combinations of NVMOP<3:0> are unimplemented.
 - **3:** Available in ICSP™ mode only; refer to the device programming specification.

6.6.1 PROGRAMMING ALGORITHM FOR FLASH PROGRAM MEMORY

The user can program one row of Flash program memory at a time. To do this, it is necessary to erase the 8-row erase block containing the desired row. The general process is:

- 1. Read eight rows of program memory (512 instructions) and store in data RAM.
- 2. Update the program data in RAM with the desired new data.
- 3. Erase the block (see Example 6-1):
 - a) Set the NVMOPx bits (NVMCON<3:0>) to '0010' to configure for block erase. Set the ERASE (NVMCON<6>) and WREN (NVMCON<14>) bits.
 - Write the starting address of the block to be erased into the TBLPAG and W registers.
 - c) Write 55h to NVMKEY.
 - d) Write AAh to NVMKEY.
 - e) Set the WR bit (NVMCON<15>). The erase cycle begins and the CPU stalls for the duration of the erase cycle. When the erase is done, the WR bit is cleared automatically.

- Write the first 64 instructions from data RAM into the program memory buffers (see Example 6-3).
- 5. Write the program block to Flash memory:
 - Set the NVMOPx bits to '0001' to configure for row programming. Clear the ERASE bit and set the WREN bit.
 - b) Write 55h to NVMKEY.
 - c) Write AAh to NVMKEY.
 - d) Set the WR bit. The programming cycle begins and the CPU stalls for the duration of the write cycle. When the write to Flash memory is done, the WR bit is cleared automatically.
- Repeat Steps 4 and 5, using the next available 64 instructions from the block in data RAM by incrementing the value in TBLPAG, until all 512 instructions are written back to Flash memory.

For protection against accidental operations, the write initiate sequence for NVMKEY must be used to allow any erase or program operation to proceed. After the programming command has been executed, the user must wait for the programming time until programming is complete. The two instructions following the start of the programming sequence should be NOPS, as shown in Example 6-4.

EXAMPLE 6-1: ERASING A PROGRAM MEMORY BLOCK (ASSEMBLY LANGUAGE CODE)

```
; Set up NVMCON for block erase operation
          #0x4042, W0
   MOV
   MOV
          W0, NVMCON
                                            ; Initialize NVMCON
; Init pointer to row to be ERASED
         #tblpage(PROG ADDR), W0
   MOV
                                           ; Initialize Program Memory (PM) Page Boundary SFR
          WO, TBLPAG
   MOV
           #tbloffset(PROG ADDR), W0
                                            ; Initialize in-page EA<15:0> pointer
   TBLWTL WO, [WO]
                                            ; Set base address of erase block
                                            ; Block all interrupts with priority <7
   DIST
          #5
                                            ; for next 5 instructions
   MOV.B #0x55, W0
          WO, NVMKEY
                                            ; Write the 0x55 key
   MOV
   MOV.B #0xAA, W1
          W1, NVMKEY
   MOV
                                            ; Write the OxAA key
   BSET
          NVMCON, #WR
                                            ; Start the erase sequence
   NOP
                                            ; Insert two NOPs after the erase
   NOP
                                             ; command is asserted
```

EXAMPLE 6-2: ERASING A PROGRAM MEMORY BLOCK ('C' LANGUAGE CODE)

EXAMPLE 6-3: LOADING THE WRITE BUFFERS

```
; Set up NVMCON for row programming operations
      MOV
           #0x4001, W0
             W0, NVMCON
                                             ; Initialize NVMCON
; Set up a pointer to the first program memory location to be written
; program memory selected, and writes enabled
      MOV #0x0000, W0
                                             ; Initialize PM Page Boundary SFR
      MOV
           WO, TBLPAG
      MOV #0x6000, W0
                                             ; An example program memory address
; Perform the TBLWT instructions to write the latches
; 0th program word
      MOV #LOW_WORD_0, W2
      MOV
             #HIGH BYTE 0, W3
                                             ;
      TBLWTL W2, [W0]
                                             ; Write PM low word into program latch
      TBLWTH W3, [W0++]
                                             ; Write PM high byte into program latch
; 1st program word
      MOV #LOW WORD 1, W2
           #HIGH_BYTE_1, W3
      MOV
      TBLWTL W2, [W0]
                                             ; Write PM low word into program latch
      TBLWTH W3, [W0++]
                                             ; Write PM high byte into program latch
 2nd program word
      MOV #LOW_WORD_2, W2
             #LOW_WURD_2, w2
#HIGH_BYTE_2, W3
      MOV
                                            ; Write PM low word into program latch
      TBLWTL W2, [W0]
      TBLWTH W3, [W0++]
                                             ; Write PM high byte into program latch
; 63rd program word
      MOV #LOW_WORD_63, W2
             #HIGH BYTE 63, W3
      VOM
      TBLWTL W2, [W0]
                                              ; Write PM low word into program latch
      TBLWTH W3, [W0]
                                              ; Write PM high byte into program latch
```

EXAMPLE 6-4: INITIATING A PROGRAMMING SEQUENCE

```
DIST #5
                               ; Block all interrupts with priority <7
                               ; for next 5 instructions
MOV.B #0x55, W0
MOV WO, NVMKEY
                              ; Write the 0x55 key
MOV.B #0xAA, W1
      W1, NVMKEY
                               ; Write the OxAA key
MOV
BSET
      NVMCON, #WR
                               ; Start the programming sequence
NOP
                               ; Required delays
NOP
BTSC NVMCON, #15
                              ; and wait for it to be
                               ; completed
```

6.6.2 PROGRAMMING A SINGLE WORD OF FLASH PROGRAM MEMORY

If a Flash location has been erased, it can be programmed using Table Write instructions to write an instruction word (24-bit) into the write latch. The TBLPAG register is loaded with the 8 Most Significant Bytes (MSBs) of the Flash address. The TBLWTL and TBLWTH instructions write the desired data into the write latches and specify the lower 16 bits of the

program memory address to write to. To configure the NVMCON register for a word write, set the NVMOPx bits (NVMCON<3:0>) to '0011'. The write is performed by executing the unlock sequence and setting the WR bit (see Example 6-5). An equivalent procedure in 'C' compiler language, using the MPLAB® C30 compiler and built-in hardware functions, is shown in Example 6-6.

EXAMPLE 6-5: PROGRAMMING A SINGLE WORD OF FLASH PROGRAM MEMORY

```
; Setup a pointer to data Program Memory
          #tblpage(PROG_ADDR), W0
   MOV
   VOM
          WO, TBLPAG
                                      ; Initialize PM Page Boundary SFR
   MOV
          #tbloffset(PROG ADDR), W0 ;Initialize a register with program memory address
        #LOW_WORD_N, W2
   VOM
   MOV
          #HIGH BYTE N, W3
   TBLWTL W2, [W0]
                                       ; Write PM low word into program latch
   TBLWTH W3, [W0++]
                                       ; Write PM high byte into program latch
; Setup NVMCON for programming one word to data Program Memory
          #0x4003, W0
          WO, NVMCON
                                      ; Set NVMOP bits to 0011
   MOV
   DISI #5
                                      ; Disable interrupts while the KEY sequence is written
   MOV.B #0x55, W0
                                      ; Write the key sequence
   MOV W0, NVMKEY
   MOV.B #0xAA, WO
   MOV
          WO, NVMKEY
        NVMCON, #WR
                                       ; Start the write cycle
   BSET
   NOP
                                       ; Required delays
   NOP
```

EXAMPLE 6-6: PROGRAMMING A SINGLE WORD OF FLASH PROGRAM MEMORY ('C' LANGUAGE CODE)

```
// C example using MPLAB C30
unsigned int offset;
unsigned long progAddr = 0xXXXXXX;
                                          // Address of word to program
unsigned int progDataL = 0xXXXX;
                                          // Data to program lower word
                                          // Data to program upper byte
unsigned char progDataH = 0xXX;
//Set up NVMCON for word programming
NVMCON = 0x4003;
                                           // Initialize NVMCON
//Set up pointer to the first memory location to be written
TBLPAG = progAddr>>16;
                                           // Initialize PM Page Boundary SFR
offset = progAddr & 0xFFFF;
                                           // Initialize lower word of address
//Perform TBLWT instructions to write latches
__builtin_tblwtl(offset, progDataL); // Write to address low word
                                         // Write to upper byte
 _builtin_tblwth(offset, progDataH);
asm("DISI #5");
                                           // Block interrupts with priority <7
                                           // for next 5 instructions
                                           // C30 function to perform unlock
builtin write NVM();
                                           // sequence and set WR
```

7.0 RESETS

Note:

This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the "dsPIC33/PIC24 Family Reference Manual", "Reset" (DS39712). The information in this data sheet supersedes the information in the FRM.

The Reset module combines all Reset sources and controls the device Master Reset Signal, SYSRST. The following is a list of device Reset sources:

· POR: Power-on Reset

MCLR: Master Clear Pin Reset

• SWR: RESET Instruction

· WDT: Watchdog Timer Reset

· BOR: Brown-out Reset

CM: Configuration Mismatch Reset

· TRAPR: Trap Conflict Reset

· IOPUWR: Illegal Opcode Reset

UWR: Uninitialized W Register Reset

A simplified block diagram of the Reset module is shown in Figure 7-1.

Any active source of Reset will make the SYSRST signal active. Many registers associated with the CPU and peripherals are forced to a known Reset state. Most registers are unaffected by a Reset; their status is unknown on POR and unchanged by all other Resets.

Note:

Refer to the specific peripheral or CPU section of this manual for register Reset states

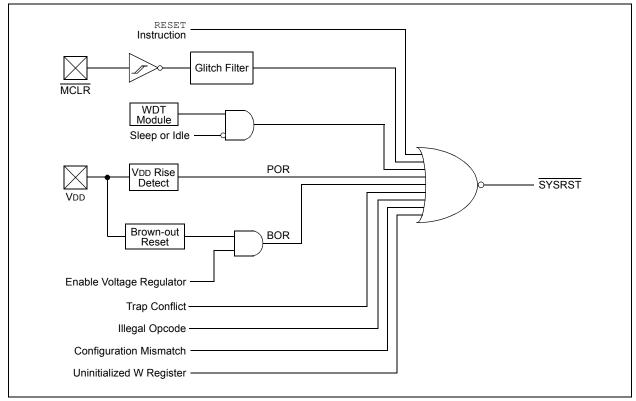
All types of device Reset will set a corresponding status bit in the RCON register to indicate the type of Reset (see Register 7-1). In addition, Reset events occurring while an extreme power-saving feature is in use (such as VBAT) will set one or more status bits in the RCON2 register (Register 7-2). A POR will clear all bits, except for the BOR and POR (RCON<1:0>) bits, which are set. The user may set or clear any bit at any time during code execution. The RCON bits only serve as status bits. Setting a particular Reset status bit in software will not cause a device Reset to occur.

The RCON register also has other bits associated with the Watchdog Timer and device power-saving states. The function of these bits is discussed in other sections of this data sheet.

Note:

The status bits in the RCON registers should be cleared after they are read so that the next RCON register values after a device Reset will be meaningful.

FIGURE 7-1: RESET SYSTEM BLOCK DIAGRAM



REGISTER 7-1: RCON: RESET CONTROL REGISTER

R/W-0	R/W-0	U-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
TRAPR ⁽¹⁾	IOPUWR ⁽¹⁾	_	RETEN ⁽²⁾	_	DPSLP ⁽¹⁾	CM ⁽¹⁾	VREGS ⁽³⁾
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-1
EXTR ⁽¹⁾	SWR ⁽¹⁾	SWDTEN ⁽⁴⁾	WDTO ⁽¹⁾	SLEEP ⁽¹⁾	IDLE ⁽¹⁾	BOR ⁽¹⁾	POR ⁽¹⁾
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Rit is set	'0' = Rit is cleared $y = Rit$ is unknown	

bit 15 **TRAPR:** Trap Reset Flag bit⁽¹⁾

1 = A Trap Conflict Reset has occurred

0 = A Trap Conflict Reset has not occurred

bit 14 IOPUWR: Illegal Opcode or Uninitialized W Access Reset Flag bit⁽¹⁾

1 = An illegal opcode detection, an illegal address mode or Uninitialized W register is used as an Address Pointer and caused a Reset

0 = An illegal opcode or Uninitialized W register Reset has not occurred

bit 13 Unimplemented: Read as '0'

bit 12 **RETEN:** Retention Mode Enable bit⁽²⁾

1 = Retention mode is enabled while device is in Sleep modes (1.2V regulator supplies to the core)

0 = Retention mode is disabled; normal voltage levels are present

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

bit 10 **DPSLP:** Deep Sleep Flag bit⁽¹⁾

1 = Device has been in Deep Sleep mode

0 = Device has not been in Deep Sleep mode

bit 9 **CM:** Configuration Word Mismatch Reset Flag bit⁽¹⁾

1 = A Configuration Word Mismatch Reset has occurred

0 = A Configuration Word Mismatch Reset has not occurred

bit 8 **VREGS:** Program Memory Power During Sleep bit (3)

1 = Program memory bias voltage remains powered during Sleep

0 = Program memory bias voltage is powered down during Sleep

bit 7 **EXTR:** External Reset (MCLR) Pin bit⁽¹⁾

1 = A Master Clear (pin) Reset has occurred

0 = A Master Clear (pin) Reset has not occurred

bit 6 **SWR:** Software Reset (Instruction) Flag bit⁽¹⁾

1 = A RESET instruction has been executed

0 = A RESET instruction has not been executed

Note 1: All of the Reset status bits may be set or cleared in software. Setting one of these bits in software does not cause a device Reset.

- 2: If the LPCFG Configuration bit is '1' (unprogrammed), the retention regulator is disabled and the RETEN bit has no effect.
- 3: Re-enabling the regulator after it enters Standby mode will add a delay, TVREG, when waking up from Sleep. Applications that do not use the voltage regulator should set this bit to prevent this delay from occurring.
- 4: If the FWDTEN Configuration bit is '1' (unprogrammed), the WDT is always enabled, regardless of the SWDTEN bit setting.

REGISTER 7-1: RCON: RESET CONTROL REGISTER (CONTINUED)

SWDTEN: Software Enable/Disable of WDT bit⁽⁴⁾ bit 5 1 = WDT is enabled 0 = WDT is disabled bit 4 **WDTO:** Watchdog Timer Time-out Flag bit⁽¹⁾ 1 = WDT time-out has occurred 0 = WDT time-out has not occurred **SLEEP:** Wake from Sleep Flag bit⁽¹⁾ bit 3 1 = Device has been in Sleep mode 0 = Device has not been in Sleep mode IDLE: Wake from Idle Flag bit(1) bit 2 1 = Device has been in Idle mode 0 = Device has not been in Idle mode BOR: Brown-out Reset Flag bit(1) bit 1 1 = A Brown-out Reset has occurred (also set after a Power-on Reset) 0 = A Brown-out Reset has not occurred **POR:** Power-on Reset Flag bit⁽¹⁾ bit 0

1 = A Power-on Reset has occurred0 = A Power-on Reset has not occurred

cause a device Reset.

- Note 1: All of the Reset status bits may be set or cleared in software. Setting one of these bits in software does not
 - 2: If the LPCFG Configuration bit is '1' (unprogrammed), the retention regulator is disabled and the RETEN bit has no effect.
 - **3:** Re-enabling the regulator after it enters Standby mode will add a delay, TVREG, when waking up from Sleep. Applications that do not use the voltage regulator should set this bit to prevent this delay from occurring.
 - **4:** If the FWDTEN Configuration bit is '1' (unprogrammed), the WDT is always enabled, regardless of the SWDTEN bit setting.

REGISTER 7-2: RCON2: RESET AND SYSTEM CONTROL REGISTER 2

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

U-0	U-0	U-0	r-0	R/CO-1	R/CO-1	R/CO-1	R/CO-0
_	_	_	_	VDDBOR ⁽¹⁾	VDDPOR ^(1,2)	VBPOR ^(1,3)	VBAT ⁽¹⁾
bit 7							bit 0

Legend:	CO = Clearable Only bit	r = Reserved bit	
R = Readable bit	W = Writable bit	U = Unimplemented bit, read	d as '0'
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-5 **Unimplemented:** Read as '0' bit 4 **Reserved:** Maintain as '0'

bit 3 VDDBOR: VDD Brown-out Reset Flag bit⁽¹⁾

1 = A VDD Brown-out Reset has occurred (set by hardware)

0 = A VDD Brown-out Reset has not occurred

bit 2 **VDDPOR:** VDD Power-on Reset Flag bit^(1,2)

1 = A VDD Power-on Reset has occurred (set by hardware)

0 = A VDD Power-on Reset has not occurred

bit 1 **VBPOR:** VBPOR Flag bit^(1,3)

1 = A VBAT POR has occurred (no battery is connected to the VBAT pin or VBAT power below the Deep Sleep Semaphore retention level is set by hardware)

0 = A VBAT POR has not occurred

bit 0 **VBAT:** VBAT Flag bit⁽¹⁾

1 = A POR exit has occurred while power was applied to the VBAT pin (set by hardware)

0 = A POR exit from VBAT has not occurred

Note 1: This bit is set in hardware only; it can only be cleared in software.

2: This bit indicates a VDD Power-on Reset. Setting the POR bit (RCON<0>) indicates a Vcore Power-on Reset.

3: This bit is set when the device is originally powered up, even if power is present on VBAT.

TABLE 7-1: RESET FLAG BIT OPERATION

Flag Bit	Setting Event	Clearing Event
TRAPR (RCON<15>)	Trap Conflict Event	POR
IOPUWR (RCON<14>)	Illegal Opcode or Uninitialized W Register Access	POR
CM (RCON<9>)	Configuration Mismatch Reset	POR
EXTR (RCON<7>)	MCLR Reset	POR
SWR (RCON<6>)	RESET Instruction	POR
WDTO (RCON<4>)	WDT Time-out	CLRWDT, PWRSAV Instructions, POR
SLEEP (RCON<3>)	PWRSAV #0 Instruction	POR
DPSLP (RCON<10>)	PWRSAV #0 Instruction while DSEN bit is Set	POR
IDLE (RCON<2>)	PWRSAV #1 Instruction	POR
BOR (RCON<1>)	POR, BOR	_
POR (RCON<0>)	POR	_

Note: All Reset flag bits may be set or cleared by the user software.

7.1 Special Function Register Reset States

Most of the Special Function Registers (SFRs) associated with the PIC24F CPU and peripherals are reset to a particular value at a device Reset. The SFRs are grouped by their peripheral or CPU function and their Reset values are specified in each section of this manual.

The Reset value for each SFR does not depend on the type of Reset, with the exception of four registers. The Reset value for the Reset Control register, RCON, will depend on the type of device Reset. The Reset value for the Oscillator Control register, OSCCON, will depend on the type of Reset and the programmed values of the FNOSCx bits in Flash Configuration Word 2 (CW2); see Table 7-2. The RCFGCAL and NVMCON registers are only affected by a POR.

7.2 Device Reset Times

The Reset times for various types of device Reset are summarized in Table 7-3. Note that the Master Reset Signal, SYSRST, is released after the POR delay time expires.

The time at which the device actually begins to execute code will also depend on the system oscillator delays, which include the Oscillator Start-up Timer (OST) and the PLL lock time. The OST and PLL lock times occur in parallel with the applicable SYSRST delay times.

The Fail-Safe Clock Monitor (FSCM) delay determines the time at which the FSCM begins to monitor the system clock source after the SYSRST signal is released.

7.3 Brown-out Reset (BOR)

PIC24FJ128GB204 family devices implement a BOR circuit that provides the user with several configuration and power-saving options. The BOR is controlled by the BOREN (CW3<12>) Configuration bit.

When BOR is enabled, any drop of VDD below the BOR threshold results in a device BOR. Threshold levels are described in **Section 33.1 "DC Characteristics"** (Parameter DC17A).

7.4 Clock Source Selection at Reset

If clock switching is enabled, the system clock source at device Reset is chosen, as shown in Table 7-2. If clock switching is disabled, the system clock source is always selected according to the Oscillator Configuration bits. For more information, refer to the "dsPIC33/PIC24 Family Reference Manual", "Oscillator" (DS39700).

TABLE 7-2: OSCILLATOR SELECTION vs.
TYPE OF RESET (CLOCK
SWITCHING ENABLED)

	<u> </u>
Reset Type	Clock Source Determinant
POR	FNOSC<2:0> Configuration bits
BOR	(CW2<10:8>)
MCLR	
WDTO	COSC<2:0> Control bits (OSCCON<14:12>)
SWR	(000001(114.122)

TABLE 7-3: RESET DELAY TIMES FOR VARIOUS DEVICE RESETS

Reset Type	Clock Source	SYSRST Delay	System Clock Delay	Notes
POR	EC	TPOR + TSTARTUP + TRST	_	1, 2, 3
	ECPLL	TPOR + TSTARTUP + TRST	TLOCK	1, 2, 3, 5
	XT, HS, SOSC	TPOR + TSTARTUP + TRST	Tost	1, 2, 3, 4
	XTPLL, HSPLL	TPOR + TSTARTUP + TRST	Tost + Tlock	1, 2, 3, 4, 5
	FRC, FRCDIV	TPOR + TSTARTUP + TRST	TFRC	1, 2, 3, 6, 7
	FRCPLL	TPOR + TSTARTUP + TRST	TFRC + TLOCK	1, 2, 3, 5, 6
	LPRC	TPOR + TSTARTUP + TRST	TLPRC	1, 2, 3, 6
BOR	EC	TSTARTUP + TRST	_	2, 3
	ECPLL	TSTARTUP + TRST	TLOCK	2, 3, 5
	XT, HS, SOSC	TSTARTUP + TRST	Tost	2, 3, 4
	XTPLL, HSPLL	TSTARTUP + TRST	Tost + Tlock	2, 3, 4, 5
	FRC, FRCDIV	TSTARTUP + TRST	TFRC	2, 3, 6, 7
	FRCPLL	TSTARTUP + TRST	TFRC + TLOCK	2, 3, 5, 6
	LPRC	TSTARTUP + TRST	TLPRC	2, 3, 6
MCLR	Any Clock	TRST	_	3
WDT	Any Clock	Trst	_	3
Software	Any clock	Trst	_	3
Illegal Opcode	Any Clock	Trst	_	3
Uninitialized W	Any Clock	Trst	_	3
Trap Conflict	Any Clock	Trst		3

Note 1: TPOR = Power-on Reset delay (10 μs nominal).

- **2:** TSTARTUP = TVREG.
- 3: TRST = Internal State Reset time (2 μs nominal).
- **4:** Tost = Oscillator Start-up Timer (OST). A 10-bit counter counts 1024 oscillator periods before releasing the oscillator clock to the system.
- 5: TLOCK = PLL lock time.
- **6:** TFRC and TLPRC = RC Oscillator start-up times.
- 7: If Two-Speed Start-up is enabled, regardless of the Primary Oscillator selected, the device starts with FRC so the system clock delay is just TFRC, and in such cases, FRC start-up time is valid; it switches to the Primary Oscillator after its respective clock delay.

7.4.1 POR AND LONG OSCILLATOR START-UP TIMES

The oscillator start-up circuitry and its associated delay timers are not linked to the device Reset delays that occur at power-up. Some crystal circuits (especially low-frequency crystals) will have a relatively long start-up time. Therefore, one or more of the following conditions is possible after \$\overline{SYSRST}\$ is released:

- The oscillator circuit has not begun to oscillate.
- The Oscillator Start-up Timer has not expired (if a crystal oscillator is used).
- The PLL has not achieved a lock (if PLL is used).

The device will not begin to execute code until a valid clock source has been released to the system. Therefore, the oscillator and PLL start-up delays must be considered when the Reset delay time must be known.

7.4.2 FAIL-SAFE CLOCK MONITOR (FSCM) AND DEVICE RESETS

If the FSCM is enabled, it will begin to monitor the system clock source when SYSRST is released. If a valid clock source is not available at this time, the device will automatically switch to the FRC Oscillator and the user can switch to the desired crystal oscillator in the Trap Service Routine (TSR).

8.0 INTERRUPT CONTROLLER

Note:

This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the "dsPIC33/PIC24 Family Reference Manual", "Interrupts" (DS70000600). The information in this data sheet supersedes the information in the FRM.

The PIC24F interrupt controller reduces the numerous peripheral interrupt request signals to a single interrupt request signal to the PIC24F CPU. It has the following features:

- · Up to 8 processor exceptions and software traps
- · Seven user-selectable priority levels
- · Interrupt Vector Table (IVT) with up to 118 vectors
- Unique vector for each interrupt or exception source
- Fixed priority within a specified user priority level
- Alternate Interrupt Vector Table (AIVT) for debug support
- · Fixed interrupt entry and return latencies

8.1 Interrupt Vector Table

The Interrupt Vector Table (IVT) is shown in Figure 8-1. The IVT resides in program memory, starting at location, 000004h. The IVT contains 126 vectors, consisting of 8 non-maskable trap vectors, plus up to 118 sources of interrupt. In general, each interrupt source has its own vector. Each interrupt vector contains a 24-bit wide address. The value programmed into each interrupt vector location is the starting address of the associated Interrupt Service Routine (ISR).

Interrupt vectors are prioritized in terms of their natural priority; this is linked to their position in the vector table. All other things being equal, lower addresses have a higher natural priority. For example, the interrupt associated with Vector 0 will take priority over interrupts at any other vector address.

PIC24FJ128GB204 family devices implement non-maskable traps and unique interrupts. These are summarized in Table 8-1 and Table 8-2.

8.1.1 ALTERNATE INTERRUPT VECTOR TABLE

The Alternate Interrupt Vector Table (AIVT) is located after the IVT, as shown in Figure 8-1. The ALTIVT (INTCON2<15>) control bit provides access to the AIVT. If the ALTIVT bit is set, all interrupt and exception processes will use the alternate vectors instead of the default vectors. The alternate vectors are organized in the same manner as the default vectors.

The AIVT supports emulation and debugging efforts by providing a means to switch between an application and a support environment without requiring the interrupt vectors to be reprogrammed. This feature also enables switching between applications for evaluation of different software algorithms at run time. If the AIVT is not needed, the AIVT should be programmed with the same addresses used in the IVT.

8.2 Reset Sequence

A device Reset is not a true exception because the interrupt controller is not involved in the Reset process. The PIC24F devices clear their registers in response to a Reset, which forces the PC to zero. The microcontroller then begins program execution at location, 000000h. The user programs a GOTO instruction at the Reset address, which redirects program execution to the appropriate start-up routine.

Note:

Any unimplemented or unused vector locations in the IVT and AIVT should be programmed with the address of a default interrupt handler routine that contains a RESET instruction.

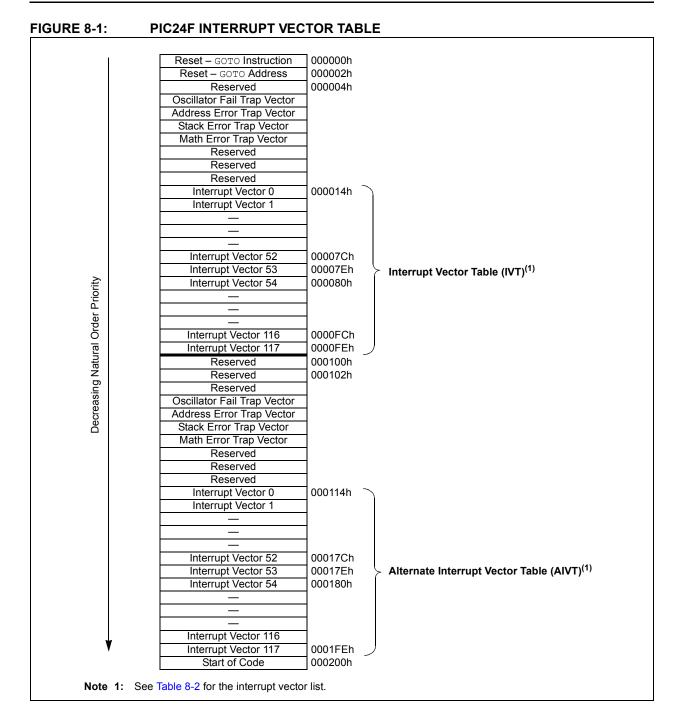


TABLE 8-1: TRAP VECTOR DETAILS

Vector Number	IVT Address	AIVT Address	Trap Source
0	000004h	000104h	Reserved
1	000006h	000106h	Oscillator Failure
2	000008h	000108h	Address Error
3	00000Ah	00010Ah	Stack Error
4	00000Ch	00010Ch	Math Error
5	00000Eh	00010Eh	Reserved
6	000010h	000110h	Reserved
7	000012h	000112h	Reserved

TABLE 8-2: IMPLEMENTED INTERRUPT VECTORS

Intownst Course	Vector	IRQ	IVT	AIVT	Interrupt Bit Locations			
Interrupt Source	#	#	Address	Address	Flag	Enable	Priority	
ADC1 Interrupt	21	13	00002Eh	00012Eh	IFS0<13>	IEC0<13>	IPC3<6:4>	
Comparator Event	26	18	000038h	000138h	IFS1<2>	IEC1<2>	IPC4<10:8>	
CRC Generator	75	67	00009Ah	00019Ah	IFS4<3>	IEC4<3>	IPC16<14:12>	
CTMU Event	85	77	0000AEh	0001AEh	IFS4<13>	IEC4<13>	IPC19<6:4>	
Cryptographic Operation Done	63	55	000082h	000182h	IFS3<7>	IEC3<7>	IPC13<14:12>	
Cryptographic Key Store Program Done	64	56	000084h	000184h	IFS3<8>	IEC3<8>	IPC14<2:0>	
Cryptographic Buffer Ready	42	34	000058h	000158h	IFS2<2>	IEC2<2>	IPC8<10:8>	
Cryptographic Rollover	43	35	00005Ah	00015Ah	IFS2<3>	IEC2<3>	IPC8<14:12>	
DMA Channel 0	12	4	00001Ch	00011Ch	IFS0<4>	IEC0<4>	IPC1<2:0>	
DMA Channel 1	22	14	000030h	000130h	IFS0<14>	IEC0<14>	IPC3<10:8>	
DMA Channel 2	32	24	000044h	000144h	IFS1<8>	IEC1<8>	IPC6<2:0>	
DMA Channel 3	44	36	00005Ch	00015Ch	IFS2<4>	IEC2<4>	IPC9<2:0>	
DMA Channel 4	54	46	000070h	000170h	IFS2<14>	IEC2<14>	IPC11<10:8>	
DMA Channel 5	69	61	00008Eh	00018Eh	IFS3<13>	IEC3<13>	IPC15<6:4>	
External Interrupt 0	8	0	000014h	000114h	IFS0<0>	IEC0<0>	IPC0<2:0>	
External Interrupt 1	28	20	00003Ch	00013Ch	IFS1<4>	IEC1<4>	IPC5<2:0>	
External Interrupt 2	37	29	00004Eh	00014Eh	IFS1<13>	IEC1<13>	IPC7<6:4>	
External Interrupt 3	61	53	00007Eh	00017Eh	IFS3<5>	IEC3<5>	IPC13<6:4>	
External Interrupt 4	62	54	000080h	000180h	IFS3<6>	IEC3<6>	IPC13<10:8>	
FRC Self-Tune	114	106	0000E8h	0001E8h	IFS6<10>	IEC6<10>	IPC26<10:8>	
I2C1 Master Event	25	17	000036h	000136h	IFS1<1>	IEC1<1>	IPC4<6:4>	
I2C1 Slave Event	24	16	000034h	000134h	IFS1<0>	IEC1<0>	IPC4<2:0>	
I2C1 Bus Collision	92	84	0000BC	0001BC	IFS5<4>	IEC5<4>	IPC21<2:0>	
I2C2 Master Event	58	50	000078h	000178h	IFS3<2>	IEC3<2>	IPC12<10:8>	
I2C2 Slave Event	57	49	000076h	000176h	IFS3<1>	IEC3<1>	IPC12<6:4>	
I2C2 Bus Collision.	93	85	0000BE	0001BE	IFS5<5>	IEC5<5>	IPC21<6:4>	
Input Capture 1	9	1	000016h	000116h	IFS0<1>	IEC0<1>	IPC0<6:4>	
Input Capture 2	13	5	00001Eh	00011Eh	IFS0<5>	IEC0<5>	IPC1<6:4>	
Input Capture 3	45	37	00005Eh	00015Eh	IFS2<5>	IEC2<5>	IPC9<6:4>	
Input Capture 4	46	38	000060h	000160h	IFS2<6>	IEC2<6>	IPC9<10:8>	
Input Capture 5	47	39	000062h	000162h	IFS2<7>	IEC2<7>	IPC9<14:12>	
Input Capture 6	48	40	000064h	000164h	IFS2<8>	IEC2<8>	IPC10<2:0>	
JTAG	125	117	0000FEh	0001FEh	IFS7<5>	IEC7<5>	IPC29<6:4>	
Input Change Notification (ICN)	27	19	00003Ah	00013Ah	IFS1<3>	IEC1<3>	IPC4<14:12>	
High/Low-Voltage Detect (HLVD)	80	72	0000A4h	0001A4h	IFS4<8>	IEC4<8>	IPC18<2:0>	
Output Compare 1	10	2	000018h	000118h	IFS0<2>	IEC0<2>	IPC0<10:8>	
Output Compare 2	14	6	000020h	000120h	IFS0<6>	IEC0<6>	IPC1<10:8>	
Output Compare 3	33	25	000046h	000146h	IFS1<9>	IEC1<9>	IPC6<6:4>	
Output Compare 4	34	26	000048h	000148h	IFS1<10>	IEC1<10>	IPC6<10:8>	
Output Compare 5	49	41	000066h	000166h	IFS2<9>	IEC2<9>	IPC10<6:4>	
Output Compare 6	50	42	000068h	000168h	IFS2<10>	IEC2<10>	IPC10<10:8>	
Enhanced Parallel Master Port (EPMP)	53	45	00006Eh	00016Eh	IFS2<13>	IEC2<13>	IPC11<6:4>	
Real-Time Clock and Calendar (RTCC)	70	62	000090h	000190h	IFS3<14>	IEC3<14>	IPC15<10:8>	

TABLE 8-2: IMPLEMENTED INTERRUPT VECTORS (CONTINUED)

Indoment Course	Vector	· IRQ	IVT	AIVT	Interrupt Bit Locations			
Interrupt Source	#	#	Address	Address	Flag	Enable	Priority	
SPI1 General	17	9	000026h	000126h	IFS0<9>	IEC0<9>	IPC2<6:4>	
SPI1 Transmit	18	10	000028h	000128h	IFS0<10>	IEC0<10>	IPC2<10:8>	
SPI1 Receive	66	58	000088h	000188h	IFS3<10>	IEC3<10>	IPC14<10:8>	
SPI2 General	40	32	000054h	000154h	IFS2<0>	IEC2<0>	IPC8<2:0>	
SPI2 Transmit	41	33	000056h	000156h	IFS2<1>	IEC2<1>	IPC8<6:4>	
SPI2 Receive	67	59	00008Ah	00018Ah	IFS3<11>	IEC3<11>	IPC14<14:12>	
SPI3 General	98	90	0000C8h	0001C8h	IFS5<10>	IEC5<10>	IPC22<10:8>	
SPI3 Transmit	99	91	0000CAh	0001CAh	IFS5<11>	IEC5<11>	IPC22<14:12>	
SPI3 Receive	68	60	000054h	000154h	IFS3<12>	IEC3<12>	IPC15<2:0>	
Timer1	11	3	00001Ah	00011Ah	IFS0<3>	IEC0<3>	IPC0<14:12>	
Timer2	15	7	000022h	000122h	IFS0<7>	IEC0<7>	IPC1<14:12>	
Timer3	16	8	000024h	000124h	IFS0<8>	IEC0<8>	IPC2<2:0>	
Timer4	35	27	00004Ah	00014Ah	IFS1<11>	IEC1<11>	IPC6<14:12>	
Timer5	36	28	00004Ch	00014Ch	IFS1<12>	IEC1<12>	IPC7<2:0>	
UART1 Error	73	65	000096h	000196h	IFS4<1>	IEC4<1>	IPC16<6:4>	
UART1 Receiver	19	11	00002Ah	00012Ah	IFS0<11>	IEC0<11>	IPC2<14:12>	
UART1 Transmitter	20	12	00002Ch	00012Ch	IFS0<12>	IEC0<12>	IPC3<2:0>	
UART2 Error	74	66	000098h	000198h	IFS4<2>	IEC4<2>	IPC16<10:8>	
UART2 Receiver	38	30	000050h	000150h	IFS1<14>	IEC1<14>	IPC7<10:8>	
UART2 Transmitter	39	31	000052h	000152h	IFS1<15>	IEC1<15>	IPC7<14:12>	
UART3 Error	89	81	0000B6h	0001B6h	IFS5<1>	IEC5<1>	IPC20<6:4>	
UART3 Receiver	90	82	0000B8h	0001B8h	IFS5<2>	IEC5<2>	IPC20<10:8>	
UART3 Transmitter	91	83	0000BAh	0001BAh	IFS5<3>	IEC5<3>	IPC20<14:12>	
UART4 Error	95	87	0000C2h	0001C2h	IFS5<7>	IEC5<7>	IPC21<14:12>	
UART4 Receiver	96	88	0000C4h	0001C4h	IFS5<8>	IEC5<8>	IPC22<2:0>	
UART4 Transmitter	97	89	0000C6h	0001C6h	IFS5<9>	IEC5<9>	IPC22<6:4>	
USB	94	86	0000C0h	0001C0h	IFS5<6>	IEC5<6>	IPC21<10:8>	

8.3 Interrupt Control and Status Registers

The PIC24FJ128GB204 family of devices implements a total of 43 registers for the interrupt controller:

- INTCON1
- INTCON2
- · IFS0 through IFS7
- IEC0 through IEC7
- IPC0 through IPC16, IPC18 through IPC22, IPC26 and IPC29
- INTTREG

Global interrupt control functions are controlled from INTCON1 and INTCON2. INTCON1 contains the Interrupt Nesting Disable (NSTDIS) bit, as well as the control and status flags for the processor trap sources. The INTCON2 register controls the external interrupt request signal behavior and the use of the Alternate Interrupt Vector Table (AIVT).

The IFSx registers maintain all of the interrupt request flags. Each source of interrupt has a status bit, which is set by the respective peripherals or an external signal and is cleared via software.

The IECx registers maintain all of the interrupt enable bits. These control bits are used to individually enable interrupts from the peripherals or external signals.

The IPCx registers are used to set the Interrupt Priority Level (IPL) for each source of interrupt. Each user interrupt source can be assigned to one of eight priority levels.

The INTTREG register contains the associated interrupt vector number and the new CPU Interrupt Priority Level, which are latched into the Vector Number (VECNUM<7:0>) and the Interrupt Level (ILR<3:0>) bit fields in the INTTREG register. The new Interrupt Priority Level is the priority of the pending interrupt.

The interrupt sources are assigned to the IFSx, IECx and IPCx registers in the order of their vector numbers, as shown in Table 8-2. For example, the INTO (External Interrupt 0) is shown as having a vector number and a natural order priority of 0. Thus, the INTOIF status bit is found in IFSO<0>, the INTOIE enable bit in IECO<0> and the INTOIP<2:0> priority bits in the first position of IPCO (IPCO<2:0>).

Although they are not specifically part of the interrupt control hardware, two of the CPU Control registers contain bits that control interrupt functionality. The ALU STATUS Register (SR) contains the IPL<2:0> bits (SR<7:5>). These indicate the current CPU Interrupt Priority Level. The user can change the current CPU priority level by writing to the IPLx bits.

The CORCON register contains the IPL3 bit, which together with the IPL<2:0> bits, indicates the current CPU priority level. IPL3 is a read-only bit so that trap events cannot be masked by the user software.

The interrupt controller has the Interrupt Controller Test register, INTTREG, which displays the status of the interrupt controller. When an interrupt request occurs, its associated vector number and the new Interrupt Priority Level are latched into INTTREG. This information can be used to determine a specific interrupt source if a generic ISR is used for multiple vectors (such as when ISR remapping is used in bootloader applications) or to check if another interrupt is pending while in an ISR.

All Interrupt registers are described in Register 8-1 through Register 8-45 in the succeeding pages.

REGISTER 8-1: SR: ALU STATUS REGISTER (IN CPU)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
_	_	_	_	_	_	_	DC ⁽¹⁾
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
IPL2 ^(2,3)	IPL1 ^(2,3)	IPL0 ^(2,3)	RA ⁽¹⁾	N ⁽¹⁾	OV ⁽¹⁾	Z ⁽¹⁾	C ⁽¹⁾
bit 7							bit 0

R = Readable bit

Legend:

W = Writable bit U = Unimplemented bit, read as '0'

'0' = Bit is cleared x = Bit is unknown -n = Value at POR '1' = Bit is set

bit 15-9 Unimplemented: Read as '0'

IPL<2:0>: CPU Interrupt Priority Level Status bits^(2,3) bit 7-5

111 = CPU Interrupt Priority Level is 7 (15); user interrupts are disabled

110 = CPU Interrupt Priority Level is 6 (14)

101 = CPU Interrupt Priority Level is 5 (13)

100 = CPU Interrupt Priority Level is 4 (12)

011 = CPU Interrupt Priority Level is 3 (11)

010 = CPU Interrupt Priority Level is 2 (10)

001 = CPU Interrupt Priority Level is 1 (9)

000 = CPU Interrupt Priority Level is 0 (8)

See Register 3-1 for the description of the remaining bits (bits 8, 4, 3, 2, 1 and 0) that are not dedicated to interrupt control functions.

- 2: The IPLx bits are concatenated with the IPL3 (CORCON<3>) bit to form the CPU Interrupt Priority Level. The value in parentheses indicates the Interrupt Priority Level if IPL3 = 1.
- 3: The IPLx bits are read-only when NSTDIS (INTCON1<15>) = 1.

REGISTER 8-2: CORCON: CPU CORE CONTROL REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

U-0	U-0	U-0	U-0	R/C-0	r-1	U-0	U-0
_	_	_	_	IPL3 ⁽¹⁾	_	_	_
bit 7							bit 0

Legend:	r = Reserved bit	C = Clearable bit	
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 15-4 Unimplemented: Read as '0'

bit 3 **IPL3:** CPU Interrupt Priority Level Status bit⁽¹⁾

1 = CPU Interrupt Priority Level is greater than 70 = CPU Interrupt Priority Level is 7 or less

bit 2 Reserved: Read as PSV bit bit 1-0 Unimplemented: Read as '0'

Note 1: The IPL3 bit is concatenated with the IPL<2:0> bits (SR<7:5>) to form the CPU Interrupt Priority Level; see Register 3-2 for bit description.

REGISTER 8-3: INTCON1: INTERRUPT CONTROL REGISTER 1

R/W-0	U-0						
NSTDIS	_	_	_	_	_	_	_
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0
_	_	_	MATHERR	ADDRERR	STKERR	OSCFAIL	_
bit 7							bit 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 15 **NSTDIS:** Interrupt Nesting Disable bit

1 = Interrupt nesting is disabled

0 = Interrupt nesting is enabled

bit 14-5 **Unimplemented:** Read as '0'

bit 4 MATHERR: Arithmetic Error Trap Status bit

1 = Overflow trap has occurred0 = Overflow trap has not occurred

bit 3 ADDRERR: Address Error Trap Status bit

1 = Address error trap has occurred

0 = Address error trap has not occurred

bit 2 STKERR: Stack Error Trap Status bit

1 = Stack error trap has occurred

0 = Stack error trap has not occurred

bit 1 OSCFAIL: Oscillator Failure Trap Status bit

1 = Oscillator failure trap has occurred

0 = Oscillator failure trap has not occurred

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

REGISTER 8-4: INTCON2: INTERRUPT CONTROL REGISTER 2

R/W-0	R-0, HSC	U-0	U-0	U-0	U-0	U-0	U-0
ALTIVT	DISI	_	_	_	_		_
bit 15	•						bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	_	INT4EP	INT3EP	INT2EP	INT1EP	INT0EP
bit 7							bit 0

Legend:	HSC = Hardware Settable/0	HSC = Hardware Settable/Clearable bit					
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 15 ALTIVT: Enable Alternate Interrupt Vector Table bit

1 = Uses Alternate Interrupt Vector Table

0 = Uses standard (default) Interrupt Vector Table

bit 14 DISI: DISI Instruction Status bit

1 = DISI instruction is active 0 = DISI instruction is not active

bit 13-5 Unimplemented: Read as '0'

bit 4 INT4EP: External Interrupt 4 Edge Detect Polarity Select bit

1 = Interrupt on negative edge0 = Interrupt on positive edge

bit 3 INT3EP: External Interrupt 3 Edge Detect Polarity Select bit

1 = Interrupt on negative edge0 = Interrupt on positive edge

bit 2 INT2EP: External Interrupt 2 Edge Detect Polarity Select bit

1 = Interrupt on negative edge0 = Interrupt on positive edge

bit 1 INT1EP: External Interrupt 1 Edge Detect Polarity Select bit

1 = Interrupt on negative edge0 = Interrupt on positive edge

bit 0 INT0EP: External Interrupt 0 Edge Detect Polarity Select bit

1 = Interrupt on negative edge0 = Interrupt on positive edge

REGISTER 8-5: IFS0: INTERRUPT FLAG STATUS REGISTER 0

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	DMA1IF	AD1IF	U1TXIF	U1RXIF	SPI1TXIF	SPI1IF	T3IF
bit 15							bit 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
T2IF	OC2IF	IC2IF	DMA0IF	T1IF	OC1IF	IC1IF	INT0IF
bit 7							bit 0

Legend:

bit 7

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

bit 14 DMA1IF: DMA Channel 1 Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 13 AD1IF: ADC1 Event Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 12 **U1TXIF:** UART1 Transmitter Interrupt Flag Status bit

1 = Interrupt request has occurred

0 = Interrupt request has not occurred

bit 11 **U1RXIF:** UART1 Receiver Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 10 SPI1TXIF: SPI1 Transmit Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 9 SPI1IF: SPI1 General Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 8 T3IF: Timer3 Interrupt Flag Status bit

1 = Interrupt request has occurred

0 = Interrupt request has not occurred

T2IF: Timer2 Interrupt Flag Status bit

1 = Interrupt request has occurred

0 = Interrupt request has not occurred

bit 6 OC2IF: Output Compare Channel 2 Interrupt Flag Status bit

1 = Interrupt request has occurred

0 = Interrupt request has not occurred

bit 5 IC2IF: Input Capture Channel 2 Interrupt Flag Status bit

1 = Interrupt request has occurred

0 = Interrupt request has not occurred

bit 4 DMA0IF: DMA Channel 0 Interrupt Flag Status bit

1 = Interrupt request has occurred

0 = Interrupt request has not occurred

bit 3 T1IF: Timer1 Interrupt Flag Status bit

1 = Interrupt request has occurred

0 = Interrupt request has not occurred

REGISTER 8-5: IFS0: INTERRUPT FLAG STATUS REGISTER 0 (CONTINUED)

bit 2 OC1IF: Output Compare Channel 1 Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 1 IC1IF: Input Capture Channel 1 Interrupt Flag Status bit

1 = Interrupt request has occurred 0 = Interrupt request has not occurred

bit 0 INT0IF: External Interrupt 0 Flag Status bit

REGISTER 8-6: IFS1: INTERRUPT FLAG STATUS REGISTER 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
U2TXIF	U2RXIF	INT2IF	T5IF	T4IF	OC4IF	OC3IF	DMA2IF
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	_	INT1IF	CNIF	CMIF	MI2C1IF	SI2C1IF
bit 7							bit 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 15 **U2TXIF:** UART2 Transmitter Interrupt Flag Status bit

1 = Interrupt request has occurred

0 = Interrupt request has not occurred

bit 14 **U2RXIF:** UART2 Receiver Interrupt Flag Status bit

1 = Interrupt request has occurred

0 = Interrupt request has not occurred

bit 13 INT2IF: External Interrupt 2 Flag Status bit

1 = Interrupt request has occurred

0 = Interrupt request has not occurred

bit 12 **T5IF:** Timer5 Interrupt Flag Status bit

1 = Interrupt request has occurred

0 = Interrupt request has not occurred

bit 11 **T4IF:** Timer4 Interrupt Flag Status bit

1 = Interrupt request has occurred

0 = Interrupt request has not occurred

bit 10 OC4IF: Output Compare Channel 4 Interrupt Flag Status bit

1 = Interrupt request has occurred

0 = Interrupt request has not occurred

bit 9 OC3IF: Output Compare Channel 3 Interrupt Flag Status bit

1 = Interrupt request has occurred

0 = Interrupt request has not occurred

bit 8 DMA2IF: DMA Channel 2 Interrupt Flag Status bit

1 = Interrupt request has occurred

0 = Interrupt request has not occurred

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

bit 4 INT1IF: External Interrupt 1 Flag Status bit

1 = Interrupt request has occurred

0 = Interrupt request has not occurred

bit 3 CNIF: Input Change Notification Interrupt Flag Status bit

1 = Interrupt request has occurred

0 = Interrupt request has not occurred

REGISTER 8-6: IFS1: INTERRUPT FLAG STATUS REGISTER 1 (CONTINUED)

bit 2 CMIF: Comparator Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 1 MI2C1IF: Master I2C1 Event Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 0 SI2C1IF: Slave I2C1 Event Interrupt Flag Status bit

REGISTER 8-7: IFS2: INTERRUPT FLAG STATUS REGISTER 2

U-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0
_	DMA4IF	PMPIF	_	_	OC6IF	OC5IF	IC6IF
bit 15							bit 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
IC5IF	IC4IF	IC3IF	DMA3IF	CRYROLLIF	CRYFREEIF	SPI2TXIF	SPI2IF
bit 7							bit 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 15 **Unimplemented:** Read as '0'

bit 14 DMA4IF: DMA Channel 4 Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 13 PMPIF: Parallel Master Port Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 12-11 **Unimplemented:** Read as '0'

bit 10 OC6IF: Output Compare Channel 6 Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 9 OC5IF: Output Compare Channel 5 Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 8 IC6IF: Input Capture Channel 6 Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 7 IC5IF: Input Capture Channel 5 Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 6 IC4IF: Input Capture Channel 4 Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 5 IC3IF: Input Capture Channel 3 Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 4 DMA3IF: DMA Channel 3 Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 3 CRYROLLIF: Cryptographic Rollover Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 2 CRYFREEIF: Cryptographic Buffer Free Status bit

REGISTER 8-7: IFS2: INTERRUPT FLAG STATUS REGISTER 2 (CONTINUED)

bit 1 SPI2TXIF: SPI2 Transmit Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 0 SPI2IF: SPI2 General Interrupt Flag Status bit

REGISTER 8-8: IFS3: INTERRUPT FLAG STATUS REGISTER 3

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0
_	RTCIF	DMA5IF	SPI3RXIF	SPI2RXIF	SPI1RXIF	_	KEYSTRIF
bit 15							bit 8

R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	U-0
CRYDNIF	INT4IF	INT3IF	_	_	MI2C2IF	SI2C2IF	_
bit 7							bit 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 15 **Unimplemented:** Read as '0'

bit 14 RTCIF: Real-Time Clock and Calendar Interrupt Flag Status bit

1 = Interrupt request has occurred

0 = Interrupt request has not occurred

bit 13 DMA5IF: DMA Channel 5 Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 12 SPI3RXIF: SPI3 Receive Interrupt Flag Status bit

1 = Interrupt request has occurred

0 = Interrupt request has not occurred

bit 11 SPI2RXIF: SPI2 Receive Interrupt Flag Status bit

1 = Interrupt request has occurred

0 = Interrupt request has not occurred

bit 10 SPI1RXIF: SPI1 Receive Interrupt Flag Status bit

1 = Interrupt request has occurred

0 = Interrupt request has not occurred

bit 9 **Unimplemented:** Read as '0'

bit 8 KEYSTRIF: Cryptographic Key Store Program Done Interrupt Flag Status bit

1 = Interrupt request has occurred

0 = Interrupt request has not occurred

bit 7 CRYDNIF: Cryptographic Operation Done Interrupt Flag Status bit

1 = Interrupt request has occurred

0 = Interrupt request has not occurred

bit 6 INT4IF: External Interrupt 4 Flag Status bit

1 = Interrupt request has occurred

0 = Interrupt request has not occurred

REGISTER 8-8: IFS3: INTERRUPT FLAG STATUS REGISTER 3 (CONTINUED)

bit 5 INT3IF: External Interrupt 3 Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 4-3 **Unimplemented:** Read as '0'

bit 2 MI2C2IF: Master I2C2 Event Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 1 SI2C2IF: Slave I2C2 Event Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 0 Unimplemented: Read as '0'

REGISTER 8-9: IFS4: INTERRUPT FLAG STATUS REGISTER 4

U-0	U-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0
_	_	CTMUIF	_	_	_	_	HLVDIF
bit 15							bit 8

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	U-0
_	_	_	_	CRCIF	U2ERIF	U1ERIF	_
bit 7							bit 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 15-14 Unimplemented: Read as '0'

bit 13 CTMUIF: CTMU Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 12-9 Unimplemented: Read as '0'

bit 8 **HLVDIF:** High/Low-Voltage Detect Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

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

bit 3 CRCIF: CRC Generator Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 2 **U2ERIF:** UART2 Error Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 1 **U1ERIF:** UART1 Error Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

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

REGISTER 8-10: IFS5: INTERRUPT FLAG STATUS REGISTER 5

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	_	_	SPI3TXIF	SPI3IF	U4TXIF	U4RXIF
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0
U4ERIF	USB1IF	I2C2BCIF	I2C1BCIF	U3TXIF	U3RXIF	U3ERIF	_
bit 7							bit 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 15-12 Unimplemented: Read as '0'

bit 11 SPI3TXIF: SPI3 Transmit Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 10 SPI3IF: SPI3 General Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 9 **U4TXIF:** UART4 Transmitter Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 8 **U4RXIF:** UART4 Receiver Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 7 **U4ERIF:** UART4 Error Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 6 USB1IF: USB1 (USB OTG) Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 5 I2C2BCIF: I2C2 Bus Collision Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 4 I2C1BCIF: I2C1 Bus Collision Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 3 U3TXIF: UART3 Transmitter Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 2 U3RXIF: UART3 Receiver Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 1 U3ERIF: UART3 Error Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

REGISTER 8-11: IFS6: INTERRUPT FLAG STATUS REGISTER 6

U-0	U-0	U-0	U-0	U-0	R/W-0	U-0	U-0
_	_	_	_	_	FSTIF	_	_
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 7							bit 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 15-11 **Unimplemented:** Read as '0'

bit 10 FSTIF: FRC Self-Tune Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

bit 9-0 **Unimplemented:** Read as '0'

REGISTER 8-12: IFS7: INTERRUPT FLAG STATUS REGISTER 7

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

U-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
_	_	JTAGIF	_	_	_	_	_
bit 7							bit 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 15-6 **Unimplemented:** Read as '0'

bit 5 JTAGIF: JTAG Controller Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurred

REGISTER 8-13: IEC0: INTERRUPT ENABLE CONTROL REGISTER 0

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	DMA1IE	AD1IE	U1TXIE	U1RXIE	SPI1TXIE	SPI1IE	T3IE
bit 15							bit 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
T2IE	OC2IE	IC2IE	DMA0IE	T1IE	OC1IE	IC1IE	INT0IE
bit 7							bit 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 15 Unimplemented: Read as '0'

bit 14 DMA1IE: DMA Channel 1 Interrupt Enable bit

1 = Interrupt request is enabled

0 = Interrupt request is not enabled

bit 13 AD1IE: ADC1 Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

bit 12 **U1TXIE:** UART1 Transmitter Interrupt Enable bit

1 = Interrupt request is enabled

0 = Interrupt request is not enabled

bit 11 **U1RXIE:** UART1 Receiver Interrupt Enable bit

1 = Interrupt request is enabled

0 = Interrupt request is not enabled

bit 10 SPI1TXIE: SPI1 Transmit Complete Interrupt Enable bit

1 = Interrupt request is enabled

0 = Interrupt request is not enabled

bit 9 SPI1IE: SPI1 General Interrupt Enable bit

1 = Interrupt request is enabled

0 = Interrupt request is not enabled

bit 8 T3IE: Timer3 Interrupt Enable bit

1 = Interrupt request is enabled

0 = Interrupt request is not enabled

bit 7 T2IE: Timer2 Interrupt Enable bit

1 = Interrupt request is enabled

0 = Interrupt request is not enabled

bit 6 OC2IE: Output Compare Channel 2 Interrupt Enable bit

1 = Interrupt request is enabled

0 = Interrupt request is not enabled

bit 5 IC2IE: Input Capture Channel 2 Interrupt Enable bit

1 = Interrupt request is enabled

0 = Interrupt request is not enabled

bit 4 **DMA0IE:** DMA Channel 0 Interrupt Enable bit

1 = Interrupt request is enabled

0 = Interrupt request is not enabled

bit 3 T1IE: Timer1 Interrupt Enable bit

1 = Interrupt request is enabled

0 = Interrupt request is not enabled

REGISTER 8-13: IEC0: INTERRUPT ENABLE CONTROL REGISTER 0 (CONTINUED)

bit 2 OC1IE: Output Compare Channel 1 Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

bit 1 IC1IE: Input Capture Channel 1 Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

bit 0 INTOIE: External Interrupt 0 Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

REGISTER 8-14: IEC1: INTERRUPT ENABLE CONTROL REGISTER 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
U2TXIE	U2RXIE	INT2IE ⁽¹⁾	T5IE	T4IE	OC4IE	OC3IE	DMA2IE
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	_	INT1IE ⁽¹⁾	CNIE	CMIE	MI2C1IE	SI2C1IE
bit 7							bit 0

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

bit 15 **U2TXIE:** UART2 Transmitter Interrupt Enable bit 1 = Interrupt request is enabled 0 = Interrupt request is not enabled bit 14 **U2RXIE:** UART2 Receiver Interrupt Enable bit 1 = Interrupt request is enabled 0 = Interrupt request is not enabled INT2IE: External Interrupt 2 Enable bit(1) bit 13 1 = Interrupt request is enabled 0 = Interrupt request is not enabled bit 12 **T5IE:** Timer5 Interrupt Enable bit 1 = Interrupt request is enabled 0 = Interrupt request is not enabled bit 11 T4IE: Timer4 Interrupt Enable bit 1 = Interrupt request is enabled 0 = Interrupt request is not enabled bit 10 OC4IE: Output Compare Channel 4 Interrupt Enable bit 1 = Interrupt request is enabled 0 = Interrupt request is not enabled bit 9 OC3IE: Output Compare Channel 3 Interrupt Enable bit 1 = Interrupt request is enabled 0 = Interrupt request is not enabled bit 8 DMA2IE: DMA Channel 2 Interrupt Enable bit 1 = Interrupt request is enabled 0 = Interrupt request is not enabled bit 7-5 Unimplemented: Read as '0' bit 4 INT1IE: External Interrupt 1 Enable bit(1) 1 = Interrupt request is enabled 0 = Interrupt request is not enabled bit 3 CNIE: Input Change Notification Interrupt Enable bit 1 = Interrupt request is enabled 0 = Interrupt request is not enabled bit 2 **CMIE:** Comparator Interrupt Enable bit 1 = Interrupt request is enabled 0 = Interrupt request is not enabled

lote 1: If an external interrupt is enabled, the interrupt input must also be configured to an available RPn or RPIn pin. For more information, see **Section 11.4 "Peripheral Pin Select (PPS)"**.

REGISTER 8-14: IEC1: INTERRUPT ENABLE CONTROL REGISTER 1 (CONTINUED)

bit 1 MI2C1IE: Master I2C1 Event Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

bit 0 SI2C1IE: Slave I2C1 Event Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

Note 1: If an external interrupt is enabled, the interrupt input must also be configured to an available RPn or RPIn pin. For more information, see **Section 11.4 "Peripheral Pin Select (PPS)"**.

REGISTER 8-15: IEC2: INTERRUPT ENABLE CONTROL REGISTER 2

U-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0
_	DMA4IE	PMPIE	_	_	OC6IE	OC5IE	IC6IE
bit 15							bit 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
IC5IE	IC4IE	IC3IE	DMA3IE	CRYROLLIE	CRYFREEIE	SPI2TXIE	SPI2IE
bit 7							bit 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 15 **Unimplemented:** Read as '0'

bit 14 DMA4IE: DMA Channel 4 Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

bit 13 PMPIE: Parallel Master Port Interrupt Enable bit

1 = Interrupt request is enabled

0 = Interrupt request is not enabled

bit 12-11 **Unimplemented:** Read as '0'

bit 10 OC6IE: Output Compare Channel 6 Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

bit 9 OC5IE: Output Compare Channel 5 Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

bit 8 IC6IE: Input Capture Channel 6 Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

bit 7 IC5IE: Input Capture Channel 5 Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

bit 6 IC4IE: Input Capture Channel 4 Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

bit 5 IC3IE: Input Capture Channel 3 Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

bit 4 DMA3IE: DMA Channel 3 Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

bit 3 CRYROLLIE: Cryptographic Rollover Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

bit 2 CRYFREEIE: Cryptographic Buffer Free Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

REGISTER 8-15: IEC2: INTERRUPT ENABLE CONTROL REGISTER 2 (CONTINUED)

bit 1 SPI2TXIE: SPI2 Transmit Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

bit 0 SPI2IE: SPI2 General Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

REGISTER 8-16: IEC3: INTERRUPT ENABLE CONTROL REGISTER 3

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0
_	RTCIE	DMA5IE	SPI3RXIE	SPI2RXIE	SPI1RXIE	_	KEYSTRIE
bit 15							bit 8

R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	U-0
CRYDNIE	INT4IE ⁽¹⁾	INT3IE ⁽¹⁾	_	_	MI2C2IE	SI2C2IE	_
bit 7							bit 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 15	Unimplemented: Read as '0'
bit 14	RTCIE: Real-Time Clock and Calendar Interrupt Enable bit
	1 = Interrupt request is enabled
	0 = Interrupt request is not enabled
bit 13	DMA5IE: DMA Channel 5 Interrupt Enable bit
	1 = Interrupt request is enabled
	0 = Interrupt request is not enabled
bit 12	SPI3RXIE: SPI3 Receive Interrupt Enable bit
	1 = Interrupt request is enabled
	0 = Interrupt request is not enabled
bit 11	SPI2RXIE: SPI2 Receive Interrupt Enable bit
	1 = Interrupt request is enabled
h:: 40	0 = Interrupt request is not enabled
bit 10	SPI1RXIE: SPI1 Receive Interrupt Enable bit
	1 = Interrupt request is enabled 0 = Interrupt request is not enabled
bit 9	Unimplemented: Read as '0'
bit 8	•
DILO	KEYSTRIE: Cryptographic Key Store Program Done Interrupt Enable bit 1 = Interrupt request is enabled
	0 = Interrupt request is enabled
bit 7	CRYDNIE: Cryptographic Operation Done Interrupt Enable bit
Sit 1	1 = Interrupt request is enabled
	0 = Interrupt request is not enabled
bit 6	INT4IE: External Interrupt 4 Enable bit ⁽¹⁾
	1 = Interrupt request is enabled
	0 = Interrupt request is not enabled
bit 5	INT3IE: External Interrupt 3 Enable bit ⁽¹⁾
	1 = Interrupt request is enabled
	0 = Interrupt request is not enabled
bit 4-3	Unimplemented: Read as '0'
bit 2	MI2C2IE: Master I2C2 Event Interrupt Enable bit
	1 = Interrupt request is enabled
	0 = Interrupt request is not enabled

Note 1: If an external interrupt is enabled, the interrupt input must also be configured to an available RPn or RPIn pin. For more information, see Section 11.4 "Peripheral Pin Select (PPS)".

REGISTER 8-16: IEC3: INTERRUPT ENABLE CONTROL REGISTER 3 (CONTINUED)

bit 1 SI2C2IE: Slave I2C2 Event Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

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

Note 1: If an external interrupt is enabled, the interrupt input must also be configured to an available RPn or RPIn pin. For more information, see Section 11.4 "Peripheral Pin Select (PPS)".

REGISTER 8-17: IEC4: INTERRUPT ENABLE CONTROL REGISTER 4

U-0	U-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0
_	_	CTMUIE	_	_	_	_	HLVDIE
bit 15							bit 8

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	U-0
_	_	_	_	CRCIE	U2ERIE	U1ERIE	_
bit 7							bit 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 15-14 **Unimplemented:** Read as '0'

bit 13 CTMUIE: CTMU Interrupt Enable bit

1 = Interrupt request is enabled

0 = Interrupt request is not enabled

bit 12-9 **Unimplemented:** Read as '0'

bit 8 **HLVDIE**: High/Low-Voltage Detect Interrupt Enable bit

1 = Interrupt request is enabled

0 = Interrupt request is not enabled

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

bit 3 CRCIE: CRC Generator Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

bit 2 **U2ERIE:** UART2 Error Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

bit 1 **U1ERIE:** UART1 Error Interrupt Enable bit

1 = Interrupt request is enabled

0 = Interrupt request is not enabled

REGISTER 8-18: IEC5: INTERRUPT ENABLE CONTROL REGISTER 5

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	_	_	SPI3TXIE	SPI3IE	U4TXIE	U4RXIE
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0
U4ERIE	USB1IE	I2C2BCIE	I2C1BCIE	U3TXIE	U3RXIE	U3ERIE	_
bit 7							bit 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 15-12 **Unimplemented:** Read as '0'

bit 11 SPI3TXIE: SPI3 Transmit Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

bit 10 SPI3IE: SPI3 General Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

bit 9 **U4TXIE:** UART4 Transmitter Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

bit 8 **U4RXIE:** UART4 Receiver Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

bit 7 **U4ERIE:** UART4 Error Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

bit 6 USB1IE: USB1 (USB OTG) Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

bit 5 I2C2BCIE: I2C2 Bus Collision Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

bit 4 I2C1BCIE: I2C1 Bus Collision Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

bit 3 U3TXIE: UART3 Transmitter Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

bit 2 U3RXIE: UART3 Receiver Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

bit 1 **U3ERIE:** UART3 Error Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

REGISTER 8-19: IEC6: INTERRUPT ENABLE CONTROL REGISTER 6

U-0	U-0	U-0	U-0	U-0	R/W-0	U-0	U-0
_	_	_	_	_	FSTIE	_	_
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 7							bit 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 15-11 **Unimplemented:** Read as '0'

bit 10 FSTIE: FRC Self-Tune Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

bit 9-0 **Unimplemented:** Read as '0'

REGISTER 8-20: IEC7: INTERRUPT ENABLE CONTROL REGISTER 7

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

U-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
_	_	JTAGIE	_	_	_	_	_
bit 7							bit 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 15-6 **Unimplemented:** Read as '0'

bit 5 **JTAGIE:** JTAG Interrupt Enable bit

1 = Interrupt request is enabled0 = Interrupt request is not enabled

REGISTER 8-21: IPC0: INTERRUPT PRIORITY CONTROL REGISTER 0

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_	T1IP2	T1IP1	T1IP0	_	OC1IP2	OC1IP1	OC1IP0
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_	IC1IP2	IC1IP1	IC1IP0	_	INT0IP2	INT0IP1	INT0IP0
bit 7							bit 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 15 **Unimplemented:** Read as '0'

bit 14-12 T1IP<2:0>: Timer1 Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

.

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 10-8 OC1IP<2:0>: Output Compare Channel 1 Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

.

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 6-4 IC1IP<2:0>: Input Capture Channel 1 Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 2-0 **INT0IP<2:0>:** External Interrupt 0 Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

•

001 = Interrupt is Priority 1

REGISTER 8-22: IPC1: INTERRUPT PRIORITY CONTROL REGISTER 1

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_	T2IP2	T2IP1	T2IP0	_	OC2IP2	OC2IP1	OC2IP0
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_	IC2IP2	IC2IP1	IC2IP0	_	DMA0IP2	DMA0IP1	DMA0IP0
bit 7							bit 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 15 **Unimplemented:** Read as '0'

bit 14-12 T2IP<2:0>: Timer2 Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

.

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 10-8 OC2IP<2:0>: Output Compare Channel 2 Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

-

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

bit 7 Unimplemented: Read as '0'

bit 6-4 IC2IP<2:0>: Input Capture Channel 2 Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

.

.

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 2-0 DMA0IP<2:0>: DMA Channel 0 Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

.

.

001 = Interrupt is Priority 1

REGISTER 8-23: IPC2: INTERRUPT PRIORITY CONTROL REGISTER 2

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_	U1RXIP2	U1RXIP1	U1RXIP0	_	SPI1TXIP2	SPI1TXIP1	SPI1TXIP0
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_	SPI1IP2	SPI1IP1	SPI1IP0	_	T3IP2	T3IP1	T3IP0
bit 7							bit 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 15 **Unimplemented:** Read as '0'

bit 14-12 U1RXIP<2:0>: UART1 Receiver Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

.

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 10-8 SPI1TXIP<2:0>: SPI1 Transmit Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

bit 7 Unimplemented: Read as '0'

bit 6-4 SPI1IP<2:0>: SPI1 General Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

.

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 2-0 T3IP<2:0>: Timer3 Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

•

001 = Interrupt is Priority 1

REGISTER 8-24: IPC3: INTERRUPT PRIORITY CONTROL REGISTER 3

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
_	_		-		DMA1IP2	DMA1IP1	DMA1IP0
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_	AD1IP2	AD1IP1	AD1IP0	_	U1TXIP2	U1TXIP1	U1TXIP0
bit 7							bit 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 15-11 **Unimplemented:** Read as '0'

bit 10-8 **DMA1IP<2:0>:** DMA Channel 1 Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

bit 7 Unimplemented: Read as '0'

bit 6-4 AD1IP<2:0>: ADC1 Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

:

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

bit 3 Unimplemented: Read as '0'

bit 2-0 **U1TXIP<2:0>:** UART1 Transmitter Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

.

001 = Interrupt is Priority 1

REGISTER 8-25: IPC4: INTERRUPT PRIORITY CONTROL REGISTER 4

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_	CNIP2	CNIP1	CNIP0	_	CMIP2	CMIP1	CMIP0
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_	MI2C1IP2	MI2C1IP1	MI2C1IP0	_	SI2C1IP2	SI2C1IP1	SI2C1IP0
bit 7							bit 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 15 **Unimplemented:** Read as '0'

bit 14-12 CNIP<2:0>: Input Change Notification Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

.

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 10-8 **CMIP<2:0>:** Comparator Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

000 - Interrupt source is disabled

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

bit 6-4 MI2C1IP<2:0>: Master I2C1 Event Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

.

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 2-0 SI2C1IP<2:0>: Slave I2C1 Event Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

•

001 = Interrupt is Priority 1

REGISTER 8-26: IPC5: INTERRUPT PRIORITY CONTROL REGISTER 5

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
_	_	_	_	_		INT1IP<2:0>	
bit 7							bit 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 15-3 Unimplemented: Read as '0'

bit 2-0 INT1IP<2:0>: External Interrupt 1 Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

.

•

001 = Interrupt is Priority 1

REGISTER 8-27: IPC6: INTERRUPT PRIORITY CONTROL REGISTER 6

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_	T4IP2	T4IP1	T4IP0	_	OC4IP2	OC4IP1	OC4IP0
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_	OC3IP2	OC3IP1	OC3IP0	_	DMA2IP2	DMA2IP1	DMA2IP0
bit 7							bit 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 15 **Unimplemented:** Read as '0'

bit 14-12 **T4IP<2:0>:** Timer4 Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

.

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 10-8 OC4IP<2:0>: Output Compare Channel 4 Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

.

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

bit 7 Unimplemented: Read as '0'

bit 6-4 OC3IP<2:0>: Output Compare Channel 3 Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 2-0 DMA2IP<2:0>: DMA Channel 2 Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

.

•

001 = Interrupt is Priority 1

REGISTER 8-28: IPC7: INTERRUPT PRIORITY CONTROL REGISTER 7

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_	U2TXIP2	U2TXIP1	U2TXIP0		U2RXIP2	U2RXIP1	U2RXIP0
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_	INT2IP2	INT2IP1	INT2IP0	_	T5IP2	T5IP1	T5IP0
bit 7							bit 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 15 **Unimplemented:** Read as '0'

bit 14-12 U2TXIP<2:0>: UART2 Transmitter Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

.

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 10-8 **U2RXIP<2:0>:** UART2 Receiver Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

bit 7 Unimplemented: Read as '0'

bit 6-4 **INT2IP<2:0>:** External Interrupt 2 Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 2-0 **T5IP<2:0>:** Timer5 Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

.

001 = Interrupt is Priority 1

REGISTER 8-29: IPC8: INTERRUPT PRIORITY CONTROL REGISTER 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_	CRYROLLIP2	CRYROLLIP1	CRYROLLIP0	_	CRYFREEIP2	CRYFREEIP1	CRYFREEIP0
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_	SPI2TXIP2	SPI2TXIP1	SPI2TXIP0	_	SPI2IP2	SPI2IP1	SPI2IP0
bit 7							bit 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 15 **Unimplemented:** Read as '0'

bit 14-12 CRYROLLIP<2:0>: Cryptographic Rollover Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

.

.

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

bit 11 Unimplemented: Read as '0'

bit 10-8 CRYFREEIP<2:0>: Cryptographic Buffer Free Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

bit 7 Unimplemented: Read as '0'

bit 6-4 SPI2TXIP<2:0>: SPI2 Transmit Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 2-0 SPI2IP<2:0>: SPI2 General Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

٠

_

001 = Interrupt is Priority 1

REGISTER 8-30: IPC9: INTERRUPT PRIORITY CONTROL REGISTER 9

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_	IC5IP2	IC5IP1	IC5IP0		IC4IP2	IC4IP1	IC4IP0
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_	IC3IP2	IC3IP1	IC3IP0	_	DMA3IP2	DMA3IP1	DMA3IP0
bit 7							bit 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 15 **Unimplemented:** Read as '0'

bit 14-12 IC5IP<2:0>: Input Capture Channel 5 Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

:

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 10-8 IC4IP<2:0>: Input Capture Channel 4 Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

bit 7 Unimplemented: Read as '0'

bit 6-4 IC3IP<2:0>: Input Capture Channel 3 Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 2-0 DMA3IP<2:0>: DMA Channel 3 Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

.

.

001 = Interrupt is Priority 1

REGISTER 8-31: IPC10: INTERRUPT PRIORITY CONTROL REGISTER 10

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
_	_		_	_	OC6IP2	OC6IP1	OC6IP0
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_	OC5IP2	OC5IP1	OC5IP0	_	IC6IP2	IC6IP1	IC6IP0
bit 7							bit 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 15-11 **Unimplemented:** Read as '0'

bit 10-8 OC6IP<2:0>: Output Compare Channel 6 Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

.

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 6-4 OC5IP<2:0>: Output Compare Channel 5 Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 2-0 IC6IP<2:0>: Input Capture Channel 6 Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

001 = Interrupt is Priority 1

REGISTER 8-32: IPC11: INTERRUPT PRIORITY CONTROL REGISTER 11

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
_	_	_			DMA4IP2	DMA4IP1	DMA4IP0
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
_	PMPIP2	PMPIP1	PMPIP0	_	_	_	_
bit 7							bit 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 15-11 **Unimplemented:** Read as '0'

bit 10-8 DMA4IP<2:0>: DMA Channel 4 Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

bit 7 Unimplemented: Read as '0'

bit 6-4 **PMPIP<2:0>:** Parallel Master Port Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

.

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

REGISTER 8-33: IPC12: INTERRUPT PRIORITY CONTROL REGISTER 12

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
_	_	_	_	_	MI2C2IP2	MI2C2IP1	MI2C2IP0
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
_	SI2C2IP2	SI2C2IP1	SI2C2IP0	_	_	_	_
bit 7							bit 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 15-11 Unimplemented: Read as '0'

bit 10-8 MI2C2IP<2:0>: Master I2C2 Event Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

.

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

bit 7 Unimplemented: Read as '0'

bit 6-4 SI2C2IP<2:0>: Slave I2C2 Event Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

_

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

REGISTER 8-34: IPC13: INTERRUPT PRIORITY CONTROL REGISTER 13

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_	CRYDNIP2	CRYDNIP1	CRYDNIP0		INT4IP2	INT4IP1	INT4IP0
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0	
_	INT3IP2	INT3IP1	INT3IP0	_	_	_	_	
bit 7 bit								

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

bit 14-12 CRYDNIP<2:0>: Cryptographic Operation Done Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

:

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 10-8 **INT4IP<2:0>:** External Interrupt 4 Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

bit 7 Unimplemented: Read as '0'

bit 6-4 **INT3IP<2:0>:** External Interrupt 3 Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

-

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

REGISTER 8-35: IPC14: INTERRUPT PRIORITY CONTROL REGISTER 14

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_	SPI2RXIP2	SPI2RXIP1	SPI2RXIPO	_	SPI1RXIP2	SPI1RXIP1	SPI1RXIPO
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
_	_	_	_	_	KEYSTRIP2	KEYSTRIP1	KEYSTRIP0
bit 7							bit 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 15 **Unimplemented:** Read as '0'

bit 14-12 SPI2RXIP<2:0>: SPI2 Receive Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 10-8 SPI1RXIP<2:0>: SPI1 Receive Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 2-0 **KEYSTRIP<2:0>:** Cryptographic Key Store Program Done Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

.

001 = Interrupt is Priority 1

REGISTER 8-36: IPC15: INTERRUPT PRIORITY CONTROL REGISTER 15

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
_	_	_	_	_	RTCIP2	RTCIP1	RTCIP0
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_	DMA5IP2	DMA5IP1	DMA5IP0	_	SPI3RXIP2	SPI3RXIP1	SPI3RXIP0
bit 7							bit 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 15-11 **Unimplemented:** Read as '0'

bit 10-8 RTCIP<2:0>: Real-Time Clock and Calendar Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

bit 7 Unimplemented: Read as '0'

bit 6-4 **DMA5IP<2:0>:** DMA Channel 5 Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 2-0 SPI3RXIP<2:0>: SPI3 Receive Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

-

•

001 = Interrupt is Priority 1

REGISTER 8-37: IPC16: INTERRUPT PRIORITY CONTROL REGISTER 16

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_	CRCIP2	CRCIP1	CRCIP0	_	U2ERIP2	U2ERIP1	U2ERIP0
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
_	U1ERIP2	U1ERIP1	U1ERIP0	_	_	_	_
bit 7							bit 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 15 **Unimplemented:** Read as '0'

bit 14-12 CRCIP<2:0>: CRC Generator Error Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

.

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 10-8 **U2ERIP<2:0>:** UART2 Error Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

bit 7 Unimplemented: Read as '0'

bit 6-4 **U1ERIP<2:0>:** UART1 Error Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

REGISTER 8-38: IPC18: INTERRUPT PRIORITY CONTROL REGISTER 18

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_			_
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
_	_	_	_	_		HLVDIP<2:0>	
bit 7							bit 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 15-3 **Unimplemented:** Read as '0'

bit 2-0 **HLVDIP<2:0>:** High/Low-Voltage Detect Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

REGISTER 8-39: IPC19: INTERRUPT PRIORITY CONTROL REGISTER 19

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
_		CTMUIP<2:0>		_	_	_	_
bit 7							bit 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 15-7 **Unimplemented:** Read as '0'

bit 6-4 CTMUIP<2:0>: CTMU Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

REGISTER 8-40: IPC20: INTERRUPT PRIORITY CONTROL REGISTER 20

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_	U3TXIP2	U3TXIP1	U3TXIP0	_	U3RXIP2	U3RXIP1	U3RXIP0
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
_	U3ERIP2	U3ERIP1	U3ERIP0	_	_	_	_
bit 7							bit 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 15 **Unimplemented:** Read as '0'

bit 14-12 U3TXIP<2:0>: UART3 Transmitter Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 10-8 U3RXIP<2:0>: UART3 Receiver Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

bit 7 Unimplemented: Read as '0'

bit 6-4 **U3ERIP<2:0>:** UART3 Error Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

REGISTER 8-41: IPC21: INTERRUPT PRIORITY CONTROL REGISTER 21

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_	U4ERIP2	U4ERIP1	U4ERIP0	_	USB1IP2	USB1IP1	USB1IP0
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_	I2C2BCIP2	I2C2BCIP1	I2C2BCIP0	_	I2C1BCIP2	I2C1BCIP1	I2C1BCIP0
bit 7							bit 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 15 **Unimplemented:** Read as '0'

bit 14-12 **U4ERIP<2:0>:** UART4 Error Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

:

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 10-8 USB1IP<2:0>: USB1 (USB OTG) Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 6-4 I2C2BCIP<2:0>: I2C2 Bus Collision Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

.

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 2-0 I2C1BCIP<2:0>: I2C1 Bus Collision Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

.

.

001 = Interrupt is Priority 1

REGISTER 8-42: IPC22: INTERRUPT PRIORITY CONTROL REGISTER 22

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_	SPI3TXIP2	SPI3TXIP1	SPI3TXIP0	_	SPI3IP2	SPI3IP1	SPI3IP0
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_	U4TXIP2	U4TXIP1	U4TXIP0	_	U4RXIP2	U4RXIP1	U4RXIP0
bit 7							bit 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 15 **Unimplemented:** Read as '0'

bit 14-12 SPI3TXIP<2:0>: SPI3 Transmit Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 10-8 SPI3IP<2:0>: SPI3 General Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

bit 6-4 **U4TXIP<2:0>:** UART4 Transmitter Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

•

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

bit 3 Unimplemented: Read as '0'

bit 2-0 U4RXIP<2:0>: UART4 Receiver Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

.

•

001 = Interrupt is Priority 1

REGISTER 8-43: IPC26: INTERRUPT PRIORITY CONTROL REGISTER 26

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
-	_	_	_	-		FSTIP<2:0>	
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 7							bit 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 15-11 **Unimplemented:** Read as '0'

bit 10-8 FSTIP<2:0>: FRC Self-Tune Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

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

REGISTER 8-44: IPC29: INTERRUPT PRIORITY CONTROL REGISTER 29

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
_		JTAGIP<2:0>		_	_	_	_
bit 7							bit 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 15-7 Unimplemented: Read as '0'

bit 6-4 **JTAGIP<2:0>:** JTAG Interrupt Priority bits

111 = Interrupt is Priority 7 (highest priority interrupt)

•

001 = Interrupt is Priority 1

000 = Interrupt source is disabled

REGISTER 8-45: INTTREG: INTERRUPT CONTROLLER TEST REGISTER

R-0	r-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
CPUIRQ	_	VHOLD	_	ILR3	ILR2	ILR1	ILR0
bit 15							bit 8

| R/W-0 |
|---------|---------|---------|---------|---------|---------|---------|---------|
| VECNUM7 | VECNUM6 | VECNUM5 | VECNUM4 | VECNUM3 | VECNUM2 | VECNUM1 | VECNUM0 |
| bit 7 | | | | | | | bit 0 |

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

bit 15 **CPUIRQ:** CPU Interrupt Request from Interrupt Controller bit

1 = An interrupt request has occurred but has not yet been Acknowledged by the CPU

0 = No interrupt request is unacknowledged

bit 14 Reserved: Maintain as '0'

bit 13 VHOLD: Vector Number Capture Configuration bit

1 = VECNUM<7:0> bits contain the value of the highest priority pending interrupt

0 = VECNUM<7:0> bits contain the value of the last Acknowledged interrupt (i.e., the last interrupt that has occurred with higher priority than the CPU, even if other interrupts are pending)

bit 12 **Unimplemented:** Read as '0'

bit 11-8 ILR<3:0>: New CPU Interrupt Priority Level bits

1111 = CPU Interrupt Priority Level is 15

.

•

0001 = CPU Interrupt Priority Level is 1

0000 = CPU Interrupt Priority Level is 0

bit 7-0 VECNUM<7:0>: Vector Number of Pending Interrupt or Last Acknowledged Interrupt bits

When VHOLD = 1:

Indicates the vector number (from 0 to 118) of the last interrupt to occur.

When VHOLD = 0:

Indicates the vector number (from 0 to 118) of the interrupt request currently being handled.

8.4 Interrupt Setup Procedures

8.4.1 INITIALIZATION

To configure an interrupt source:

- Set the NSTDIS (INTCON1<15>) control bit if nested interrupts are not desired.
- Select the user-assigned priority level for the interrupt source by writing the control bits in the appropriate IPCx register. The priority level will depend on the specific application and type of interrupt source. If multiple priority levels are not desired, the IPCx register control bits for all enabled interrupt sources may be programmed to the same non-zero value.

Note: At a device Reset, the IPCx registers are initialized, such that all user interrupt sources are assigned to Priority Level 4.

- Clear the interrupt flag status bit associated with the peripheral in the associated IFSx register.
- 4. Enable the interrupt source by setting the interrupt enable control bit associated with the source in the appropriate IECx register.

8.4.2 INTERRUPT SERVICE ROUTINE (ISR)

The method that is used to declare an Interrupt Service Routine (ISR) and initialize the IVT with the correct vector address will depend on the programming language (i.e., 'C' or assembler) and the language development toolsuite that is used to develop the application. In general, the user must clear the interrupt flag in the appropriate IFSx register for the source of the interrupt that the ISR handles; otherwise, the ISR will be re-entered immediately after exiting the routine. If the ISR is coded in assembly language, it must be terminated using a RETFIE instruction to unstack the saved PC value, SRL value and old CPU priority level.

8.4.3 TRAP SERVICE ROUTINE (TSR)

A Trap Service Routine (TSR) is coded like an ISR, except that the appropriate trap status flag in the INTCON1 register must be cleared to avoid re-entry into the TSR.

8.4.4 INTERRUPT DISABLE

All user interrupts can be disabled using the following procedure:

- 1. Push the current SR value onto the software stack using the PUSH instruction.
- 2. Force the CPU to Priority Level 7 by inclusive ORing the value, 0Eh, with SRL.

To enable user interrupts, the POP instruction may be used to restore the previous SR value.

Note that only user interrupts with a priority level of 7 or less can be disabled. Trap sources (Levels 8-15) cannot be disabled.

The DISI instruction provides a convenient way to disable interrupts of Priority Levels 1-6 for a fixed period of time. Level 7 interrupt sources are not disabled by the DISI instruction.

PIC24FJ128GB204 FAMILY					
OTES:					

9.0 OSCILLATOR CONFIGURATION

Note:

This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the "dsPIC33/PIC24 Family Reference Manual", "Oscillator" (DS39700).

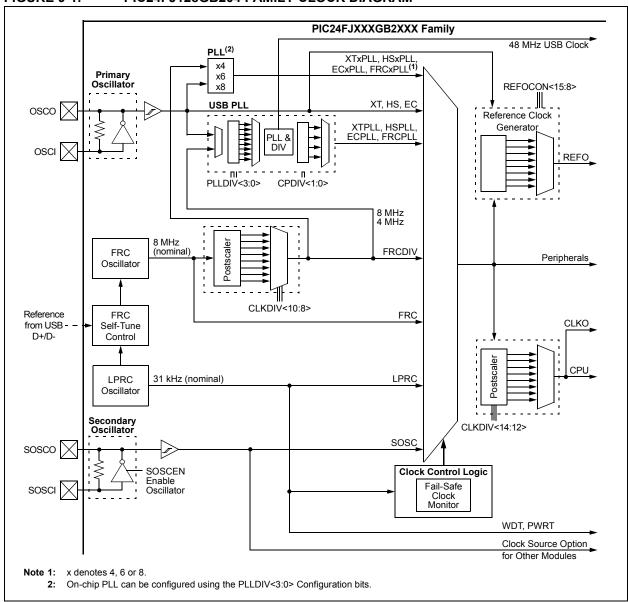
The oscillator system for PIC24FJ128GB204 family devices has the following features:

- A total of four external and internal oscillator options as clock sources, providing 15 different Clock modes
- An on-chip, USB PLL block to provide a stable 48 MHz clock for the USB module, as well as a range of frequency options for the system clock

- An on-chip PLL (x4, x6, x8) block available for the Primary Oscillator (POSC) source or FRCDIV (see Section 9.8 "On-Chip PLL")
- Software-controllable switching between various clock sources
- Software-controllable postscaler for selective clocking of CPU for system power savings
- A Fail-Safe Clock Monitor (FSCM) that detects clock failure and permits safe application recovery or shutdown
- A separate and independently configurable system clock output for synchronizing external hardware

A simplified diagram of the oscillator system is shown in Figure 9-1.

FIGURE 9-1: PIC24FJ128GB204 FAMILY CLOCK DIAGRAM



9.1 CPU Clocking Scheme

The system clock source can be provided by one of four sources:

- Primary Oscillator (POSC) on the OSCI and OSCO pins
- Secondary Oscillator (SOSC) on the SOSCI and SOSCO pins
- · Fast Internal RC (FRC) Oscillator
- · Low-Power Internal RC (LPRC) Oscillator

The Primary Oscillator and FRC sources have the option of using the internal USB PLL block, which generates both the USB module clock and a separate system clock from the 96 MHz PLL. Refer to Section 9.6 "Oscillator Modes and USB Operation" for additional information.

The internal FRC provides an 8 MHz clock source. It can optionally be reduced by the programmable clock divider to provide a range of system clock frequencies.

The selected clock source generates the processor and peripheral clock sources. The processor clock source is divided by two to produce the internal instruction cycle clock, FCY. In this document, the instruction cycle clock is also denoted by Fosc/2. The internal instruction cycle clock, Fosc/2, can be provided on the OSCO I/O pin for some operating modes of the Primary Oscillator.

9.2 Initial Configuration on POR

The oscillator source (and operating mode) that is used at a device Power-on Reset event is selected using Configuration bit settings. The Oscillator Configuration bit settings are located in the Configuration registers in program memory (for more information, refer to Section 30.1 "Configuration Bits"). The Primary Oscillator Configuration bits, POSCMD<1:0> (Configuration Word 2<1:0>), and the Initial Oscillator Select Configuration bits, FNOSC<2:0> (Configuration Word 2<10:8>), select the oscillator source that is used at a Power-on Reset. The FRC Primary Oscillator with Postscaler (FRCDIV) is the default (unprogrammed) selection. The Secondary Oscillator, or one of the internal oscillators, may be chosen by programming these bit locations.

The Configuration bits allow users to choose between the various clock modes, as shown in Table 9-1.

9.2.1 CLOCK SWITCHING MODE CONFIGURATION BITS

The FCKSM<1:0> Configuration bits (Configuration Word 2<7:6>) are used to jointly configure device clock switching and the Fail-Safe Clock Monitor (FSCM). Clock switching is enabled only when FCKSM1 is programmed ('0'). The FSCM is enabled only when FCKSM<1:0> are both programmed ('00').

TABLE 9-1: CONFIGURATION BIT VALUES FOR CLOCK SELECTION

Oscillator Mode	Oscillator Source	POSCMD<1:0>	FNOSC<2:0>	Notes
Fast RC Oscillator with Postscaler (FRCDIV)	Internal	11	111	1, 2
(Reserved)	Internal	XX	110	1
Low-Power RC Oscillator (LPRC)	Internal	11	101	1
Secondary (Timer1) Oscillator (SOSC)	Secondary	11	100	1
Primary Oscillator (XT) with PLL Module (XTPLL)	Primary	01	011	
Primary Oscillator (EC) with PLL Module (ECPLL)	Primary	00	011	
Primary Oscillator (HS)	Primary	10	010	
Primary Oscillator (XT)	Primary	01	010	
Primary Oscillator (EC)	Primary	0.0	010	
Fast RC Oscillator with PLL Module (FRCPLL)	Internal	11	001	1
Fast RC Oscillator (FRC)	Internal	11	000	1

Note 1: OSCO pin function is determined by the OSCIOFCN Configuration bit.

2: This is the default oscillator mode for an unprogrammed (erased) device.

9.3 Control Registers

The operation of the oscillator is controlled by three Special Function Registers:

- OSCCON
- CLKDIV
- OSCTUN

The OSCCON register (Register 9-1) is the main control register for the oscillator. It controls clock source switching and allows the monitoring of clock sources.

The CLKDIV register (Register 9-2) controls the features associated with Doze mode, as well as the postscaler for the FRC Oscillator.

The OSCTUN register (Register 9-3) allows the user to fine-tune the FRC Oscillator over a range of approximately ±1.5%. It also controls the FRC self-tuning features, described in Section 9.5 "FRC Self-Tuning".

REGISTER 9-1: OSCCON: OSCILLATOR CONTROL REGISTER

U-0	R-0	R-0	R-0	U-0	R/W-x ⁽¹⁾	R/W-x ⁽¹⁾	R/W-x ⁽¹⁾
_	COSC2	COSC1	COSC0	_	NOSC2	NOSC1	NOSC0
bit 15							bit 8

R/SO-0	R/W-0	R-0 ⁽³⁾	U-0	R/CO-0	R/W-0	R/W-0	R/W-0
CLKLOCK	IOLOCK ⁽²⁾	LOCK	_	CF	POSCEN	SOSCEN	OSWEN
bit 7							bit 0

Legend:	CO = Clearable Only bit	SO = Settable Only bit	
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 15 Unimplemented: Read as '0'
- bit 14-12 COSC<2:0>: Current Oscillator Selection bits
 - 111 = Fast RC Oscillator with Postscaler (FRCDIV)
 - 110 = Reserved
 - 101 = Low-Power RC Oscillator (LPRC)
 - 100 = Secondary Oscillator (SOSC)
 - 011 = Primary Oscillator with PLL module (XTPLL, HSPLL, ECPLL)
 - 010 = Primary Oscillator (XT, HS, EC)
 - 001 = Fast RC Oscillator with Postscaler and PLL module (FRCPLL)
 - 000 = Fast RC Oscillator (FRC)
- bit 11 **Unimplemented:** Read as '0'
- bit 10-8 NOSC<2:0>: New Oscillator Selection bits⁽¹⁾
 - 111 = Fast RC Oscillator with Postscaler (FRCDIV)
 - 110 = Reserved
 - 101 = Low-Power RC Oscillator (LPRC)
 - 100 = Secondary Oscillator (SOSC)
 - 011 = Primary Oscillator with PLL module (XTPLL, HSPLL, ECPLL)
 - 010 = Primary Oscillator (XT, HS, EC)
 - 001 = Fast RC Oscillator with Postscaler and PLL module (FRCPLL)
 - 000 = Fast RC Oscillator (FRC)
- **Note 1:** Reset values for these bits are determined by the FNOSCx Configuration bits.
 - 2: The state of the IOLOCK bit can only be changed once an unlocking sequence has been executed. In addition, if the IOL1WAY Configuration bit is '1', once the IOLOCK bit is set, it cannot be cleared.
 - 3: This bit also resets to '0' during any valid clock switch or whenever a non-PLL Clock mode is selected.

REGISTER 9-1: OSCCON: OSCILLATOR CONTROL REGISTER (CONTINUED)

bit 7 CLKLOCK: Clock Selection Lock Enable bit

If FSCM is enabled (FCKSM1 = 1):

1 = Clock and PLL selections are locked

0 = Clock and PLL selections are not locked and may be modified by setting the OSWEN bit

If FSCM is disabled (FCKSM1 = 0):

Clock and PLL selections are never locked and may be modified by setting the OSWEN bit.

bit 6 **IOLOCK:** I/O Lock Enable bit⁽²⁾

1 = I/O lock is active

0 = I/O lock is not active

bit 5 LOCK: PLL Lock Status bit⁽³⁾

1 = PLL module is in lock or PLL module start-up timer is satisfied

0 = PLL module is out of lock, PLL start-up timer is running or PLL is disabled

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

1 = FSCM has detected a clock failure

0 = No clock failure has been detected

bit 2 POSCEN: Primary Oscillator (POSC) Sleep Enable bit

1 = Primary Oscillator continues to operate during Sleep mode

0 = Primary Oscillator is disabled during Sleep mode

bit 1 SOSCEN: 32 kHz Secondary Oscillator (SOSC) Enable bit

1 = Enables Secondary Oscillator

0 = Disables Secondary Oscillator

bit 0 OSWEN: Oscillator Switch Enable bit

1 = Initiates an oscillator switch to a clock source specified by the NOSC<2:0> bits

0 = Oscillator switch is complete

Note 1: Reset values for these bits are determined by the FNOSCx Configuration bits.

2: The state of the IOLOCK bit can only be changed once an unlocking sequence has been executed. In addition, if the IOL1WAY Configuration bit is '1', once the IOLOCK bit is set, it cannot be cleared.

3: This bit also resets to '0' during any valid clock switch or whenever a non-PLL Clock mode is selected.

REGISTER 9-2: CLKDIV: CLOCK DIVIDER REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-1
ROI	DOZE2	DOZE1	DOZE0	DOZEN ⁽¹⁾	RCDIV2	RCDIV1	RCDIV0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0
CPDIV1	CPDIV0	PLLEN	_	_	_	_	_
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit	t, read as '0'
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15 ROI: Recover on Interrupt bit

1 = Interrupts clear the DOZEN bit and reset the CPU peripheral clock ratio to 1:1

0 = Interrupts have no effect on the DOZEN bit

bit 14-12 DOZE<2:0>: CPU Peripheral Clock Ratio Select bits

111 = 1:128

110 = 1:64

101 = 1:32

100 = 1:16

011 = 1:8

010 = 1:4

001 = 1:2

000 = 1:1

bit 11 **DOZEN:** Doze Enable bit⁽¹⁾

1 = DOZE<2:0> bits specify the CPU peripheral clock ratio

0 = CPU peripheral clock ratio is set to 1:1

bit 10-8 RCDIV<2:0>: FRC Postscaler Select bits

111 = 31.25 kHz (divide-by-256)

110 = 125 kHz (divide-by-64)

101 = 250 kHz (divide-by-32)

100 = 500 kHz (divide-by-16)

011 = 1 MHz (divide-by-8)

010 = 2 MHz (divide-by-4)

001 = 4 MHz (divide-by-2)

000 = 8 MHz (divide-by-1)

bit 7-6 CPDIV<1:0>: USB System Clock Select bits (postscaler selected from 32 MHz clock branch)

11 = 4 MHz (divide-bv-8)(2)

10 = 8 MHz (divide-by-4)(2)

01 = 16 MHz (divide-by-2)

00 = 32 MHz (divide-by-1)

bit 5 PLLEN: PLL Enable bit

1 = PLL is enabled

0 = PLL is disabled

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

Note 1: This bit is automatically cleared when the ROI bit is set and an interrupt occurs.

2: This setting is not allowed while the USB module is enabled.

REGISTER 9-3: OSCTUN: FRC OSCILLATOR TUNE REGISTER

W = Writable bit

R/W-0	U-0	R/W-0	R/W-0	R-0	R/W-0	R-0	R/W-0
STEN	_	STSIDL	STSRC ⁽¹⁾	STLOCK	STLPOL	STOR	STORPOL
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	TUN5	TUN4	TUN3	TUN2	TUN1	TUN0
bit 7							bit 0

U = Unimplemented bit, read as '0'

IX - IXCauai	JIE DIL	VV - VVIIIable bit	o – onimplemented bit,	read as 0
-n = Value a	at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown
bit 15	_	C Self-Tune Enable bit		
		•	bits are controlled by hardwar	
		• • • • • • • • • • • • • • • • • • • •	cation may optionally control TI	JNx bits
bit 14	Unimplem	ented: Read as '0'		
bit 13		RC Self-Tune Stop in Idle b		
		ning stops during Idle mode		
		ning continues during Idle r		
bit 12		RC Self-Tune Reference C		
			atch the USB host clock toleran	
			atch the 32.768 kHz SOSC tole	rance
bit 11		FRC Self-Tune Lock Status		
			£0.2% of the STSRC reference	
			±0.2% of the STSRC reference	accuracy
bit 10		FRC Self-Tune Lock Interru	•	
		une lock interrupt is genera		
		une lock interrupt is genera		
bit 9	STOR: FRO	C Self-Tune Out of Range	Status bit	
			eyond the range of TUN<5:0>;	.
			he tunable range; tuning is per	formed
bit 8		: FRC Self-Tune Out of Ra	• •	
			s generated when STOR is = 0	
		• .	s generated when STOR is = 1	
bit 7-6	-	ented: Read as '0'		
bit 5-0		FRC Oscillator Tuning bits		
		Maximum frequency deviat	tion	
	011110 =			
	•			
	•			
	000001 =			
		Center frequency, oscillato	r is running at factory calibrated	d frequency
	111111 =		-	
	•			
	•			
	• 100001 -			
	100001 =			

Note 1: Use of either clock recovery source has specific application requirements. For more information, see **Section 9.5 "FRC Self-Tuning"**.

100000 = Minimum frequency deviation

Legend:

R = Readable bit

9.4 Clock Switching Operation

With few limitations, applications are free to switch between any of the four clock sources (POSC, SOSC, FRC and LPRC) under software control and at any time. To limit the possible side effects that could result from this flexibility, PIC24F devices have a safeguard lock built into the switching process.

Note: The Primary Oscillator mode has three different submodes (XT, HS and EC) which are determined by the POSCMDx Configuration bits. While an application can switch to and from Primary Oscillator mode in software, it cannot switch between the different primary submodes without reprogramming the device.

9.4.1 ENABLING CLOCK SWITCHING

To enable clock switching, the FCKSM1 Configuration bit in CW2 must be programmed to '0'. (For more information, refer to **Section 30.1 "Configuration Bits"**.) If the FCKSM1 Configuration bit is unprogrammed ('1'), the clock switching function and Fail-Safe Clock Monitor function are disabled; this is the default setting.

The NOSCx control bits (OSCCON<10:8>) do not control the clock selection when clock switching is disabled. However, the COSC<2:0> bits (OSCCON<14:12>) will reflect the clock source selected by the FNOSCx Configuration bits.

The OSWEN control bit (OSCCON<0>) has no effect when clock switching is disabled; it is held at '0' at all times.

9.4.2 OSCILLATOR SWITCHING SEQUENCE

At a minimum, performing a clock switch requires this basic sequence:

- If desired, read the COSC<2:0> bits (OSCCON<14:12>) to determine the current oscillator source.
- 2. Perform the unlock sequence to allow a write to the OSCCON register high byte.
- Write the appropriate value to the NOSCx bits (OSCCON<10:8>) for the new oscillator source.
- Perform the unlock sequence to allow a write to the OSCCON register low byte.
- Set the OSWEN bit to initiate the oscillator switch.

Once the basic sequence is completed, the system clock hardware responds automatically as follows:

- The clock switching hardware compares the COSCx bits with the new value of the NOSCx bits. If they are the same, then the clock switch is a redundant operation. In this case, the OSWEN bit is cleared automatically and the clock switch is aborted.
- If a valid clock switch has been initiated, the LOCK (OSCCON<5>) and CF (OSCCON<3>) bits are cleared.
- The new oscillator is turned on by the hardware if it is not currently running. If a crystal oscillator must be turned on, the hardware will wait until the OST expires. If the new source is using the PLL, then the hardware waits until a PLL lock is detected (LOCK = 1).
- The hardware waits for 10 clock cycles from the new clock source and then performs the clock switch.
- The hardware clears the OSWEN bit to indicate a successful clock transition. In addition, the NOSCx bits value is transferred to the COSCx bits.
- 6. The old clock source is turned off at this time, with the exception of LPRC (if WDT or FSCM is enabled) or SOSC (if SOSCEN remains set).
 - **Note 1:** The processor will continue to execute code throughout the clock switching sequence. Timing-sensitive code should not be executed during this time.
 - 2: Direct clock switches between any Primary Oscillator mode with PLL and FRCPLL mode are not permitted. This applies to clock switches in either direction. In these instances, the application must switch to FRC mode as a transitional clock source between the two PLL modes.

A recommended code sequence for a clock switch includes the following:

- Disable interrupts during the OSCCON register unlock and write sequence.
- Execute the unlock sequence for the OSCCON high byte by writing 78h and 9Ah to OSCCON<15:8> in two back-to-back instructions.
- Write new oscillator source to the NOSCx bits in the instruction immediately following the unlock sequence.
- 4. Execute the unlock sequence for the OSCCON low byte by writing 46h and 57h to OSCCON<7:0> in two back-to-back instructions.
- 5. Set the OSWEN bit in the instruction immediately following the unlock sequence.
- Continue to execute code that is not clock-sensitive (optional).
- Invoke an appropriate amount of software delay (cycle counting) to allow the selected oscillator and/or PLL to start and stabilize.
- Check to see if OSWEN is '0'. If it is, the switch
 was successful. If OSWEN is still set, then
 check the LOCK bit to determine the cause of
 the failure

The core sequence for unlocking the OSCCON register and initiating a clock switch is shown in Example 9-1.

EXAMPLE 9-1: BASIC CODE SEQUENCE FOR CLOCK SWITCHING

```
;Place the new oscillator selection in WO
;OSCCONH (high byte) Unlock Sequence
MOV
           #OSCCONH, w1
           #0x78, w2
MOV
           #0x9A, w3
MOV.b
          w2, [w1]
          w3, [w1]
MOV.b
;Set new oscillator selection
MOV.b
           WREG, OSCCONH
;OSCCONL (low byte) unlock sequence
           #OSCCONL, w1
MOV
MOV
           #0x46, w2
MOV
           #0x57, w3
MOV.b
           w2, [w1]
MOV.b
           w3, [w1]
;Start oscillator switch operation
BSET
           OSCCON, #0
```

9.5 FRC Self-Tuning

PIC24FJ128GB204 family devices include an automatic mechanism to calibrate the FRC during run time. This system uses clock recovery from a source of known accuracy to maintain the FRC within a very narrow margin of its nominal 8 MHz frequency. This allows for a frequency accuracy that exceeds 0.25%, which is well within the requirements of the "USB 2.0 Specification".

Note: The self-tune feature maintains sufficient accuracy for operation in USB Device mode. For applications that function as a USB host, a high-accuracy clock source (±0.05%) is still required.

The self-tune system is controlled by the bits in the upper half of the OSCTUN register. Setting the STEN bit (OSCTUN<15>) enables the system, causing it to recover a calibration clock from a source selected by the STSRC bit (OSCTUN<12>). When STSRC = 1, the system uses the Start-of-Frame (SOF) packets from an external USB host for its source. When STSRC = 0, the system uses the crystal controlled SOSC for its calibration source. Regardless of the source, the system uses the TUN<5:0> bits (OSCTUN<5:0>) to change the FRC's frequency. Frequency monitoring and adjustment is dynamic, occurring continuously during run time. While the system is active, the TUNx bits cannot be written to by software.

Note: To use the USB as a clock recovery source (STSRC = 1), the microcontroller must be configured for USB operation and connected to an external USB device.

If the SOSC is to be used as the clock recovery source (STSRC = 0), the SOSC

The self-tune system can generate a hardware interrupt, FSTIF. The interrupt can result from a drift of the FRC from the reference, by greater than 0.2%, in either direction or whenever the frequency deviation is beyond the ability of the TUNx bits to correct (i.e., greater than 1.5%). The STLOCK and STOR status bits (OSCTUN<11,9>) are used to indicate these conditions

must always be enabled.

The STLPOL and STORPOL bits (OSCTUN<10,8>) configure the FSTIF interrupt to occur in the presence or the absence of the conditions. It is the user's responsibility to monitor both the STLOCK and STOR bits to determine the exact cause of the interrupt.

Note: The STLPOL and STORPOL bits should be ignored when the self-tune system is disabled (STEN = 0).

9.6 Oscillator Modes and USB Operation

Because of the timing requirements imposed by USB, an internal clock of 48 MHz is required at all times while the USB module is enabled and not in a suspended operating state. Since this is well beyond the maximum CPU clock speed, a method is provided to internally generate both the USB and system clocks from a single oscillator source. PIC24FJ128GB204 family devices use the same clock structure as most other PIC24FJ devices, but include a two-branch PLL system to generate the two clock signals.

The USB PLL block is shown in Figure 9-2. In this system, the input from the Primary Oscillator is divided down by a PLL prescaler to generate a 4 MHz output. This is used to drive an on-chip, 96 MHz PLL frequency multiplier to drive the two clock branches. One branch uses a fixed, divide-by-2 frequency divider to generate the 48 MHz USB clock. The other branch uses a fixed, divide-by-3 frequency divider and configurable PLL prescaler/divider to generate a range of system clock frequencies. The CPDIVx bits select the system clock speed. Available clock options are listed in Table 9-2.

The USB PLL prescaler does not automatically sense the incoming oscillator frequency. The user must manually configure the PLL divider to generate the required 4 MHz output using the PLLDIV<3:0> Configuration bits. This limits the choices for Primary Oscillator frequency to a total of 8 possibilities, as shown in Table 9-3.

TABLE 9-2: SYSTEM CLOCK OPTIONS DURING USB OPERATION

MCU Clock Division (CPDIV<1:0>)	Microcontroller Clock Frequency
None (00)	32 MHz
÷2 (01)	16 MHz
÷4 (10) ⁽¹⁾	8 MHz
÷8 (11) ⁽¹⁾	4 MHz

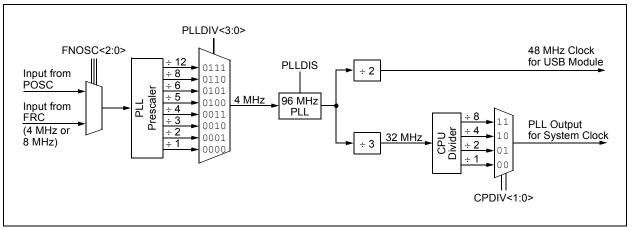
Note 1: Not compatible with USB operation; the USB module must be disabled to use this system clock option.

TABLE 9-3: VALID PRIMARY OSCILLATOR CONFIGURATIONS FOR USB OPERATIONS

Input Oscillator Frequency	Clock Mode	PLL Division (PLLDIV<3:0>)
48 MHz	ECPLL	÷12 (0111)
32 MHz	ECPLL	÷8 (0110)
24 MHz	HSPLL, ECPLL	÷6 (0101)
20 MHz	HSPLL, ECPLL	÷ 5 (0100)
16 MHz	HSPLL, ECPLL	÷4 (0011)
12 MHz	HSPLL, ECPLL	÷3 (0010)
8 MHz	ECPLL, XTPLL, FRCPLL ⁽¹⁾	÷2 (0001)
4 MHz	ECPLL, XTPLL, FRCPLL ⁽¹⁾	÷1 (0000)

Note 1: Requires the use of the FRC self-tune feature to maintain the required clock accuracy.

FIGURE 9-2: USB PLL BLOCK



9.6.1 CONSIDERATIONS FOR USB OPERATION

When using the USB On-The-Go module in PIC24FJ128GB204 family devices, users must always observe these rules in configuring the system clock:

- The Oscillator modes listed in Table 9-3 are the only oscillator configurations that permit USB operation. There is no provision to provide a separate external clock source to the USB module.
- For USB operation, the selected clock source (EC, HS or XT) must meet the USB clock tolerance requirements.
- When the FRCPLL Oscillator mode is used for USB applications, the FRC self-tune system should be used as well. While the FRC is accurate, the only two ways to ensure the level of accuracy required by the "USB 2.0 Specification", throughout the application's operating range, are either the self-tune system or manually changing the TUNx bits.
- The user must always ensure that the FRC source is configured to provide a frequency of 4 MHz or 8 MHz (RCDIV<2:0> = 001 or 000) and that the USB PLL prescaler is configured appropriately.
- All other Oscillator modes are available; however, USB operation is not possible when these modes are selected. They may still be useful in cases where other power levels of operation are desirable and the USB module is not needed (for example, the application is Sleeping and waiting for a bus attachment).

9.7 Reference Clock Output

In addition to the CLKO output (Fosc/2) available in certain Oscillator modes, the device clock in the PIC24FJ128GB204 family devices can also be configured to provide a reference clock output signal to a port pin. This feature is available in all oscillator configurations and allows the user to select a greater range of clock submultiples to drive external devices in the application.

This reference clock output is controlled by the REFOCONL, REFOCONH and REFOTRIML registers (Register 9-4, Register 9-5 and Register 9-6). Setting the ROEN bit (REFOCONL<15>) enables the module. Setting the ROOUT bit (REFOCONL<12>) makes the clock signal available on the REFO pin.

The RODIVx bits (REFOCONH<14:0>) enable the selection of 32768 different clock divider options.

9.7.1 CLOCK SOURCE REQUEST

The ROSELx bits determine different base clock sources for the module.

If the selected clock source has a global device enable (via device Configuration fuse settings), the user must enable the clock source before selecting it as a base clock source.

The ROACTIVE bit (REFOCONL<8>) synchronizes the REFO module during the turn-on and turn-off of the module.

Note: Once the ROEN bit is set, it should not be cleared until the ROACTIVE bit is read as '1'.

9.7.2 CLOCK SWITCHING

The base clock to the module can be switched. First, turn off the module by clearing the ROEN bit (REFOCONL<15> = 0) and wait for the ROACTIVE (REFOCONL<8>) bit to be cleared by the hardware. This avoids a glitch in the REFO output.

The ROTRIMx and RODIVx bits can be changed on-the-fly. Follow the below mentioned steps before changing the ROTRIMx and RODIVx bits.

- REFO is not actively performing the divider switch (ROSWEN = 0).
- Update the ROTRIMx and RODIVx bits with the latest values.
- · Set the ROSWEN bit.
- · Wait for the ROSWEN bit to be cleared by hardware.

The ROTRIMx bits allow a fractional divisor to be added to the integer divisor, specified in the RODIVx register bits.

EQUATION 9-1: FRACTIONAL DIVISOR FOR ROTRIMX BITS

For RODIV<14:0> = 0, No Divide: RODIV<14:0> > 0, Period = 2 * (RODIVx + ROTRIMx)

9.7.3 OPERATION IN SLEEP MODE

The ROSLP and ROSELx bits (REFOCONL<11,3:0>) control the availability of the reference output during Sleep mode.

The ROSLP bit determines if the reference source is available on the REFO pin when the device is in Sleep mode.

To use the reference clock output in Sleep mode, the ROSLP bit must be set and the reference base clock should not be the system clock or peripheral clock (ROSELx bits should not be '0b0000' or '0b0001').

The device clock must also be configured for either:

- One of the Primary modes (EC, HS or XT); the POSCEN bit should be set
- · The Secondary Oscillator bit (SOSCEN) should be set
- · The LPRC Oscillator

If one of the above conditions is not met, then the oscillators on OSC1, OSC2 and SOSCI will be powered down when the device enters Sleep mode.

REGISTER 9-4: REFOCONL: REFERENCE OSCILLATOR CONTROL LOW REGISTER

R/W-0	U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
ROEN	_	ROSIDL	ROOUT	ROSLP	_	ROSWEN	ROACTIVE
bit 15							bit 8

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	_	_	ROSEL3	ROSEL2	ROSEL1	ROSEL0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, rea	d as '0'
-n = Value at POR	'1' = Rit is set	'0' = Rit is cleared	y = Rit is unknown

-n = Value	at POR 11'	= Bit is set	'0' = Bit is cleared	x = Bit is unknov
bit 15	ROEN: Reference 1 = Reference osc 0 = Reference osc		Enable bit	
bit 14	Unimplemented:	Read as '0'		
bit 13	-		ut Stop in Idle Mode bit	
		cillator is disabled i cillator continues to		
bit 12	ROOUT: Reference	ce Clock Output Er	nable bit	
	1 = REFO clock o 0 = REFO clock o	output is driven on to output is disabled	he REFO pin	
bit 11	ROSLP: Reference	e Oscillator Outpu	t in Sleep Mode bit	
		·	nues to run in Sleep mode abled in Sleep mode	
bit 10	Unimplemented:	Read as '0'		
bit 9	ROSWEN: Refere	ence Oscillator Clo	ck Source Switch Enable bit	
		ck source switchin	g is currently in progress g has completed	
bit 8	ROACTIVE: Refe	rence Clock Requ	est Status bit	
			e (user should not update the RE ctive (user can update the REFC	
bit 7-4	Unimplemented:	Read as '0'		
	•	litional ROSELx bit	•	
bit 3-0		ference Clock Sou		
	1001-1111 = Res	served	ces to be used as the reference of	clock:
	1000 = REFI (Ref	ference Clock Inpu	t)	
	0111 = Reserved 0110 = 8x PLL or			
		y Oscillator (SOSC		
	0100 = Low-Powe	er RC Oscillator (Ll	PRC)	
		oscillator (XT, HS, E	- C)	
	0001 = Periphera	l Clock (PBCLK) –	Internal instruction cycle clock, F	CY
	0000 = System C	lock (Fosc)		

REGISTER 9-5: REFOCONH: REFERENCE OSCILLATOR CONTROL HIGH REGISTER

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_				RODIV<14:8>			
bit 15	_			_			bit 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		
RODIV<7:0>									
bit 7							bit 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 15 **Unimplemented:** Read as '0'

bit 14-0 RODIV<14:0>: Reference Oscillator Divisor Select bits

Specifies the 1/2 period of the reference clock in the source clocks.

For example: Period of ref_clk_output ≤ [Reference Source * 2] * RODIV<14:0>:

1111111111111111 = REFO clock is the base clock frequency divided by 65,534 (32,767 * 2)

111111111111110 = REFO clock is the base clock frequency divided by 65,532 (32,766 * 2)

•

•

.

00000000000011 = REFO clock is the base clock frequency divided by 6 (3 * 2)

00000000000000 = REFO clock is the base clock frequency divided by 4 (2 * 2)

00000000000000 = REFO clock is the base clock frequency divided by 2 (1 * 2)

00000000000000 = REFO clock is the same frequency as the base clock (no divider)(1)

Note 1: The ROTRIMx values are ignored.

REGISTER 9-6: REFOTRIML: REFERENCE OSCILLATOR TRIM REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0		
ROTRIM<15:8>									
bit 15							bit 8		

R/W-0	U-0						
ROTRIM7	_	_	_	_	_	_	_
bit 7							bit 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 15-7 **ROTRIM<15:7>:** Reference Oscillator Trim bits

Provides fractional additive to the RODIVx value for the 1/2 period of the REFO clock.

111111111 = 511/512 (0.998046875) divisor added to RODIVx value

1111111110 = 510/512 (0.99609375) divisor added to RODIVx value

•

•

100000000 = 256/512 (0.5000) divisor added to RODIVx value

•

٠

000000010 = 2/512 (0.00390625) divisor added to RODIVx value 000000001 = 1/512 (0.001953125) divisor added to RODIVx value

000000000 = 0/512 (0.0) divisor added to RODIV value

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

9.8 On-Chip PLL

An on-chip PLL (x4, x6, x8) can be selected by the Configuration bits, PLLDIV<3:0>. The Primary Oscillator and FRC sources (FRCDIV) have the option of using this PLL.

Using the internal FRC source, the PLL module can generate the following frequencies, as shown in Table 9-4.

TABLE 9-4: VALID FRC CONFIGURATION FOR ON-CHIP PLL⁽¹⁾

FRC	RCDIV<2:0> (FRCDIV)	x4 PLL	x6 PLL	x8 PLL
8 MHz	000 (divide-by-1)	32 MHz	_	_
8 MHz	001 (divide-by-2)	16 MHz	24 MHz	32 MHz
8 MHz	010 (divide-by-4)	8 MHz	12 MHz	16 MHz

Note 1: The minimum frequency input to the on-chip PLL is 2 MHz.

NOTES:	
--------	--

10.0 POWER-SAVING FEATURES

Note:

This data sheet summarizes the features of this group of PIC24FJ devices. It is not intended to be a comprehensive reference source. For more information, refer to the "dsPIC33/PIC24 Family Reference Manual", "Power-Saving Features with Deep Sleep" (DS39727).

The PIC24FJ128GB204 family of devices provides the ability to manage power consumption by selectively managing clocking to the CPU and the peripherals. In general, a lower clock frequency and a reduction in the number of circuits being clocked reduces consumed power.

PIC24FJ128GB204 family devices manage power consumption with five strategies:

- · Instruction-Based Power Reduction Modes
- · Hardware-Based Power Reduction Features
- · Clock Frequency Control
- Software Controlled Doze Mode
- · Selective Peripheral Control in Software

Combinations of these methods can be used to selectively tailor an application's power consumption, while still maintaining critical application features, such as timing-sensitive communications.

10.1 Overview of Power-Saving Modes

In addition to full-power operation, otherwise known as Run mode, the PIC24FJ128GB204 family of devices offers three instruction-based, power-saving modes and one hardware-based mode:

- Idle
- Sleep (Sleep and Low-Voltage Sleep)
- · Deep Sleep
- VBAT (with and without RTCC)

All four modes can be activated by powering down different functional areas of the microcontroller, allowing progressive reductions of operating and Idle power consumption. In addition, three of the modes can be tailored for more power reduction at a trade-off of some operating features. Table 10-1 lists all of the operating modes, in order of increasing power savings. Table 10-2 summarizes how the microcontroller exits the different modes. Specific information is provided in the following sections.

TABLE 10-1: OPERATING MODES FOR PIC24FJ128GB204 FAMILY DEVICES

			Active Systems					
Mode	Entry	Core	Peripherals	Data RAM Retention	RTCC ⁽¹⁾	DSGPR0/ DSGPR1 Retention		
Run (default)	N/A	Υ	Y	Υ	Υ	Y		
Idle	Instruction	N	Y	Υ	Υ	Y		
Sleep:								
Sleep	Instruction	N	S ⁽²⁾	Υ	Υ	Y		
Low-Voltage Sleep	Instruction + RETEN bit	N	S ⁽²⁾	Y	Y	Y		
Deep Sleep:								
Deep Sleep	Instruction + DSEN bit	N	N	N	Y	Y		
VBAT:								
with RTCC	Hardware	N	N	N	Y	Y		

Note 1: If RTCC is otherwise enabled in firmware.

2: A select peripheral can operate during this mode from LPRC or an external clock.

TABLE 10-2: EXITING POWER-SAVING MODES

	Exit Conditions								Code
Mode	Inter	rupts		Resets		RTCC	WDT	V DD	Execution
	All	INT0	All	POR	MCLR	Alarm	WDT	Restore ⁽²⁾	Resumes
Idle	Υ	Y	Υ	Υ	Y	Y	Υ	N/A	Next instruction
Sleep (all modes)	Y	Υ	Υ	Y	Υ	Y	Υ	N/A	
Deep Sleep	N	Y	N	Y	Υ	Y	Y ⁽¹⁾	N/A	Reset vector
VBAT	N	N	N	N	N	N	N	Υ	Reset vector

Note 1: Deep Sleep WDT.

2: A POR or POR like Reset results whenever VDD is removed and restored in any mode except for Retention Deep Sleep.

10.1.1 INSTRUCTION-BASED POWER-SAVING MODES

Three of the power-saving modes are entered through the execution of the PWRSAV instruction. Sleep mode stops clock operation and halts all code execution. Idle mode halts the CPU and code execution, but allows peripheral modules to continue operation. Deep Sleep mode stops clock operation, code execution, and all peripherals, except RTCC and DSWDT. It also freezes I/O states and removes power to Flash memory, and may remove power to SRAM.

The assembly syntax of the PWRSAV instruction is shown in Example 10-1. Sleep and Idle modes are entered directly with a single assembler command. Deep Sleep requires an additional sequence to unlock and enable the entry into Deep Sleep, which is described in Section 10.4.1 "Entering Deep Sleep Mode".

Note: SLEEP_MODE and IDLE_MODE are constants defined in the assembler include file for the selected device.

Sleep and Idle modes can be exited as a result of an enabled interrupt, WDT time-out or a device Reset. When the device exits these modes, it is said to "wake-up".

The features enabled with the low-voltage/retention regulator results in some changes to the way that Sleep and Deep Sleep modes behave. See Section 10.3 "Sleep Mode" and Section 10.4 "Deep Sleep Mode" for additional information.

10.1.1.1 Interrupts Coincident with Power Save Instructions

Any interrupt that coincides with the execution of a PWRSAV instruction will be held off until entry into Sleep or Idle mode has completed. The device will then wake-up from Sleep or Idle mode.

For Deep Sleep mode, interrupts that coincide with the execution of the PWRSAV instruction may be lost. If the low-voltage/retention regulator is not enabled, the microcontroller resets on leaving Deep Sleep and the interrupt will be lost.

Interrupts that occur during the Deep Sleep unlock sequence will interrupt the mandatory five-instruction cycle sequence timing and cause a failure to enter Deep Sleep. For this reason, it is recommended to disable all interrupts during the Deep Sleep unlock sequence.

EXAMPLE 10-1: PWRSAV INSTRUCTION SYNTAX

```
// Syntax to enter Sleep mode:
PWRSAV #SLEEP_MODE
                      ; Put the device into SLEEP mode
//
//Synatx to enter Idle mode:
PWRSAV
        #IDLE MODE ; Put the device into IDLE mode
// Syntax to enter Deep Sleep mode:
// First use the unlock sequence to set the DSEN bit (see Example 10-2)
BSET
        DSCON, #DSEN ; Enable Deep Sleep
        DSCON, #DSEN
BSET
                         ; Enable Deep Sleep (repeat the command)
PWRSAV
        #SLEEP MODE
                         ; Put the device into Deep SLEEP mode
```

10.1.2 HARDWARE-BASED POWER-SAVING MODE

The hardware-based VBAT mode does not require any action by the user during code development. Instead, it is a hardware design feature that allows the microcontroller to retain critical data (using the DSGPRx registers) and maintain the RTCC when VDD is removed from the application. This is accomplished by supplying a backup power source to a specific power pin. VBAT mode is described in more detail in **Section 10.5 "VBAT Mode"**.

10.1.3 LOW-VOLTAGE/RETENTION REGULATOR

PIC24FJ128GB204 family devices incorporate a second on-chip voltage regulator, designed to provide power to select microcontroller features at 1.2V nominal. This regulator allows features, such as data RAM and the WDT, to be maintained in power-saving modes, where they would otherwise be inactive, or maintain them at a lower power than would otherwise be the case.

The low-voltage/retention regulator is only available when Sleep mode is invoked. It is controlled by the LPCFG Configuration bit (CW1<10>) and in firmware by the RETEN bit (RCON<12>). LPCFG must be programmed (= 0) and the RETEN bit must be set (= 1) for the regulator to be enabled.

10.2 Idle Mode

Idle mode includes these features:

- The CPU will stop executing instructions.
- · The WDT is automatically cleared.
- The system clock source remains active. By default, all peripheral modules continue to operate normally from the system clock source, but can also be selectively disabled (see Section 10.8 "Selective Peripheral Module Control").
- If the WDT or FSCM is enabled, the LPRC will also remain active.

The device will wake from Idle mode on any of these events:

- · Any interrupt that is individually enabled
- · Any device Reset
- · A WDT time-out

On wake-up from Idle, the clock is reapplied to the CPU and instruction execution begins immediately, starting with the instruction following the PWRSAV instruction or the first instruction in the Interrupt Service Routine (ISR).

10.3 Sleep Mode

Sleep mode includes these features:

- The system clock source is shut down. If an on-chip oscillator is used, it is turned off.
- The device current consumption will be reduced to a minimum provided that no I/O pin is sourcing current.
- · The I/O pin directions and states are frozen.
- The Fail-Safe Clock Monitor does not operate during Sleep mode since the system clock source is disabled.
- The LPRC clock will continue to run in Sleep mode if the WDT or RTCC, with LPRC as the clock source, is enabled.
- The WDT, if enabled, is automatically cleared prior to entering Sleep mode.
- Some device features or peripherals may continue to operate in Sleep mode. This includes items, such as the Input Change Notification on the I/O ports or peripherals that use an external clock input. Any peripheral that requires the system clock source for its operation will be disabled in Sleep mode.

The device will wake-up from Sleep mode on any of these events:

- On any interrupt source that is individually enabled
- · On any form of device Reset
- · On a WDT time-out

On wake-up from Sleep, the processor will restart with the same clock source that was active when Sleep mode was entered.

10.3.1 LOW-VOLTAGE/RETENTION SLEEP MODE

Low-Voltage/Retention Sleep mode functions as Sleep mode with the same features and wake-up triggers. The difference is that the low-voltage/retention regulator allows core digital logic voltage (VCORE) to drop to 1.2V nominal. This permits an incremental reduction of power consumption over what would be required if VCORE was maintained at a 1.8V (minimum) level.

Low-Voltage Sleep mode requires a longer wake-up time than Sleep mode, due to the additional time required to bring VCORE back to 1.8V (known as TREG). In addition, the use of the low-voltage/retention regulator limits the amount of current that can be sourced to any active peripherals, such as the RTCC, etc.

10.4 Deep Sleep Mode

Deep Sleep mode provides the lowest levels of power consumption available from the instruction-based modes.

Deep Sleep modes have these features:

- The system clock source is shut down. If an on-chip oscillator is used, it is turned off.
- The device current consumption will be reduced to a minimum.
- The I/O pin directions and states are frozen.
- The Fail-Safe Clock Monitor does not operate during Sleep mode since the system clock source is disabled.
- The LPRC clock will continue to run in Deep Sleep mode if the WDT, or RTCC with LPRC as the clock source, is enabled.
- The dedicated Deep Sleep WDT and BOR systems, if enabled, are used.
- The RTCC and its clock source continue to run, if enabled. All other peripherals are disabled.

Entry into Deep Sleep mode is completely under software control. Exit from the Deep Sleep modes can be triggered from any of the following events:

- · POR event
- MCLR event
- RTCC alarm (If the RTCC is present)
- · External Interrupt 0
- · Deep Sleep Watchdog Timer (DSWDT) time-out

10.4.1 ENTERING DEEP SLEEP MODE

Deep Sleep mode is entered by setting the DSEN bit in the DSCON register and then executing a Sleep command (PWRSAV #SLEEP_MODE), within one instruction cycle, to minimize the chance that Deep Sleep will be spuriously entered.

If the PWRSAV command is not given within one instruction cycle, the DSEN bit will be cleared by the hardware and must be set again by the software before entering Deep Sleep mode. The DSEN bit is also automatically cleared when exiting Deep Sleep mode.

Note:

To re-enter Deep Sleep after a Deep Sleep wake-up, allow a delay of at least 3 TCY after clearing the RELEASE bit.

The sequence to enter Deep Sleep mode is:

- If the application requires the Deep Sleep WDT, enable it and configure its clock source. For more information on Deep Sleep WDT, see Section 10.4.5 "Deep Sleep WDT".
- If the application requires Deep Sleep BOR, enable it by programming the DSBOREN Configuration bit (CW4<6>).
- 3. If the application requires wake-up from Deep Sleep on RTCC alarm, enable and configure the RTCC module. For more information on RTCC, see Section 22.0 "Real-Time Clock and Calendar (RTCC)".
- If needed, save any critical application context data by writing it to the DSGPR0 and DSGPR1 registers (optional).
- Enable Deep Sleep mode by setting the DSEN bit (DSCON<15>).

Note: A repeat sequence is required to set the DSEN bit. The repeat sequence (repeating the instruction twice) is required to write into any of the Deep Sleep registers (DSCON, DSWAKE, DSGPR0, DSGPR1). This is required to prevent the user from entering Deep Sleep by mistake. Any write to these registers has to be done twice to actually complete the write (see Example 10-2).

Enter Deep Sleep mode by issuing 3 NOP commands and then a PWRSAV #0 instruction.

Any time the DSEN bit is set, all bits in the DSWAKE register will be automatically cleared.

EXAMPLE 10-2: THE REPEAT SEQUENCE

```
Example 1:

mov #8000, w2 ; enable DS

mov w2, DSCON ; second write required to actually write to DSCON

Example 2:
bset DSCON, #15 nop nop nop bset DSCON, #15 ; enable DS (two writes required)
```

10.4.2 EXITING DEEP SLEEP MODES

Deep Sleep modes exit on any one of the following events:

- POR event on VDD supply. If there is no DSBOR circuit to re-arm the VDD supply POR circuit, the external VDD supply must be lowered to the natural arming voltage of the POR circuit.
- DSWDT time-out. When the DSWDT times out, the device exits Deep Sleep.
- RTCC alarm (if RTCEN = 1).
- Assertion ('0') of the MCLR pin.
- Assertion of the INTO pin (if the interrupt was enabled before Deep Sleep mode was entered).
 The polarity configuration is used to determine the assertion level ('0' or '1') of the pin that will cause an exit from Deep Sleep mode. Exiting from Deep Sleep mode requires a change on the INTO pin while in Deep Sleep mode.

Note: Any interrupt pending when entering Deep Sleep mode is cleared.

Exiting Deep Sleep generally does not retain the state of the device and is equivalent to a Power-on Reset (POR) of the device. Exceptions to this include the RTCC (if present), which remains operational through the wake-up, the DSGPRx registers and the DSWDT.

Wake-up events that occur from the time Deep Sleep exits, until the time the POR sequence completes, are not ignored. The DSWAKE register will capture ALL wake-up events, from setting the DSEN bit to clearing the RELEASE bit.

The sequence for exiting Deep Sleep mode is:

- After a wake-up event, the device exits Deep Sleep and performs a POR. The DSEN bit is cleared automatically. Code execution resumes at the Reset vector.
- To determine if the device exited Deep Sleep, read the Deep Sleep bit, DPSLP (RCON<10>).
 This bit will be set if there was an exit from Deep Sleep mode. If the bit is set, clear it.
- Determine the wake-up source by reading the DSWAKE register.
- Determine if a DSBOR event occurred during Deep Sleep mode by reading the DSBOR bit (DSCON<1>).
- If application context data has been saved, read it back from the DSGPR0 and DSGPR1 registers.
- 6. Clear the RELEASE bit (DSCON<0>).

10.4.3 SAVING CONTEXT DATA WITH THE DSGPRx REGISTERS

As exiting Deep Sleep mode causes a POR, most Special Function Registers reset to their default POR values. In addition, because VCORE power is not supplied in Deep Sleep mode, information in data RAM may be lost when exiting this mode.

Applications which require critical data to be saved prior to Deep Sleep, may use the Deep Sleep General Purpose registers, DSGPR0 and DSGPR1, or data EEPROM (if available). Unlike other SFRs, the contents of these registers are preserved while the device is in Deep Sleep mode. After exiting Deep Sleep, software can restore the data by reading the registers and clearing the RELEASE bit (DSCON<0>).

Note: User software should enable the DSSWEN (CW4<8>) Configuration Fuse bit for saving critical data in the DSGPRx registers.

10.4.4 I/O PINS IN DEEP SLEEP MODES

During Deep Sleep, the general purpose I/O pins retain their previous states and the Secondary Oscillator (SOSC) will remain running, if enabled. Pins that are configured as inputs (TRISx bit set), prior to entry into Deep Sleep, remain high-impedance during Deep Sleep. Pins that are configured as outputs (TRISx bit clear), prior to entry into Deep Sleep, remain as output pins during Deep Sleep. While in this mode, they continue to drive the output level determined by their corresponding LATx bit at the time of entry into Deep Sleep.

Once the device wakes back up, all I/O pins continue to maintain their previous states, even after the device has finished the POR sequence and is executing application code again. Pins configured as inputs during Deep Sleep remain high-impedance and pins configured as outputs continue to drive their previous value. After waking up, the TRISx and LATx registers, and the SOSCEN bit (OSCCON<1>) are reset. If firmware modifies any of these bits or registers, the I/O will not immediately go to the newly configured states. Once the firmware clears the RELEASE bit (DSCON<0>), the I/O pins are "released". This causes the I/O pins to take the states configured by their respective TRISx and LATx bit values.

This means that keeping the SOSC running after waking up requires the SOSCEN bit to be set before clearing RELEASE.

If the Deep Sleep BOR (DSBOR) is enabled, and a DSBOR or a true POR event occurs during Deep Sleep, the I/O pins will be immediately released, similar to clearing the RELEASE bit. All previous state information will be lost, including the general purpose DSGPR0 and DSGPR1 contents.

If a MCLR Reset event occurs during Deep Sleep, the DSGPRx, DSCON and DSWAKE registers will remain valid, and the RELEASE bit will remain set. The state of the SOSC will also be retained. The I/O pins, however, will be reset to their MCLR Reset state. Since RELEASE is still set, changes to the SOSCEN bit (OSCCON<1>) cannot take effect until the RELEASE bit is cleared.

In all other Deep Sleep wake-up cases, application firmware must clear the RELEASE bit in order to reconfigure the I/O pins.

10.4.5 DEEP SLEEP WDT

To enable the DSWDT in Deep Sleep mode, program the Configuration bit, DSWDTEN (CW4<7>). The device WDT need not be enabled for the DSWDT to function. Entry into Deep Sleep modes automatically resets the DSWDT.

The DSWDT clock source is selected by the DSWDTOSC Configuration bit (CW4<5>). The postscaler options are programmed by the DSWDTPS<4:0> Configuration bits (CW4<4:0>). The minimum time-out period that can be achieved is 1 ms and the maximum is 25.7 days. For more information on the CW4 Configuration register and DSWDT configuration options, refer to Section 30.0 "Special Features".

10.4.5.1 Switching Clocks in Deep Sleep

Both the RTCC and the DSWDT may run from either SOSC or the LPRC clock source. This allows both the RTCC and DSWDT to run without requiring both the LPRC and SOSC to be enabled together, reducing power consumption.

Running the RTCC from LPRC will result in a loss of accuracy in the RTCC of approximately 5 to 10%. If a more accurate RTCC is required, it must be run from the SOSC clock source. The RTCC clock source is selected with the RTCLK<1:0> bits (RTCPWC<11:10>).

Under certain circumstances, it is possible for the DSWDT clock source to be off when entering Deep Sleep mode. In this case, the clock source is turned on automatically (if DSWDT is enabled), without the need for software intervention. However, this can cause a delay in the start of the DSWDT counters. In order to avoid this delay when using SOSC as a clock source, the application can activate SOSC prior to entering Deep Sleep mode.

10.4.6 CHECKING AND CLEARING THE STATUS OF DEEP SLEEP

Upon entry into Deep Sleep mode, the status bit, DPSLP (RCON<10>), becomes set and must be cleared by the software.

On power-up, the software should read this status bit to determine if the Reset was due to an exit from Deep Sleep mode and clear the bit if it is set. Of the four possible combinations of DPSLP and POR bit states, the following three cases can be considered:

- Both the DPSLP and POR bits are cleared. In this case, the Reset was due to some event other than a Deep Sleep mode exit.
- The DPSLP bit is clear, but the POR bit is set; this
 is a normal Power-on Reset.
- Both the DPSLP and POR bits are set. This
 means that Deep Sleep mode was entered, the
 device was powered down and Deep Sleep mode
 was exited.

10.4.7 POWER-ON RESETS (PORs)

VDD voltage is monitored to produce PORs. Since exiting from Deep Sleep mode functionally looks like a POR, the technique described in **Section 10.4.6** "Checking and Clearing the Status of Deep Sleep" should be used to distinguish between Deep Sleep and a true POR event. When a true POR occurs, the entire device, including all Deep Sleep logic (Deep Sleep registers, RTCC, DSWDT, etc.) is reset.

10.5 VBAT Mode

This mode represents the lowest power state that the microcontroller can achieve and still resume operation. VBAT mode is automatically triggered when the microcontroller's main power supply on VDD fails. When this happens, the microcontroller's on-chip power switch connects to a backup power source, such as a battery, supplied to the VBAT pin. This maintains a few key systems at an extremely low-power draw until VDD is restored.

The power supplied on VBAT only runs two systems: the RTCC and the Deep Sleep Semaphore registers (DSGPR0 and DSGPR1). To maintain these systems during a sudden loss of VDD, it is essential to connect a power source, other than VDD or AVDD, to the VBAT pin.

When the RTCC is enabled, it continues to operate with the same clock source (SOSC or LPRC) that was selected prior to entering VBAT mode. There is no provision to switch to a lower power clock source after the mode switch.

Since the loss of VDD is usually an unforeseen event, it is recommended that the contents of the Deep Sleep Semaphore registers be loaded with the data to be retained at an early point in code execution.

10.5.1 VBAT MODE WITH NO RTCC

By disabling RTCC operation during VBAT mode, power consumption is reduced to the lowest of all power-saving modes. In this mode, only the Deep Sleep Semaphore registers are maintained.

10.5.2 WAKE-UP FROM VBAT MODES

When VDD is restored to a device in VBAT mode, it automatically wakes. Wake-up occurs with a POR, after which, the device starts executing code from the Reset vector. All SFRs, except the Deep Sleep Semaphores, are reset to their POR values. IF the RTCC was not configured to run during VBAT mode, it will remain disabled and RTCC will not run. Wake-up timing is similar to that for a normal POR.

To differentiate a wake-up from VBAT mode, from other POR states, check the VBAT status bit (RCON2<0>). If this bit is set while the device is starting to execute the code from the Reset vector, it indicates that there has been an exit from VBAT mode. The application must clear the VBAT bit to ensure that future VBAT wake-up events are captured.

If a POR occurs without a power source connected to the VBAT pin, the VBPOR bit (RCON2<1>) is set. If this bit is set on a Power-on Reset, it indicates that a battery needs to be connected to the VBAT pin.

In addition, if the VBAT power source falls below the level needed for Deep Sleep Semaphore operation while in VBAT mode (e.g., the battery has been drained), the VBPOR bit will be set. VBPOR is also set when the microcontroller is powered up the very first time, even if power is supplied to VBAT.

10.5.3 I/O PINS DURING VBAT MODES

All I/O pins switch to Input mode during VBAT mode. The only exceptions are the SOSCI and SOSCO pins, which maintain their states if the Secondary Oscillator is being used as the RTCC clock source. It is the user's responsibility to restore the I/O pins to their proper states using the TRISx and LATx bits once VDD has been restored.

10.5.4 SAVING CONTEXT DATA WITH THE DSGPRx REGISTERS

As with Deep Sleep mode (i.e., without the low-voltage/retention regulator), all SFRs are reset to their POR values after VDD has been restored. Only the Deep Sleep Semaphore registers are preserved. Applications which require critical data to be saved should save it in DSGPR0 and DSGPR1.

Note: If the VBAT mode is not used, it is recommended to connect the VBAT pin to VDD.

The POR should be enabled for the reliable operation of the VBAT.

REGISTER 10-1: DSCON: DEEP SLEEP CONTROL REGISTER⁽¹⁾

R/W-0	U-0						
DSEN	_	_	_	_	_	_	_
bit 15	•						bit 8

U-0	U-0	U-0	U-0	U-0	r-0	R/W-0	R/C-0, HS
_	_	_	_	_	_	DSBOR ⁽²⁾	RELEASE
bit 7							bit 0

Legend:	C = Clearable bit	U = Unimplemented bit, read as '0'	
R = Readable bit	W = Writable bit	HS = Hardware Settable bit	r = Reserved bit
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15 DSEN: Deep Sleep Enable bit

1 = Enters Deep Sleep on execution of PWRSAV #0 0 = Enters normal Sleep on execution of PWRSAV #0

bit 14-3 **Unimplemented:** Read as '0' bit 2 **Reserved:** Maintain as '0'

bit 1 DSBOR: Deep Sleep BOR Event bit⁽²⁾

1 = The DSBOR was active and a BOR event was detected during Deep Sleep

0 = The DSBOR was not active, or was active, but did not detect a BOR event during Deep Sleep

bit 0 RELEASE: I/O Pin State Release bit

1 = Upon waking from Deep Sleep, I/O pins maintain their states previous to the Deep Sleep entry

0 = Releases I/O pins from their state previous to Deep Sleep entry, and allows their respective TRISx and LATx bits to control their states

Note 1: All register bits are reset only in the case of a POR event outside of Deep Sleep mode.

2: Unlike all other events, a Deep Sleep BOR event will NOT cause a wake-up from Deep Sleep; this re-arms the POR.

REGISTER 10-2: DSWAKE: DEEP SLEEP WAKE-UP SOURCE REGISTER (1)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0, HS
_	_	_	_		_		DSINT0
bit 15							bit 8

R/W-0, HS	U-0	U-0	R/W-0, HS	R/W-0, HS	R/W-0, HS	U-0	U-0
DSFLT	_	_	DSWDT	DSRTCC	DSMCLR	_	_
bit 7							bit 0

Legend:HS = Hardware Settable bitR = Readable bitW = Writable bitU = Unimplemented bit, read as '0'-n = Value at POR'1' = Bit is set'0' = Bit is clearedx = Bit is unknown

bit 15-9 **Unimplemented:** Read as '0'

bit 8 **DSINT0:** Deep Sleep Interrupt-on-Change bit

1 = Interrupt-on-change was asserted during Deep Sleep0 = Interrupt-on-change was not asserted during Deep Sleep

bit 7 DSFLT: Deep Sleep Fault Detect bit

1 = A Fault occurred during Deep Sleep and some Deep Sleep configuration settings may have been

corrupted

0 = No Fault was detected during Deep Sleep

bit 6-5 **Unimplemented:** Read as '0'

bit 4 **DSWDT:** Deep Sleep Watchdog Timer Time-out bit

1 = The Deep Sleep Watchdog Timer timed out during Deep Sleep

0 = The Deep Sleep Watchdog Timer did not time out during Deep Sleep

bit 3 DSRTCC: Deep Sleep Real-Time Clock and Calendar Alarm bit

1 = The Real-Time Clock and Calendar triggered an alarm during Deep Sleep

 $_{\mathrm{0}}$ = The Real-Time Clock and Calendar did not trigger an alarm during Deep Sleep

bit 2 DSMCLR: Deep Sleep MCLR Event bit

1 = The $\overline{\text{MCLR}}$ pin was active and was asserted during Deep Sleep

0 = The MCLR pin was not active, or was active, but not asserted during Deep Sleep

bit 1-0 Unimplemented: Read as '0'

Note 1: All register bits are cleared when the DSEN (DSCON<15>) bit is set.

REGISTER 10-3: RCON2: RESET AND SYSTEM CONTROL REGISTER 2

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

U-0	U-0	U-0	r-0	R/CO-1	R/CO-1	R/CO-1	R/CO-0
_	_	_	_	VDDBOR ⁽¹⁾	VDDPOR ^(1,2)	VBPOR ^(1,3)	VBAT ⁽¹⁾
bit 7							bit 0

Legend:	CO = Clearable Only bit	nly bit r = Reserved bit	
R = Readable bit	W = Writable bit	U = Unimplemented bit, read	d as '0'
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-5 **Unimplemented:** Read as '0'

bit 4 Reserved: Maintain as '0'

bit 3 **VDDBOR:** VDD Brown-out Reset Flag bit⁽¹⁾

1 = A VDD Brown-out Reset has occurred (set by hardware)

0 = A VDD Brown-out Reset has not occurred

bit 2 **VDDPOR:** VDD Power-on Reset Flag bit^(1,2)

1 = A VDD Power-on Reset has occurred (set by hardware)

0 = A VDD Power-on Reset has not occurred

bit 1 **VBPOR:** VBAT Power-on Reset Flag bit^(1,3)

1 = A VBAT POR has occurred (no battery connected to the VBAT pin or VBAT power is below Deep Sleep Semaphore retention level; set by hardware)

0 = A VBAT POR has not occurred

bit 0 **VBAT:** VBAT Flag bit⁽¹⁾

1 = A POR exit has occurred while power is applied to the VBAT pin (set by hardware)

0 = A POR exit from VBAT has not occurred

Note 1: This bit is set in hardware only; it can only be cleared in software.

2: This bit indicates a VDD Power-on Reset. Setting the POR bit (RCON<0>) indicates a VCORE Power-on Reset.

3: This bit is set when the device is originally powered up, even if power is present on VBAT.

10.6 Clock Frequency and Clock Switching

In Run and Idle modes, all PIC24FJ devices allow for a wide range of clock frequencies to be selected under application control. If the system clock configuration is not locked, users can choose low-power or high-precision oscillators by simply changing the NOSCx bits. The process of changing a system clock during operation, as well as limitations to the process, are discussed in more detail in Section 9.0 "Oscillator Configuration".

10.7 Doze Mode

Generally, changing clock speed and invoking one of the power-saving modes are the preferred strategies for reducing power consumption. There may be circumstances, however, where this is not practical. For example, it may be necessary for an application to maintain uninterrupted synchronous communication, even while it is doing nothing else. Reducing system clock speed may introduce communication errors, while using a power-saving mode may stop communications completely.

Doze mode is a simple and effective alternative method to reduce power consumption while the device is still executing code. In this mode, the system clock continues to operate from the same source and at the same speed. Peripheral modules continue to be clocked at the same speed, while the CPU clock speed is reduced. Synchronization between the two clock domains is maintained, allowing the peripherals to access the SFRs while the CPU executes code at a slower rate.

Doze mode is enabled by setting the DOZEN bit (CLKDIV<11>). The ratio between peripheral and core clock speed is determined by the DOZE<2:0> bits (CLKDIV<14:12>). There are eight possible configurations, from 1:1 to 1:128, with 1:1 being the default.

It is also possible to use Doze mode to selectively reduce power consumption in event driven applications. This allows clock-sensitive functions, such as synchronous communications, to continue without interruption while the CPU Idles, waiting for something to invoke an interrupt routine. Enabling the automatic return to full-speed CPU operation on interrupts is enabled by setting the ROI bit (CLKDIV<15>). By default, interrupt events have no effect on Doze mode operation.

10.8 Selective Peripheral Module Control

Idle and Doze modes allow users to substantially reduce power consumption by slowing or stopping the CPU clock. Even so, peripheral modules still remain clocked, and thus, consume power. There may be cases where the application needs what these modes do not provide: the allocation of power resources to CPU processing, with minimal power consumption from the peripherals.

PIC24F devices address this requirement by allowing peripheral modules to be selectively disabled, reducing or eliminating their power consumption. This can be done with two control bits:

- The Peripheral Enable bit, generically named, "XXXEN", located in the module's main control SER
- The Peripheral Module Disable (PMD) bit, generically named, "XXXMD", located in one of the PMDx Control registers (XXXMD bits are in the PMD1, PMD2, PMD3, PMD4, PMD6, PMD7, PMD8 registers).

Both bits have similar functions in enabling or disabling its associated module. Setting the PMD bit for a module disables all clock sources to that module, reducing its power consumption to an absolute minimum. In this state, the control and status registers associated with the peripheral will also be disabled, so writes to those registers will have no effect and read values will be invalid. Many peripheral modules have a corresponding PMD bit.

In contrast, disabling a module by clearing its XXXEN bit disables its functionality, but leaves its registers available to be read and written to. Power consumption is reduced, but not by as much as the use of the PMD bits. Most peripheral modules have an enable bit; exceptions include capture, compare and RTCC.

To achieve more selective power savings, peripheral modules can also be selectively disabled when the device enters Idle mode. This is done through the control bit of the generic name format, "XXXSIDL". By default, all modules that can operate during Idle mode will do so. Using the disable on Idle feature disables the module while in Idle mode, allowing further reduction of power consumption during Idle mode, enhancing power savings for extremely critical power applications.

0GBZU4 	FAIVIIL	. Y		
	8GB2U4	8GBZU4 FAMIL	8GBZU4 FAIVIILY	8GBZU4 FAMILY

11.0 I/O PORTS

Note:

This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the "dsPIC33/PIC24 Family Reference Manual", "I/O Ports with Peripheral Pin Select (PPS)" (DS39711). The information in this data sheet supersedes the information in the FRM.

All of the device pins (except VDD, VSS, MCLR and OSCI/CLKI) are shared between the peripherals and the Parallel I/O ports. All I/O input ports feature Schmitt Trigger (ST) inputs for improved noise immunity.

11.1 Parallel I/O (PIO) Ports

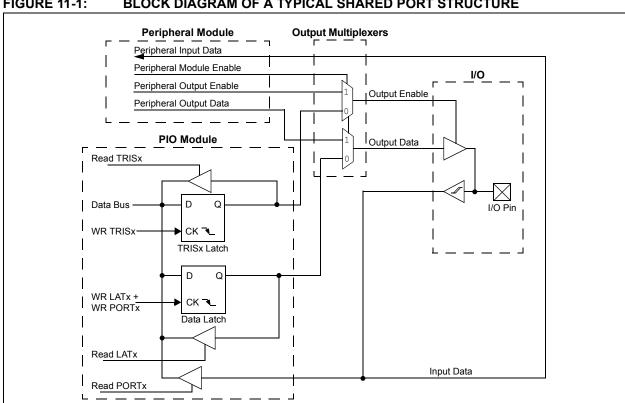
A Parallel I/O port that shares a pin with a peripheral is. in general, subservient to the peripheral. The peripheral's output buffer data and control signals are provided to a pair of multiplexers. The multiplexers select whether the peripheral or the associated port has ownership of the output data and control signals of the I/O pin. The logic also prevents "loop through", in which a port's digital output can drive the input of a peripheral that shares the same pin. Figure 11-1 shows how ports are shared with other peripherals and the associated I/O pin to which they are connected.

When a peripheral is enabled and the peripheral is actively driving an associated pin, the use of the pin as a general purpose output pin is disabled. The I/O pin may be read, but the output driver for the parallel port bit will be disabled. If a peripheral is enabled, but the peripheral is not actively driving a pin, that pin may be driven by a port.

All port pins have three registers directly associated with their operation as digital I/Os and one register associated with their operation as analog inputs. The Data Direction register (TRIS) determines whether the pin is an input or an output. If the data direction bit is a '1', then the pin is an input. All port pins are defined as inputs after a Reset. Reads from the Output Latch register (LAT), read the latch; writes to the latch, write the latch. Reads from the port (PORT), read the port pins; writes to the port pins, write the latch.

Any bit, and its associated data and control registers that are not valid for a particular device, will be disabled. That means the corresponding LATx and TRISx registers, and the port pin, will read as zeros.

When a pin is shared with another peripheral or function that is defined as an input only, it is regarded as a dedicated port because there is no other competing source of inputs.



BLOCK DIAGRAM OF A TYPICAL SHARED PORT STRUCTURE FIGURE 11-1:

11.1.1 I/O PORT WRITE/READ TIMING

One instruction cycle is required between a port direction change or port write operation and a read operation of the same port. Typically, this instruction would be a NOP.

11.1.2 OPEN-DRAIN CONFIGURATION

In addition to the PORTx, LATx and TRISx registers for data control, each port pin can also be individually configured for either a digital or open-drain output. This is controlled by the Open-Drain Control register, ODCx, associated with each port. Setting any of the bits configures the corresponding pin to act as an open-drain output.

The open-drain feature allows the generation of outputs higher than VDD (e.g., 5V), on any desired digital only pins, by using external pull-up resistors. The maximum open-drain voltage allowed is the same as the maximum VIH specification.

11.2 Configuring Analog Port Pins (ANSx)

The ANSx and TRISx registers control the operation of the pins with analog function. Each port pin with analog function is associated with one of the ANSx bits (see Register 11-1 through Register 11-3), which decides if the pin function should be analog or digital. Refer to Table 11-1 for detailed behavior of the pin for different ANSx and TRISx bit settings.

When reading the PORTx register, all pins configured as analog input channels will read as cleared (a low level).

11.2.1 ANALOG INPUT PINS AND VOLTAGE CONSIDERATIONS

The voltage tolerance of pins used as device inputs is dependent on the pin's input function. Most input pins are able to handle DC voltages of up to 5.5V, a level typical for digital logic circuits. However, several pins can only tolerate voltages up to VDD. Voltage excursions beyond VDD on these pins should always be avoided.

Table 11-2 summarizes the different voltage tolerances. For more information, refer to **Section 33.0** "Electrical Characteristics" for more details.

TABLE 44.4.	CONFIGURING ANALOG/DIGITAL	FUNCTION OF AN UO BIN
1481 F 11-1.	CONFIGURING ANALOGADICALIAL	FUNGTION OF AN I/O PIN

Pin Function	ANSx Setting	TRISx Setting	Comments
Analog Input	1	1	It is recommended to keep ANSx = 1.
Analog Output	1	1	It is recommended to keep ANSx = 1.
Digital Input	0	1	Firmware must wait at least one instruction cycle after configuring a pin as a digital input before a valid input value can be read.
Digital Output	0	0	Make sure to disable the analog output function on the pin if any is present.

TABLE 11-2: INPUT VOLTAGE LEVELS FOR PORT OR PIN TOLERATED DESCRIPTION INPUT

Port or Pin	Tolerated Input	Description		
PORTA<10:7,4> ⁽¹⁾		Tile of a six of the object of the original six of the		
PORTB<11:10,8:4>	5.5V	Tolerates input levels above VDD; useful for most standard logic.		
PORTC<9:3>(1)		ioi most standard logic.		
PORTA<3:0>				
PORTB<15:13,9,3:0>	Vdd	Only VDD input levels are tolerated.		
PORTC<2:0>(1)				

Note 1: Not all of these pins are implemented in 28-pin devices. Refer to **Section 1.0 "Device Overview"** for a complete description of port pin implementation.

REGISTER 11-1: ANSA: PORTA ANALOG FUNCTION SELECTION REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

U-0	U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1
_	_	_	_		ANSA	\<3:0>	
bit 7							bit 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 15-4 **Unimplemented:** Read as '0'

bit 3-0 ANSA<3:0>: PORTA Analog Function Selection bits

1 = Pin is configured in Analog mode; I/O port read is disabled0 = Pin is configured in Digital mode; I/O port read is enabled

REGISTER 11-2: ANSB: PORTB ANALOG FUNCTION SELECTION REGISTER

R/W-1	R/W-1	R/W-1	U-0	U-0	U-0	R/W-1	U-0
	ANSB<15:13>		_	_	_	ANSB9	_
bit 15							bit 8

U-0	R/W-1	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1
_	ANSB6	_	_		ANSE	3<3:0>	
bit 7							bit 0

Legend:

R = Readable bit W = Writable bit U = Unimplemented bit, read as '0'

bit 15-13 ANSB<15:13>: PORTB Analog Function Selection bits

1 = Pin is configured in Analog mode; I/O port read is disabled 0 = Pin is configured in Digital mode; I/O port read is enabled

bit 12-10 **Unimplemented:** Read as '0'

bit 9 ANSB9: PORTB Analog Function Selection bit

1 = Pin is configured in Analog mode; I/O port read is disabled 0 = Pin is configured in Digital mode; I/O port read is enabled

bit 8-7 **Unimplemented:** Read as '0'

bit 6 ANSB6: PORTB Analog Function Selection bit

1 = Pin is configured in Analog mode; I/O port read is disabled 0 = Pin is configured in Digital mode; I/O port read is enabled

bit 5-4 **Unimplemented:** Read as '0'

bit 3-0 ANSB<3:0>: PORTB Analog Function Selection bits

1 = Pin is configured in Analog mode; I/O port read is disabled0 = Pin is configured in Digital mode; I/O port read is enabled

REGISTER 11-3: ANSC: PORTC ANALOG FUNCTION SELECTION REGISTER⁽¹⁾

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-1	R/W-1
_	_	_	_	_		ANSC<2:0>	
bit 7							bit 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 15-3 **Unimplemented:** Read as '0'

bit 2-0 ANSC<2:0>: Analog Function Selection bits

1 = Pin is configured in Analog mode; I/O port read is disabled 0 = Pin is configured in Digital mode; I/O port read is enabled

Note 1: These pins are not available in 28-pin devices.

11.3 Input Change Notification (ICN)

The Input Change Notification function of the I/O ports allows the PIC24FJ128GB204 family of devices to generate interrupt requests to the processor in response to a Change-of-State (COS) on selected input pins. This feature is capable of detecting input Change-of-States, even in Sleep mode, when the clocks are disabled. Depending on the device pin count, there are up to 82 external inputs that may be selected (enabled) for generating an interrupt request on a Change-of-State.

Registers, CNEN1 through CNEN3, contain the interrupt enable control bits for each of the CN input pins. Setting any of these bits enables a CN interrupt for the corresponding pins.

Each CN pin has both a weak pull-up and a weak pull-down connected to it. The pull-ups act as a current source that is connected to the pin, while the pull-downs act as a current sink that is connected to the pin. These eliminate the need for external resistors when push button or keypad devices are connected. The pull-ups and pull-downs are separately enabled using the CNPU1 through CNPU3 registers (for pull-ups), and the CNPD1 through CNPD3 registers (for pull-downs). Each CN pin has individual control bits for its pull-up and pull-down. Setting a control bit enables the weak pull-up or pull-down for the corresponding pin.

When the internal pull-up is selected, the pin pulls up to VDD-1.1V (typical). When the internal pull-down is selected, the pin pulls down to Vss.

Note:

Pull-ups on Input Change Notification pins should always be disabled whenever the port pin is configured as a digital output.

EXAMPLE 11-1: PORT READ/WRITE IN ASSEMBLY

```
MOV 0xFF00, W0 ; Configure PORTB<15:8> as inputs
MOV W0, TRISB ; and PORTB<7:0> as outputs
NOP ; Delay 1 cycle
BTSS PORTB, #13 ; Next Instruction
```

EXAMPLE 11-2: PORT READ/WRITE IN 'C'

11.4 Peripheral Pin Select (PPS)

A major challenge in general purpose devices is providing the largest possible set of peripheral features while minimizing the conflict of features on I/O pins. In an application that needs to use more than one peripheral multiplexed on a single pin, inconvenient work arounds in application code, or a complete redesign, may be the only option.

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

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

11.4.1 AVAILABLE PINS

The PPS feature is used with a range of up to 44 pins, depending on the particular device and its pin count. Pins that support the Peripheral Pin Select feature include the designation, "RPn" or "RPln", in their full pin designation, where "n" is the remappable pin number. "RP" is used to designate pins that support both remappable input and output functions, while "RPI" indicates pins that support remappable input functions only.

PIC24FJ128GB204 family devices support a larger number of remappable input only pins than remappable input/output pins. In this device family, there are up to 25 remappable input/output pins, depending on the pin count of the particular device selected. These pins are numbered, RP0 through RP25.

See Table 1-3 for a summary of pinout options in each package offering.

11.4.2 AVAILABLE PERIPHERALS

The peripherals managed by the PPS are all digital only peripherals. These include general serial communications (UART and SPI), general purpose timer clock inputs, timer related peripherals (input capture and output compare) and external interrupt inputs. Also included are the outputs of the comparator module, since these are discrete digital signals.

PPS is not available for these peripherals:

- I²C[™] (input and output)
- USB (all module inputs and outputs)
- · Change Notification Inputs
- RTCC Alarm Output(s)
- EPMP Signals (input and output)
- Analog (inputs and outputs)
- INITO

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

11.4.2.1 Peripheral Pin Select Function Priority

Pin-selectable peripheral outputs (e.g., OC, UART transmit) will take priority over general purpose digital functions on a pin, such as EPMP and port I/O. Specialized digital outputs (e.g., USB on USB-enabled devices) will take priority over PPS outputs on the same pin. The pin diagrams list peripheral outputs in the order of priority. Refer to them for priority concerns on a particular pin.

Unlike PIC24F devices with fixed peripherals, pin-selectable peripheral inputs will never take ownership of a pin. The pin's output buffer will be controlled by the TRISx setting or by a fixed peripheral on the pin. If the pin is configured in Digital mode, then the PPS input will operate correctly. If an analog function is enabled on the pin, the PPS input will be disabled.

11.4.3 CONTROLLING PERIPHERAL PIN SELECT

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

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

11.4.3.1 Input Mapping

The inputs of the Peripheral Pin Select options are mapped on the basis of the peripheral; that is, a control register associated with a peripheral dictates the pin it will be mapped to. The RPINRx registers are used to configure peripheral input mapping (see Register 11-4 through Register 11-22).

Each register contains two sets of 6-bit fields, with each set associated with one of the pin-selectable peripherals. Programming a given peripheral's bit field with an appropriate 6-bit value maps the RPn/RPIn pin with that value to that peripheral. For any given device, the valid range of values for any of the bit fields corresponds to the maximum number of Peripheral Pin Selections supported by the device.

TABLE 11-3: SELECTABLE INPUT SOURCES (MAPS INPUT TO FUNCTION)(1)

Input Name	Function Name	Register	Function Mapping Bits
DSM Modulation Input	MDMIN	RPINR30	MDMIR<5:0>
DSM Carrier 1 Input	MDCIN1	RPINR31	MDC1R<5:0>
DSM Carrier 2 Input	MDCIN2	RPINR31	MDC2R<5:0>
External Interrupt 1	INT1	RPINR0	INT1R<5:0>
External Interrupt 2	INT2	RPINR1	INT2R<5:0>
External Interrupt 3	INT3	RPINR1	INT3R<5:0>
External Interrupt 4	INT4	RPINR2	INT4R<5:0>
Input Capture 1	IC1	RPINR7	IC1R<5:0>
Input Capture 2	IC2	RPINR7	IC2R<5:0>
Input Capture 3	IC3	RPINR8	IC3R<5:0>
Input Capture 4	IC4	RPINR8	IC4R<5:0>
Input Capture 5	IC5	RPINR9	IC5R<5:0>
Input Capture 6	IC6	RPINR9	IC6R<5:0>
Output Compare Fault A	OCFA	RPINR11	OCFAR<5:0>
Output Compare Fault B	OCFB	RPINR11	OCFBR<5:0>
Output Compare Trigger 1	OCTRIG1	RPINR0	OCTRIG1R<5:0>
Output Compare Trigger 2	OCTRIG2	RPINR2	OCTRIG2R<5:0>
SPI1 Clock Input	SCK1IN	RPINR20	SCK1R<5:0>
SPI1 Data Input	SDI1	RPINR20	SDI1R<5:0>
SPI1 Slave Select Input	SS1IN	RPINR21	SS1R<5:0>
SPI2 Clock Input	SCK2IN	RPINR22	SCK2R<5:0>
SPI2 Data Input	SDI2	RPINR22	SDI2R<5:0>
SPI2 Slave Select Input	SS2IN	RPINR23	SS2R<5:0>
SPI3 Clock Input	SCK3IN	RPINR28	SCK3R<5:0>
SPI3 Data Input	SDI3	RPINR28	SDI3R<5:0>
SPI3 Slave Select Input	SS3IN	RPINR29	SS3R<5:0>
Generic Timer External Clock	TMRCK	RPINR23	TMRCKR<5:0>
UART1 Clear-to-Send	U1CTS	RPINR18	U1CTSR<5:0>
UART1 Receive	U1RX	RPINR18	U1RXR<5:0>
UART2 Clear-to-Send	U2CTS	RPINR19	U2CTSR<5:0>
UART2 Receive	U2RX	RPINR19	U2RXR<5:0>
UART3 Clear-to-Send	U3CTS	RPINR21	U3CTSR<5:0>
UART3 Receive	U3RX	RPINR17	U3RXR<5:0>
UART4 Clear-to-Send	U4CTS	RPINR27	U4CTSR<5:0>
UART4 Receive	U4RX	RPINR27	U4RXR<5:0>

Note 1: Unless otherwise noted, all inputs use the Schmitt Trigger (ST) input buffers.

11.4.3.2 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 RPORx registers are used to control output mapping. Each register contains two 6-bit fields, with each field being associated with one RPn pin (see Register 11-23 through Register 11-35). The value of the bit field

corresponds to one of the peripherals and that peripheral's output is mapped to the pin (see Table 11-4).

Because of the mapping technique, the list of peripherals for output mapping also includes a null value of '000000'. This permits any given pin to remain disconnected from the output of any of the pin-selectable peripherals.

TABLE 11-4: SELECTABLE OUTPUT SOURCES (MAPS FUNCTION TO OUTPUT)

Function	Output Name	
NULL ⁽²⁾	Null	
C1OUT	Comparator 1 Output	
C2OUT	Comparator 2 Output	
U1TX	UART1 Transmit	
U1RTS ⁽³⁾	UART1 Request-to-Send	
U2TX	UART2 Transmit	
U2RTS ⁽³⁾	UART2 Request-to-Send	
SDO1	SPI1 Data Output	
SCK1OUT	SPI1 Clock Output	
SS1OUT	SPI1 Slave Select Output	
SDO2	SPI2 Data Output	
SCK2OUT	SPI2 Clock Output	
SS2OUT	SPI2 Slave Select Output	
OC1	Output Compare 1	
OC2	Output Compare 2	
OC3	Output Compare 3	
OC4	Output Compare 4	
OC5	Output Compare 5	
OC6	Output Compare 6	
U3TX	UART3 Transmit	
U3RTS	UART3 Request-to-Send	
U4TX	UART4 Transmit	
U4RTS ⁽³⁾	UART4 Request-to-Send	
SDO3	SPI3 Data Output	
SCK3OUT	SPI3 Clock Output	
SS3OUT	SPI3 Slave Select Output	
C3OUT	Comparator 3 Output	
MDOUT	DSM Modulator Output	
	NULL ⁽²⁾ C10UT C2OUT U1TX U1RTS ⁽³⁾ U2TX U2RTS ⁽³⁾ SD01 SCK10UT SS10UT SD02 SCK2OUT SS2OUT OC1 OC2 OC3 OC4 OC5 OC6 U3TX U3RTS U4TX U4RTS ⁽³⁾ SD03 SCK3OUT SS3OUT C3OUT	

Note 1: Setting the RPORx register with the listed value assigns that output function to the associated RPn pin.

^{2:} The NULL function is assigned to all RPn outputs at device Reset and disables the RPn output function.

^{3:} IrDA® BCLK functionality uses this output.

11.4.3.3 Mapping Limitations

The control schema of the Peripheral Pin Select is extremely flexible. Other than systematic blocks that prevent signal contention, caused by two physical pins being configured as the same functional input or two functional outputs configured as the same pin, there are no hardware enforced lockouts. The flexibility extends to the point of allowing a single input to drive multiple peripherals or a single functional output to drive multiple output pins.

11.4.3.4 Mapping Exceptions for PIC24FJ128GB204 Family Devices

Although the PPS registers theoretically allow for up to 24 remappable I/O pins, not all of these are implemented in all devices. For PIC24FJ128GB204 family devices, the maximum number of remappable pins available is 24, which includes one input only pin. The differences in available remappable pins are summarized in Table 11-5.

When developing applications that use remappable pins, users should also keep these things in mind:

- For the RPINRx registers, bit combinations corresponding to an unimplemented pin for a particular device are treated as invalid; the corresponding module will not have an input mapped to it.
- For RPORx registers, the bit fields corresponding to an unimplemented pin will also be unimplemented; writing to these fields will have no effect.

11.4.4 CONTROLLING CONFIGURATION CHANGES

Because peripheral remapping can be changed during run time, some restrictions on peripheral remapping are needed to prevent accidental configuration changes. PIC24F devices include three features to prevent alterations to the peripheral map:

- · Control register lock sequence
- · Continuous state monitoring
- · Configuration bit remapping lock

11.4.4.1 Control Register Lock

Under normal operation, writes to the RPINRx and RPORx registers are not allowed. Attempted writes will appear to execute normally, but the contents of the registers will remain unchanged. To change these registers, they must be unlocked in hardware. The register lock is controlled by the IOLOCK bit (OSCCON<6>). Setting IOLOCK prevents writes to the control registers; clearing IOLOCK allows writes.

To set or clear IOLOCK, a specific command sequence must be executed:

- Write 46h to OSCCON<7:0>.
- 2. Write 57h to OSCCON<7:0>.
- 3. Clear (or set) IOLOCK as a single operation.

Unlike the similar sequence with the oscillator's LOCK bit, IOLOCK remains in one state until changed. This allows all of the Peripheral Pin Selects to be configured with a single unlock sequence, followed by an update to all control registers, then locked with a second lock sequence.

11.4.4.2 Continuous State Monitoring

In addition to being protected from direct writes, the contents of the RPINRx and RPORx registers are constantly monitored in hardware by shadow registers. If an unexpected change in any of the registers occurs (such as cell disturbances caused by ESD or other external events), a Configuration Mismatch Reset will be triggered.

11.4.4.3 Configuration Bit Pin Select Lock

As an additional level of safety, the device can be configured to prevent more than one write session to the RPINRx and RPORx registers. The IOL1WAY (CW4<15>) Configuration bit blocks the IOLOCK bit from being cleared after it has been set once. If IOLOCK remains set, the register unlock procedure will 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. Programming IOL1WAY allows users unlimited access (with the proper use of the unlock sequence) to the Peripheral Pin Select registers.

TABLE 11-5: REMAPPABLE PIN EXCEPTIONS FOR PIC24FJ128GB204 FAMILY DEVICES

Device		RPn Pins (I/O)	RPIn Pins		
Device	Total	Unimplemented	Total	Unimplemented	
PIC24FJXXXGB202	14	RP4, RP12	1	_	
PIC24FJXXXGB204	24	RP4, RP12	1	_	

11.4.5 CONSIDERATIONS FOR PERIPHERAL PIN SELECTION

The ability to control Peripheral Pin Selection introduces several considerations into application design that could be overlooked. This is particularly true for several common peripherals that are available only as remappable peripherals.

The main consideration is that the Peripheral Pin Selects are not available on default pins in the device's default (Reset) state. Since all RPINRx registers reset to '1111111' and all RPORx registers reset to '000000', all Peripheral Pin Select inputs are tied to Vss and all Peripheral Pin Select outputs are disconnected.

This situation requires the user to initialize the device with the proper peripheral configuration before any other application code is executed. Since the IOLOCK bit resets in the unlocked state, it is not necessary to execute the unlock sequence after the device has come out of Reset. For application safety, however, it is best to set IOLOCK and lock the configuration after writing to the control registers.

Because the unlock sequence is timing-critical, it must be executed as an assembly language routine in the same manner as changes to the oscillator configuration. If the bulk of the application is written in 'C', or another high-level language, the unlock sequence should be performed by writing in-line assembly.

Choosing the configuration requires the review of all Peripheral Pin Selects and their pin assignments, especially those that will not be used in the application. In all cases, unused pin-selectable peripherals should be disabled completely. Unused peripherals should have their inputs assigned to an unused RPn/RPIn pin function. I/O pins with unused RPn functions should be configured with the null peripheral output.

The assignment of a peripheral to a particular pin does not automatically perform any other configuration of the pin's I/O circuitry. In theory, this means adding a pin-selectable output to a pin may mean inadvertently driving an existing peripheral input when the output is driven. Users must be familiar with the behavior of other fixed peripherals that share a remappable pin and know when to enable or disable them. To be safe, fixed digital peripherals that share the same pin should be disabled when not in use.

Along these lines, configuring a remappable pin for a specific peripheral does not automatically turn that feature on. The peripheral must be specifically configured for operation and enabled as if it were tied to a fixed pin. Where this happens in the application code (immediately following a device Reset and peripheral configuration or inside the main application routine) depends on the peripheral and its use in the application.

A final consideration is that Peripheral Pin Select functions neither override analog inputs nor reconfigure pins with analog functions for digital I/O. If a pin is configured as an analog input on device Reset, it must be explicitly reconfigured as a digital I/O when used with a Peripheral Pin Select.

Example 11-3 shows a configuration for bidirectional communication with flow control using UART1. The following input and output functions are used:

Input Functions: U1RX, U1CTS
 Output Functions: U1TX, U1RTS

EXAMPLE 11-3: CONFIGURING UART1 INPUT AND OUTPUT FUNCTIONS

```
// Unlock Registers
asm volatile ("MOV
                       #OSCCON, w1
                                      \n"
                "MOV
                       #0x46, w2
                                      \n"
                "MOV
                                      \n"
                       #0x57, w3
                "MOV.b w2, [w1]
                                     \n"
                "MOV.b w3, [w1]
                                      \n"
                "BCLR OSCCON, #6")
// or use C30 built-in macro:
// __builtin_write_OSCCONL(OSCCON & 0xbf);
// Configure Input Functions (Table 11-3)
   // Assign U1RX To Pin RP0
   RPINR18bits.U1RXR = 0;
   // Assign U1CTS To Pin RP1
   RPINR18bits.U1CTSR = 1;
// Configure Output Functions (Table 11-4)
   // Assign U1TX To Pin RP2
   RPOR1bits.RP2R = 3;
   // Assign U1RTS To Pin RP3
   RPOR1bits.RP3R = 4;
// Lock Registers
asm volatile
              ("MOV #OSCCON, w1
                                      \n"
                                      \n"
               VOM"
                     #0x46, w2
                      #0x57, w3
               "MOV
                                      \n"
               "MOV.b w2, [w1]
                                      \n"
               "MOV.b w3, [w1]
                                      \n"
               "BSET OSCCON, #6")
// or use C30 built-in macro:
// builtin write OSCCONL(OSCCON | 0x40);
```

11.4.6 PERIPHERAL PIN SELECT REGISTERS

The PIC24FJ128GB204 family of devices implements a total of 32 registers for remappable peripheral configuration:

Note: Input and Output register values can only be changed if IOLOCK (OSCCON<6>) = 0.

See Section 11.4.4.1 "Control Register Lock" for a specific command sequence.

- Input Remappable Peripheral Registers (19)
- Output Remappable Peripheral Registers (13)

REGISTER 11-4: RPINR0: PERIPHERAL PIN SELECT INPUT REGISTER 0

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	INT1R5	INT1R4	INT1R3	INT1R2	INT1R1	INT1R0
bit 15							bit 8

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_		OCTRIG1R5	OCTRIG1R4	OCTRIG1R3	OCTRIG1R2	OCTRIG1R1	OCTRIG1R0
bit 7							bit 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 15-14 Unimplemented: Read as '0'

bit 13-8 INT1R<5:0>: Assign External Interrupt 1 (INT1) to Corresponding RPn or RPIn Pin bits

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

bit 5-0 OCTRIG1R<5:0>: Assign Output Compare Trigger 1 to Corresponding RPn or RPIn Pin bits

REGISTER 11-5: RPINR1: PERIPHERAL PIN SELECT INPUT REGISTER 1

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	INT3R5	INT3R4	INT3R3	INT3R2	INT3R1	INT3R0
bit 15							bit 8

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	INT2R5	INT2R4	INT2R3	INT2R2	INT2R1	INT2R0
bit 7							bit 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 15-14 Unimplemented: Read as '0'

bit 13-8 INT3R<5:0>: Assign External Interrupt 3 (INT3) to Corresponding RPn or RPIn Pin bits

bit 7-6 Unimplemented: Read as '0'

bit 5-0 INT2R<5:0>: Assign External Interrupt 2 (INT2) to Corresponding RPn or RPIn Pin bits

REGISTER 11-6: RPINR2: PERIPHERAL PIN SELECT INPUT REGISTER 2

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	OCTRIG2R5	OCTRIG2R4	OCTRIG2R3	OCTRIG2R2	OCTRIG2R1	OCTRIG2R0
bit 15							bit 8

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	INT4R5	INT4R4	INT4R3	INT4R2	INT4R1	INT4R0
bit 7							bit 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 15-14 **Unimplemented:** Read as '0'

bit 13-8 OCTRIG2R<5:0>: Assign Output Compare Trigger 2 to Corresponding RPn or RPIn Pin bits

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

bit 5-0 INT4R<5:0>: Assign External Interrupt 4 (INT4) to Corresponding RPn or RPIn Pin bits

REGISTER 11-7: RPINR7: PERIPHERAL PIN SELECT INPUT REGISTER 7

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	IC2R5	IC2R4	IC2R3	IC2R2	IC2R1	IC2R0
bit 15							bit 8

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	IC1R5	IC1R4	IC1R3	IC1R2	IC1R1	IC1R0
bit 7							bit 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 15-14 **Unimplemented:** Read as '0'

bit 13-8 IC2R<5:0>: Assign Input Capture 2 (IC2) to Corresponding RPn or RPIn Pin bits

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

bit 5-0 IC1R<5:0>: Assign Input Capture 1 (IC1) to Corresponding RPn or RPIn Pin bits

REGISTER 11-8: RPINR8: PERIPHERAL PIN SELECT INPUT REGISTER 8

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	IC4R5	IC4R4	IC4R3	IC4R2	IC4R1	IC4R0
bit 15							bit 8

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	IC3R5	IC3R4	IC3R3	IC3R2	IC3R1	IC3R0
bit 7							bit 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 15-14 Unimplemented: Read as '0'

bit 13-8 IC4R<5:0>: Assign Input Capture 4 (IC4) to Corresponding RPn or RPIn Pin bits

bit 7-6 Unimplemented: Read as '0'

bit 5-0 IC3R<5:0>: Assign Input Capture 3 (IC3) to Corresponding RPn or RPIn Pin bits

REGISTER 11-9: RPINR9: PERIPHERAL PIN SELECT INPUT REGISTER 9

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	IC6R5	IC6R4	IC6R3	IC6R2	IC6R1	IC6R0
bit 15							bit 8

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	IC5R5	IC5R4	IC5R3	IC5R2	IC5R1	IC5R0
bit 7							bit 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 15-14 Unimplemented: Read as '0'

bit 13-8 IC6R<5:0>: Assign Input Capture 6 (IC6) to Corresponding RPn or RPIn Pin bits

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

bit 5-0 IC5R<5:0>: Assign Input Capture 5 (IC5) to Corresponding RPn or RPIn Pin bits

REGISTER 11-10: RPINR11: PERIPHERAL PIN SELECT INPUT REGISTER 11

	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
	_	_	OCFBR5	OCFBR4	OCFBR3	OCFBR2	OCFBR1	OCFBR0
Ī	bit 15							bit 8

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	OCFAR5	OCFAR4	OCFAR3	OCFAR2	OCFAR1	OCFAR0
bit 7							bit 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 15-14 Unimplemented: Read as '0'

bit 13-8 OCFBR<5:0>: Assign Output Compare Fault B (OCFB) to Corresponding RPn or RPIn Pin bits

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

bit 5-0 OCFAR<5:0>: Assign Output Compare Fault A (OCFA) to Corresponding RPn or RPIn Pin bits

REGISTER 11-11: RPINR17: PERIPHERAL PIN SELECT INPUT REGISTER 17

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_			U3RXI	R<5:0>		
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 7							bit 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 15-14 **Unimplemented:** Read as '0'

bit 13-8 U3RXR<5:0>: Assign UART3 Receive (U3RX) to Corresponding RPn or RPIn Pin bits

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

REGISTER 11-12: RPINR18: PERIPHERAL PIN SELECT INPUT REGISTER 18

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	U1CTSR5	U1CTSR4	U1CTSR3	U1CTSR2	U1CTSR1	U1CTSR0
bit 15							bit 8

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	U1RXR5	U1RXR4	U1RXR3	U1RXR2	U1RXR1	U1RXR0
bit 7							bit 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 15-14 Unimplemented: Read as '0'

bit 13-8 **U1CTSR<5:0>:** Assign UART1 Clear-to-Send (U1CTS) to Corresponding RPn or RPIn Pin bits

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

bit 5-0 U1RXR<5:0>: Assign UART1 Receive (U1RX) to Corresponding RPn or RPIn Pin bits

REGISTER 11-13: RPINR19: PERIPHERAL PIN SELECT INPUT REGISTER 19

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	U2CTSR5	U2CTSR4	U2CTSR3	U2CTSR2	U2CTSR1	U2CTSR0
bit 15							bit 8

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	U2RXR5	U2RXR4	U2RXR3	U2RXR2	U2RXR1	U2RXR0
bit 7							bit 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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **U2CTSR<5:0>:** Assign UART2 Clear-to-Send (U2CTS) to Corresponding RPn or RPIn Pin bits

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

bit 5-0 **U2RXR<5:0>:** Assign UART2 Receive (U2RX) to Corresponding RPn or RPIn Pin bits

REGISTER 11-14: RPINR20: PERIPHERAL PIN SELECT INPUT REGISTER 20

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	SCK1R5	SCK1R4	SCK1R3	SCK1R2	SCK1R1	SCK1R0
bit 15							bit 8

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	SDI1R5	SDI1R4	SDI1R3	SDI1R2	SDI1R1	SDI1R0
bit 7							bit 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 15-14 **Unimplemented:** Read as '0'

bit 13-8 SCK1R<5:0>: Assign SPI1 Clock Input (SCK1IN) to Corresponding RPn or RPIn Pin bits

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

bit 5-0 SDI1R<5:0>: Assign SPI1 Data Input (SDI1) to Corresponding RPn or RPIn Pin bits

REGISTER 11-15: RPINR21: PERIPHERAL PIN SELECT INPUT REGISTER 21

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	U3CTSR5	U3CTSR4	U3CTSR3	U3CTSR2	U3CTSR1	U3CTSR0
bit 15							bit 8

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	SS1R5	SS1R4	SS1R3	SS1R2	SS1R1	SS1R0
bit 7							bit 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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **U3CTSR<5:0>:** Assign UART3 Clear-to-Send (U3CTS) to Corresponding RPn or RPIn Pin bits

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

bit 5-0 SS1R<5:0>: Assign SPI1 Slave Select Input (SS1IN) to Corresponding RPn or RPIn Pin bits

REGISTER 11-16: RPINR22: PERIPHERAL PIN SELECT INPUT REGISTER 22

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	SCK2R5	SCK2R4	SCK2R3	SCK2R2	SCK2R1	SCK2R0
bit 15							bit 8

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	SDI2R5	SDI2R4	SDI2R3	SDI2R2	SDI2R1	SDI2R0
bit 7							bit 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 15-14 Unimplemented: Read as '0'

bit 13-8 SCK2R<5:0>: Assign SPI2 Clock Input (SCK2IN) to Corresponding RPn or RPIn Pin bits

bit 7-6 Unimplemented: Read as '0'

bit 5-0 SDI2R<5:0>: Assign SPI2 Data Input (SDI2) to Corresponding RPn or RPIn Pin bits

REGISTER 11-17: RPINR23: PERIPHERAL PIN SELECT INPUT REGISTER 23

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	TMRCKR5	TMRCKR4	TMRCKR3	TMRCKR2	TMRCKR1	TMRCKR0
bit 15							bit 8

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	SS2R5	SS2R4	SS2R3	SS2R2	SS2R1	SS2R0
bit 7							bit 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 15-14 Unimplemented: Read as '0'

bit 13-8 TMRCKR<5:0>: Assign General Timer External Input (TMRCK) to Corresponding RPn or RPIn Pin bits

bit 7-6 Unimplemented: Read as '0'

bit 5-0 SS2R<5:0>: Assign SPI2 Slave Select Input (SS2IN) to Corresponding RPn or RPIn Pin bits

REGISTER 11-18: RPINR27: PERIPHERAL PIN SELECT INPUT REGISTER 27

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	U4CTSR5	U4CTSR4	U4CTSR3	U4CTSR2	U4CTSR1	U4CTSR0
bit 15							bit 8

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	U4RXR5	U4RXR4	U4RXR3	U4RXR2	U4RXR1	U4RXR0
bit 7							bit 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 15-14 Unimplemented: Read as '0'

bit 13-8 **U4CTSR<5:0>:** Assign UART4 Clear-to-Send Input (U4CTS) to Corresponding RPn or RPIn Pin bits

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

bit 5-0 U4RXR<5:0>: Assign UART4 Receive Input (U4RX) to Corresponding RPn or RPIn Pin bits

REGISTER 11-19: RPINR28: PERIPHERAL PIN SELECT INPUT REGISTER 28

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	SCK3R5	SCK3R4	SCK3R3	SCK3R2	SCK3R1	SCK3R0
bit 15							bit 8

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	SDI3R5	SDI3R4	SDI3R3	SDI3R2	SDI3R1	SDI3R0
bit 7							bit 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 15-14 **Unimplemented:** Read as '0'

bit 13-8 SCK3R<5:0>: Assign SPI3 Clock Input (SCK3IN) to Corresponding RPn or RPIn Pin bits

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

bit 5-0 SDI3R<5:0>: Assign SPI3 Data Input (SDI3) to Corresponding RPn or RPIn Pin bits

REGISTER 11-20: RPINR29: PERIPHERAL PIN SELECT INPUT REGISTER 29

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_			SS3R	R<5:0>		
bit 7							bit 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 15-6 Unimplemented: Read as '0'

bit 5-0 SS3R<5:0>: Assign SPI3 Slave Select Input (SS3IN) to Corresponding RPn or RPIn Pin bits

REGISTER 11-21: RPINR30: PERIPHERAL PIN SELECT INPUT REGISTER 30

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_			MDMIF	R<5:0>		
bit 7							bit 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 15-6 **Unimplemented:** Read as '0'

bit 5-0 MDMIR<5:0>: Assign TX Modulation Input (MDMI) to Corresponding RPn or RPIn Pin bits

REGISTER 11-22: RPINR31: PERIPHERAL PIN SELECT INPUT REGISTER 31

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	MDC2R5	MDC2R4	MDC2R3	MDC2R2	MDC2R1	MDC2R0
bit 15							bit 8

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	MDC1R5	MDC1R4	MDC1R3	MDC1R2	MDC21R1	MDC1R0
bit 7							bit 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 15-14 **Unimplemented:** Read as '0'

bit 13-8 MDC2R<5:0>: Assign TX Carrier 2 Input (MDCIN2) to Corresponding RPn or RPIn Pin bits

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

bit 5-0 MDC1R<5:0>: Assign TX Carrier 1 Input (MDCIN1) to Corresponding RPn or RPIn Pin bits

REGISTER 11-23: RPOR0: PERIPHERAL PIN SELECT OUTPUT REGISTER 0

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	RP1R5	RP1R4	RP1R3	RP1R2	RP1R1	RP1R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	RP0R5	RP0R4	RP0R3	RP0R2	RP0R1	RP0R0
bit 7							bit 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 15-14 Unimplemented: Read as '0'

bit 13-8 **RP1R<5:0>:** RP1 Output Pin Mapping bits

Peripheral Output Number n is assigned to pin, RP1 (see Table 11-4 for peripheral function numbers).

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

bit 5-0 **RP0R<5:0>:** RP0 Output Pin Mapping bits

Peripheral Output Number n is assigned to pin, RP0 (see Table 11-4 for peripheral function numbers).

REGISTER 11-24: RPOR1: PERIPHERAL PIN SELECT OUTPUT REGISTER 1

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	RP3R5	RP3R4	RP3R3	RP3R2	RP3R1	RP3R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	RP2R5	RP2R4	RP2R3	RP2R2	RP2R1	RP2R0
bit 7							bit 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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **RP3R<5:0>:** RP3 Output Pin Mapping bits

Peripheral Output Number n is assigned to pin, RP3 (see Table 11-4 for peripheral function numbers).

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

bit 5-0 RP2R<5:0>: RP2 Output Pin Mapping bits

Peripheral Output Number n is assigned to pin, RP2 (see Table 11-4 for peripheral function numbers).

REGISTER 11-25: RPOR2: PERIPHERAL PIN SELECT OUTPUT REGISTER 2

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_			RP5R	R<5:0>		
bit 15	-				_		bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 7							bit 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 15-14 Unimplemented: Read as '0'

bit 13-8 RP5R<5:0>: RP5 Output Pin Mapping bits

Peripheral Output Number n is assigned to pin, RP5 (see Table 11-4 for peripheral function numbers).

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

REGISTER 11-26: RPOR3: PERIPHERAL PIN SELECT OUTPUT REGISTER 3

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	RP7R5	RP7R4	RP7R3	RP7R2	RP7R1	RP7R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	RP6R5	RP6R4	RP6R3	RP6R2	RP6R1	RP6R0
bit 7							bit 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 15-14 **Unimplemented:** Read as '0'

bit 13-8 RP7R<5:0>: RP7 Output Pin Mapping bits

Peripheral Output Number n is assigned to pin, RP7 (see Table 11-4 for peripheral function numbers).

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

bit 5-0 **RP6R<5:0>:** RP6 Output Pin Mapping bits

Peripheral Output Number n is assigned to pin, RP6 (see Table 11-4 for peripheral function numbers).

REGISTER 11-27: RPOR4: PERIPHERAL PIN SELECT OUTPUT REGISTER 4

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	RP9R5	RP9R4	RP9R3	RP9R2	RP9R1	RP9R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	RP8R5	RP8R4	RP8R3	RP8R2	RP8R1	RP8R0
bit 7							bit 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 15-14 Unimplemented: Read as '0'

bit 13-8 **RP9R<5:0>:** RP9 Output Pin Mapping bits

Peripheral Output Number n is assigned to pin, RP9 (see Table 11-4 for peripheral function numbers).

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

bit 5-0 RP8R<5:0>: RP8 Output Pin Mapping bits

Peripheral Output Number n is assigned to pin, RP8 (see Table 11-4 for peripheral function numbers).

REGISTER 11-28: RPOR5: PERIPHERAL PIN SELECT OUTPUT REGISTER 5

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	RP11R5	RP11R4	RP11R3	RP11R2	RP11R1	RP11R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	RP10R5	RP10R4	RP10R3	RP10R2	RP10R1	RP10R0
bit 7							bit 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 15-14 Unimplemented: Read as '0'

bit 13-8 **RP11R<5:0>:** RP11 Output Pin Mapping bits

Peripheral Output Number n is assigned to pin, RP11 (see Table 11-4 for peripheral function numbers).

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

bit 5-0 RP10R<5:0>: RP10 Output Pin Mapping bits

Peripheral Output Number n is assigned to pin, RP10 (see Table 11-4 for peripheral function numbers).

REGISTER 11-29: RPOR6: PERIPHERAL PIN SELECT OUTPUT REGISTER 6

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_			RP13F	R<5:0>		
bit 15	_	_					bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 7							bit 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 15-14 **Unimplemented:** Read as '0'

bit 13-8 RP13R<5:0>: RP13 Output Pin Mapping bits

Peripheral Output Number n is assigned to pin, RP13 (see Table 11-4 for peripheral function numbers).

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

REGISTER 11-30: RPOR7: PERIPHERAL PIN SELECT OUTPUT REGISTER 7

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	RP15R5	RP15R4	RP15R3	RP15R2	RP15R1	RP15R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	RP14R5	RP14R4	RP14R3	RP14R2	RP14R1	RP14R0
bit 7							bit 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 15-14 Unimplemented: Read as '0'

bit 13-8 RP15R<5:0>: RP15 Output Pin Mapping bits

Peripheral Output Number n is assigned to pin, RP15 (see Table 11-4 for peripheral function numbers).

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

bit 5-0 RP14R<5:0>: RP14 Output Pin Mapping bits

Peripheral Output Number n is assigned to pin, RP14 (see Table 11-4 for peripheral function numbers).

REGISTER 11-31: RPOR8: PERIPHERAL PIN SELECT OUTPUT REGISTER 8⁽¹⁾

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	RP17R5	RP17R4	RP17R3	RP17R2	RP17R1	RP17R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	RP16R5	RP16R4	RP16R3	RP16R2	RP16R1	RP16R0
bit 7							bit 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 15-14 Unimplemented: Read as '0'

bit 13-8 **RP17R<5:0>:** RP17 Output Pin Mapping bits

Peripheral Output Number n is assigned to pin, RP17 (see Table 11-4 for peripheral function numbers).

bit 7-6 Unimplemented: Read as '0'

bit 5-0 **RP16R<5:0>:** RP16 Output Pin Mapping bits

Peripheral Output Number n is assigned to pin, RP16 (see Table 11-4 for peripheral function numbers).

Note 1: These pins are not available in 28-pin devices.

REGISTER 11-32: RPOR9: PERIPHERAL PIN SELECT OUTPUT REGISTER 9⁽¹⁾

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	RP19R5	RP19R4	RP19R3	RP19R2	RP19R1	RP19R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	RP18R5	RP18R4	RP18R3	RP18R2	RP18R1	RP18R0
bit 7							bit 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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **RP19R<5:0>:** RP19 Output Pin Mapping bits

Peripheral Output Number n is assigned to pin, RP19 (see Table 11-4 for peripheral function numbers).

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

bit 5-0 RP18R<5:0>: RP18 Output Pin Mapping bits

Peripheral Output Number n is assigned to pin, RP18 (see Table 11-4 for peripheral function numbers).

Note 1: These pins are not available in 28-pin devices.

REGISTER 11-33: RPOR10: PERIPHERAL PIN SELECT OUTPUT REGISTER 10⁽¹⁾

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	RP21R5	RP21R4	RP21R3	RP21R2	RP21R1	RP21R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	RP20R5	RP20R4	RP20R3	RP20R2	RP20R1	RP20R0
bit 7							bit 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 15-14 **Unimplemented:** Read as '0'

bit 13-8 RP21R<5:0>: RP21 Output Pin Mapping bits

Peripheral Output Number n is assigned to pin, RP21 (see Table 11-4 for peripheral function numbers).

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

bit 5-0 RP20R<5:0>: RP20 Output Pin Mapping bits

Peripheral Output Number n is assigned to pin, RP20 (see Table 11-4 for peripheral function numbers).

Note 1: These pins are not available in 28-pin devices.

REGISTER 11-34: RPOR11: PERIPHERAL PIN SELECT OUTPUT REGISTER 11(1)

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	RP23R5	RP23R4	RP23R3	RP23R2	RP23R1	RP23R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	RP22R5	RP22R4	RP22R3	RP22R2	RP22R1	RP22R0
bit 7							bit 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 15-14 Unimplemented: Read as '0'

bit 13-8 **RP23R<5:0>:** RP23 Output Pin Mapping bits

Peripheral Output Number n is assigned to pin, RP23 (see Table 11-4 for peripheral function numbers).

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

bit 5-0 RP22R<5:0>: RP22 Output Pin Mapping bits

Peripheral Output Number n is assigned to pin, RP22 (see Table 11-4 for peripheral function numbers).

Note 1: These pins are not available in 28-pin devices.

REGISTER 11-35: RPOR12: PERIPHERAL PIN SELECT OUTPUT REGISTER 12⁽¹⁾

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	RP25R5	RP25R4	RP25R3	RP25R2	RP25R1	RP25R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	RP24R5	RP24R4	RP24R3	RP24R2	RP24R1	RP24R0
bit 7							bit 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 15-14 Unimplemented: Read as '0'

bit 13-8 **RP25R<5:0>:** RP25 Output Pin Mapping bits

Peripheral Output Number n is assigned to pin, RP25 (see Table 11-4 for peripheral function numbers).

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

bit 5-0 RP24R<5:0>: RP24 Output Pin Mapping bits

Peripheral Output Number n is assigned to pin, RP24 (see Table 11-4 for peripheral function numbers).

Note 1: These pins are not available in 28-pin devices.

NOTES:	
--------	--

12.0 TIMER1

Note:

This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the "dsPIC33/PIC24 Family Reference Manual", "Timers" (DS39704). The information in this data sheet supersedes the information in the FRM.

The Timer1 module is a 16-bit timer, which can serve as the time counter for the Real-Time Clock (RTC) or operate as a free-running, interval timer/counter. Timer1 can operate in three modes:

- · 16-Bit Timer
- 16-Bit Synchronous Counter
- 16-Bit Asynchronous Counter

Timer1 also supports these features:

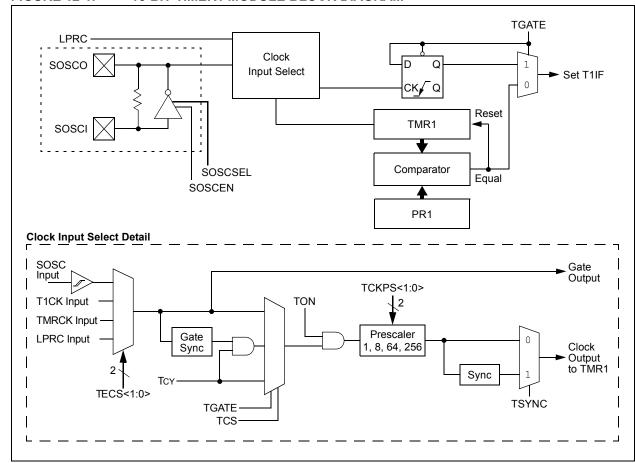
- · Timer Gate Operation
- · Selectable Prescaler Settings
- · Timer Operation during CPU Idle and Sleep modes
- Interrupt on 16-Bit Period Register Match or Falling Edge of External Gate Signal

Figure 12-1 shows a block diagram of the 16-bit timer module.

To configure Timer1 for operation:

- Set the TON bit (= 1).
- Select the timer prescaler ratio using the TCKPS<1:0> bits.
- 3. Set the Clock and Gating modes using the TCS, TECS<1:0> and TGATE bits.
- 4. Set or clear the TSYNC bit to configure synchronous or asynchronous operation.
- Load the timer period value into the PR1 register.
- 6. If interrupts are required, set the Timer1 Interrupt Enable bit, T1IE. Use the Timer1 Interrupt Priority bits, T1IP<2:0>, to set the interrupt priority.

FIGURE 12-1: 16-BIT TIMER1 MODULE BLOCK DIAGRAM



REGISTER 12-1: T1CON: TIMER1 CONTROL REGISTER⁽¹⁾

R/W-0	U-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0
TON	_	TSIDL	_	_	_	TECS1	TECS0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	U-0
_	TGATE	TCKPS1	TCKPS0	_	TSYNC	TCS	_
bit 7							bit 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 15 TON: Timer1 On bit

1 = Starts 16-bit Timer1

0 = Stops 16-bit Timer1

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

bit 13 TSIDL: Timer1 Stop in Idle Mode bit

1 = Discontinues module operation when device enters Idle mode

0 = Continues module operation in Idle mode

bit 12-10 Unimplemented: Read as '0'

bit 9-8 **TECS<1:0>:** Timer1 Extended Clock Source Select bits (selected when TCS = 1)

When TCS = 1:

11 = Generic Timer (TMRCK) external input

10 = LPRC Oscillator

01 = T1CK external clock input

00 = SOSC

When TCS = 0:

These bits are ignored; Timer1 is clocked from the internal system clock (Fosc/2).

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

bit 6 TGATE: Timer1 Gated Time Accumulation Enable bit

When TCS = 1: This bit is ignored.

When TCS = 0:

1 = Gated time accumulation is enabled0 = Gated time accumulation is disabled

bit 5-4 TCKPS<1:0>: Timer1 Input Clock Prescale Select bits

11 = 1:256

10 = 1:64

01 = 1:8

00 = 1:1

bit 3 Unimplemented: Read as '0'

Note 1: Changing the value of T1CON while the timer is running (TON = 1) causes the timer prescale counter to reset and is not recommended.

REGISTER 12-1: T1CON: TIMER1 CONTROL REGISTER⁽¹⁾ (CONTINUED)

bit 2 TSYNC: Timer1 External Clock Input Synchronization Select bit

When TCS = 1:

1 = Synchronizes external clock input

0 = Does not synchronize external clock input

 $\frac{\text{When TCS = } 0:}{\text{This bit is ignored.}}$

bit 1 TCS: Timer1 Clock Source Select bit

1 = Extended clock selected by the TECS<1:0> bits

0 = Internal clock (Fosc/2)

bit 0 Unimplemented: Read as '0'

Note 1: Changing the value of T1CON while the timer is running (TON = 1) causes the timer prescale counter to reset and is not recommended.

PIC24FJ128GB204 FAMILY							
OTES:							

13.0 TIMER2/3 AND TIMER4/5

Note:

This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the "dsPIC33/PIC24 Family Reference Manual", "Timers" (DS39704). The information in this data sheet supersedes the information in the FRM.

The Timer2/3 and Timer4/5 modules are 32-bit timers, which can also be configured as four independent, 16-bit timers with selectable operating modes.

As 32-bit timers, Timer2/3 and Timer4/5 can each operate in three modes:

- Two Independent 16-Bit Timers with all 16-Bit Operating modes (except Asynchronous Counter mode)
- · Single 32-Bit Timer
- · Single 32-Bit Synchronous Counter

They also support these features:

- · Timer Gate Operation
- Selectable Prescaler Settings
- · Timer Operation during Idle and Sleep modes
- · Interrupt on a 32-Bit Period Register Match
- A/D Event Trigger (only on Timer2/3 in 32-bit mode and Timer3 in 16-bit mode)

Individually, all four of the 16-bit timers can function as synchronous timers or counters. They also offer the features listed above, except for the A/D Event Trigger. This trigger is implemented only on Timer2/3 in 32-bit mode and Timer3 in 16-bit mode. The operating modes and enabled features are determined by setting the appropriate bit(s) in the T2CON, T3CON, T4CON and T5CON registers. T2CON and T4CON are shown in Register 13-1; T3CON and T5CON are shown in Register 13-2.

For 32-bit timer/counter operation, Timer2 and Timer4 are the least significant word; Timer3 and Timer5 are the most significant word of the 32-bit timers.

Note

For 32-bit operation, T3CON and T5CON control bits are ignored. Only T2CON and T4CON control bits are used for setup and control. Timer2 and Timer4 clock and gate inputs are utilized for the 32-bit timer modules, but an interrupt is generated with the Timer3 or Timer5 interrupt flags.

To configure Timer2/3 or Timer4/5 for 32-bit operation:

- 1. Set the T32 or T45 bit (T2CON<3> or T4CON<3> = 1).
- Select the prescaler ratio for Timer2 or Timer4 using the TCKPS<1:0> bits.
- Set the Clock and Gating modes using the TCS and TGATE bits. If TCS is set to an external clock, RPINRx (TxCK) must be configured to an available RPn/RPIn pin. For more information, see Section 11.4 "Peripheral Pin Select (PPS)".
- 4. Load the timer period value. PR3 (or PR5) will contain the most significant word (msw) of the value, while PR2 (or PR4) contains the least significant word (lsw).
- If interrupts are required, set the Timer3/5 Interrupt Enable bit, T3IE or T5IE. Use the Timer3/5 Interrupt Priority bits, T3IP<2:0> or T5IP<2:0>, to set the interrupt priority. Note that while Timer2 or Timer4 controls the timer, the interrupt appears as a Timer3 or Timer5 interrupt.
- 6. Set the TON bit (= 1).

The timer value, at any point, is stored in the register pair, TMR<3:2> (or TMR<5:4>). TMR3 (TMR5) always contains the most significant word of the count, while TMR2 (TMR4) contains the least significant word.

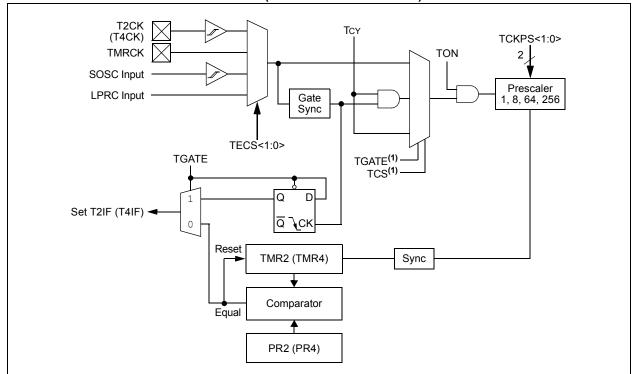
To configure any of the timers for individual 16-bit operation:

- Clear the T32 bit corresponding to that timer (T2CON<3> for Timer2 and Timer3 or T4CON<3> for Timer4 and Timer5).
- Select the timer prescaler ratio using the TCKPS<1:0> bits.
- 3. Set the Clock and Gating modes using the TCS and TGATE bits. See **Section 11.4 "Peripheral Pin Select (PPS)"** for more information.
- 4. Load the timer period value into the PRx register.
- 5. If interrupts are required, set the Timerx Interrupt Enable bit, TxIE. Use the Timerx Interrupt Priority bits, TxIP<2:0>, to set the interrupt priority.
- 6. Set the TON (TxCON<15> = 1) bit.

TCY TCKPS<1:0> **TMRCK** 2 SOSC Input Prescaler LPRC Input Gate 1, 8, 64, 256 Sync TECS<1:0> TGATE(2) **TGATE** TCS⁽²⁾ D 1 Q Set T3IF (T5IF) 0 Q PR3 (PR5) PR2 (PR4) Equal Comparator A/D Event Trigger(3) MSB LSB TMR3 TMR2 Sync (TMR4) Reset (TMR5) Read TMR2 (TMR4)(1) Write TMR2 (TMR4)(1) 16 TMR3HLD 16 (TMR5HLD) Data Bus<15:0> < Note 1: The 32-Bit Timer Configuration bit, T32, must be set for 32-bit timer/counter operation. All control bits are respective to the T2CON and T4CON registers. The timer clock input must be assigned to an available RPn/RPIn pin before use. See Section 11.4 "Peripheral Pin Select (PPS)" for more information. 3: The A/D Event Trigger is available only on Timer2/3 in 32-bit mode and Timer3 in 16-bit mode.

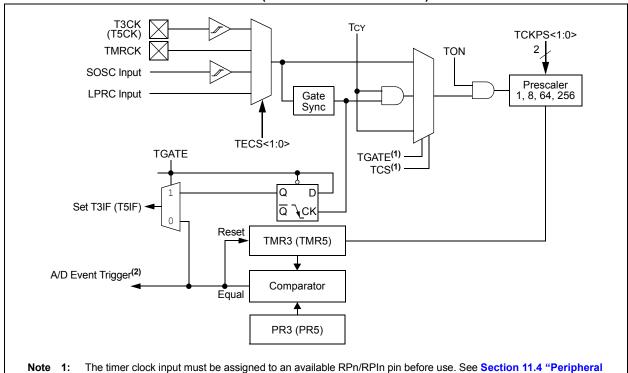
FIGURE 13-1: TIMER2/3 AND TIMER4/5 (32-BIT) BLOCK DIAGRAM

FIGURE 13-2: TIMER2 AND TIMER4 (16-BIT SYNCHRONOUS) BLOCK DIAGRAM



Note 1: The timer clock input must be assigned to an available RPn/RPIn pin before use. See Section 11.4 "Peripheral Pin Select (PPS)" for more information.

FIGURE 13-3: TIMER3 AND TIMER5 (16-BIT ASYNCHRONOUS) BLOCK DIAGRAM



2: The A/D Event Trigger is available only on Timer3.

REGISTER 13-1: TxCON: TIMER2 AND TIMER4 CONTROL REGISTER⁽¹⁾

R/W-0	U-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0
TON	_	TSIDL	_	_	_	TECS1 ⁽²⁾	TECS0 ⁽²⁾
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	U-0
_	TGATE	TCKPS1	TCKPS0	T32 ⁽³⁾	_	TCS ⁽²⁾	_
bit 7							bit 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 15 **TON:** Timerx On bit

When TxCON<3> = 1:

1 = Starts 32-bit Timerx/y

0 = Stops 32-bit Timerx/y

When TxCON<3> = 0:

1 = Starts 16-bit Timerx

0 = Stops 16-bit Timerx

bit 14 Unimplemented: Read as '0'

bit 13 TSIDL: Timerx Stop in Idle Mode bit

1 = Discontinues module operation when device enters Idle mode

0 = Continues module operation in Idle mode

bit 12-10 Unimplemented: Read as '0'

bit 9-8 **TECS<1:0>:** Timerx Extended Clock Source Select bits (selected when TCS = 1)(2)

When TCS = 1:

11 = Generic Timer (TMRCK) external input

10 = LPRC Oscillator

01 = TxCK external clock input

00 = SOSC

When TCS = 0

These bits are ignored; Timerx is clocked from the internal system clock (Fosc/2).

bit 7 Unimplemented: Read as '0'

bit 6 TGATE: Timerx Gated Time Accumulation Enable bit

When TCS = 1:

This bit is ignored.

When TCS = 0:

1 = Gated time accumulation is enabled

0 = Gated time accumulation is disabled

bit 5-4 TCKPS<1:0>: Timerx Input Clock Prescale Select bits

11 = 1:256

10 = 1:64

01 = 1:8

00 = 1:1

- **Note 1:** Changing the value of TxCON while the timer is running (TON = 1) causes the timer prescale counter to reset and is not recommended.
 - 2: If TCS = 1 and TECS<1:0> = x1, the selected external timer input (TMRCK or TxCK) must be configured to an available RPn/RPIn pin. For more information, see Section 11.4 "Peripheral Pin Select (PPS)".
 - 3: In T4CON, the T45 bit is implemented instead of T32 to select 32-bit mode. In 32-bit mode, the T3CON or T5CON control bits do not affect 32-bit timer operation.

REGISTER 13-1: TxCON: TIMER2 AND TIMER4 CONTROL REGISTER⁽¹⁾ (CONTINUED)

bit 3 T32: 32-Bit Timer Mode Select bit⁽³⁾

1 = Timerx and Timery form a single 32-bit timer 0 = Timerx and Timery act as two 16-bit timers

In 32-bit mode, T3CON control bits do not affect 32-bit timer operation.

bit 2 Unimplemented: Read as '0'

bit 1 TCS: Timerx Clock Source Select bit⁽²⁾

1 = Timer source is selected by TECS<1:0>

0 = Internal clock (Fosc/2)

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

- **Note 1:** Changing the value of TxCON while the timer is running (TON = 1) causes the timer prescale counter to reset and is not recommended.
 - 2: If TCS = 1 and TECS<1:0> = x1, the selected external timer input (TMRCK or TxCK) must be configured to an available RPn/RPIn pin. For more information, see Section 11.4 "Peripheral Pin Select (PPS)".
 - **3:** In T4CON, the T45 bit is implemented instead of T32 to select 32-bit mode. In 32-bit mode, the T3CON or T5CON control bits do not affect 32-bit timer operation.

REGISTER 13-2: TyCON: TIMER3 AND TIMER5 CONTROL REGISTER⁽¹⁾

R/W-0	U-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0
TON ⁽²⁾	_	TSIDL ⁽²⁾	_	_	_	TECS1 ^(2,3)	TECS0 ^(2,3)
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	U-0
_	TGATE ⁽²⁾	TCKPS1 ⁽²⁾	TCKPS0 ⁽²⁾	_	_	TCS ^(2,3)	
bit 7							bit 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 15 **TON:** Timery On bit⁽²⁾

1 = Starts 16-bit Timery

0 = Stops 16-bit Timery

bit 14 Unimplemented: Read as '0'

bit 13 **TSIDL:** Timery Stop in Idle Mode bit⁽²⁾

1 = Discontinues module operation when device enters Idle mode

0 = Continues module operation in Idle mode

bit 12-10 Unimplemented: Read as '0'

bit 9-8 **TECS<1:0>:** Timery Extended Clock Source Select bits (selected when TCS = 1)(2,3)

11 = Generic Timer (TMRCK) external input

10 = LPRC Oscillator

01 = TxCK external clock input

00 = SOSC

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

bit 6 **TGATE:** Timery Gated Time Accumulation Enable bit⁽²⁾

When TCS = 1: This bit is ignored. When TCS = 0:

1 = Gated time accumulation is enabled0 = Gated time accumulation is disabled

bit 5-4 TCKPS<1:0>: Timery Input Clock Prescale Select bits(2)

11 = 1:256

10 = 1:64

01 = 1:8

00 = 1:1

bit 3-2 **Unimplemented:** Read as '0'

bit 1 TCS: Timery Clock Source Select bit^(2,3)

1 = External clock from pin, TyCK (on the rising edge)

0 = Internal clock (Fosc/2)

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

Note 1: Changing the value of TyCON while the timer is running (TON = 1) causes the timer prescale counter to reset and is not recommended.

- 2: When 32-bit operation is enabled (T2CON<3> or T4CON<3> = 1), these bits have no effect on Timery operation; all timer functions are set through T2CON and T4CON.
- 3: If TCS = 1 and TECS<1:0> = x1, the selected external timer input (TyCK) must be configured to an available RPn/RPIn pin. For more information, see Section 11.4 "Peripheral Pin Select (PPS)".

14.0 INPUT CAPTURE WITH DEDICATED TIMERS

Note:

This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the "dsPIC33/PIC24 Family Reference Manual", "Input Capture with Dedicated Timer" (DS39722). The information in this data sheet supersedes the information in the FRM.

Devices in the PIC24FJ128GB204 family contain six independent input capture modules. Each of the modules offers a wide range of configuration and operating options for capturing external pulse events and generating interrupts.

Key features of the input capture module include:

- Hardware-configurable for 32-bit operation in all modes by cascading two adjacent modules
- Synchronous and Trigger modes of output compare operation, with up to 30 user-selectable sync/trigger sources available
- A 4-level FIFO buffer for capturing and holding timer values for several events
- · Configurable interrupt generation
- Up to 6 clock sources available for each module, driving a separate, internal 16-bit counter

The module is controlled through two registers: ICxCON1 (Register 14-1) and ICxCON2 (Register 14-2). A general block diagram of the module is shown in Figure 14-1.

14.1 General Operating Modes

14.1.1 SYNCHRONOUS AND TRIGGER MODES

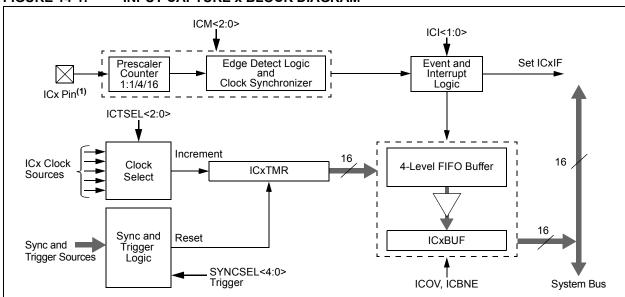
When the input capture module operates in a Free-Running mode, the internal 16-bit counter, ICxTMR, counts up continuously, wrapping around from FFFFh to 0000h on each overflow. Its period is synchronized to the selected external clock source. When a capture event occurs, the current 16-bit value of the internal counter is written to the FIFO buffer.

In Synchronous mode, the module begins capturing events on the ICx pin as soon as its selected clock source is enabled. Whenever an event occurs on the selected sync source, the internal counter is reset. In Trigger mode, the module waits for a sync event from another internal module to occur before allowing the internal counter to run.

Standard, free-running operation is selected by setting the SYNCSELx bits (ICxCON2<4:0>) to '00000' and clearing the ICTRIG bit (ICxCON2<7>). Synchronous and Trigger modes are selected any time the SYNCSELx bits are set to any value except '00000'. The ICTRIG bit selects either Synchronous or Trigger mode; setting the bit selects Trigger mode operation. In both modes, the SYNCSELx bits determine the sync/trigger source.

When the SYNCSELx bits are set to '00000' and ICTRIG is set, the module operates in Software Trigger mode. In this case, capture operations are started by manually setting the TRIGSTAT bit (ICxCON2<6>).

FIGURE 14-1: INPUT CAPTURE x BLOCK DIAGRAM



Note 1: The ICx inputs must be assigned to an available RPn/RPIn pin before use. See Section 11.4 "Peripheral Pin Select (PPS)" for more information.

14.1.2 CASCADED (32-BIT) MODE

By default, each module operates independently with its own 16-bit timer. To increase resolution, adjacent even and odd modules can be configured to function as a single 32-bit module. (For example, Modules 1 and 2 are paired, as are Modules 3 and 4, and so on.) The odd numbered module, Input Capture x (ICx), provides the Least Significant 16 bits of the 32-bit register pairs and the even numbered module, Input Capture y (ICy), provides the Most Significant 16 bits. Wrap arounds of the ICx registers cause an increment of their corresponding ICy registers.

Cascaded operation is configured in hardware by setting the IC32 bits (ICxCON2<8>) for both modules.

14.2 Capture Operations

The input capture module can be configured to capture timer values and generate interrupts on rising edges on ICx or all transitions on ICx. Captures can be configured to occur on all rising edges or just some (every 4th or 16th). Interrupts can be independently configured to generate on each event or a subset of events.

To set up the module for capture operations:

- Configure the ICx input for one of the available Peripheral Pin Select pins.
- If Synchronous mode is to be used, disable the sync source before proceeding.
- Make sure that any previous data has been removed from the FIFO by reading ICxBUF until the ICBNE bit (ICxCON1<3>) is cleared.
- Set the SYNCSELx bits (ICxCON2<4:0>) to the desired sync/trigger source.
- Set the ICTSELx bits (ICxCON1<12:10>) for the desired clock source.
- Set the ICIx bits (ICxCON1<6:5>) to the desired interrupt frequency
- 7. Select Synchronous or Trigger mode operation:
 - a) Check that the SYNCSELx bits are not set to '00000'.
 - b) For Synchronous mode, clear the ICTRIG bit (ICxCON2<7>).
 - c) For Trigger mode, set ICTRIG and clear the TRIGSTAT bit (ICxCON2<6>).
- 8. Set the ICMx bits (ICxCON1<2:0>) to the desired operational mode.
- 9. Enable the selected sync/trigger source.

For 32-bit cascaded operations, the setup procedure is slightly different:

- 1. Set the IC32 bits for both modules (ICyCON2<8> and ICxCON2<8>), enabling the even numbered module first. This ensures that the modules will start functioning in unison.
- Set the ICTSELx and SYNCSELx bits for both modules to select the same sync/trigger and time base source. Set the even module first, then the odd module. Both modules must use the same ICTSELx and SYNCSELx bit settings.
- Clear the ICTRIG bit of the even module (ICyCON2<7>). This forces the module to run in Synchronous mode with the odd module, regardless of its trigger setting.
- Use the odd module's ICIx bits (ICxCON1<6:5>) to set the desired interrupt frequency.
- Use the ICTRIG bit of the odd module (ICxCON2<7>) to configure Trigger or Synchronous mode operation.

Note: For Synchronous mode operation, enable the sync source as the last step. Both input capture modules are held in Reset until the sync source is enabled.

Use the ICMx bits of the odd module (ICxCON1<2:0>) to set the desired Capture mode.

The module is ready to capture events when the time base and the sync/trigger source are enabled. When the ICBNE bit (ICxCON1<3>) becomes set, at least one capture value is available in the FIFO. Read input capture values from the FIFO until the ICBNE clears to '0'.

For 32-bit operation, read both the ICxBUF and ICyBUF for the full 32-bit timer value (ICxBUF for the lsw, ICyBUF for the msw). At least one capture value is available in the FIFO buffer when the odd module's ICBNE bit (ICxCON1<3>) becomes set. Continue to read the buffer registers until ICBNE is cleared (performed automatically by hardware).

REGISTER 14-1: ICxCON1: INPUT CAPTURE x CONTROL REGISTER 1

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0
_	_	ICSIDL	ICTSEL2	ICTSEL1	ICTSEL0	_	_
bit 15							bit 8

U-0	R/W-0	R/W-0	R-0, HSC	R-0, HSC	R/W-0	R/W-0	R/W-0
_	ICI1	ICI0	ICOV	ICBNE	ICM2 ⁽¹⁾	ICM1 ⁽¹⁾	ICM0 ⁽¹⁾
bit 7							bit 0

Legend:HSC = Hardware Settable/Clearable bitR = Readable bitW = Writable bitU = Unimplemented bit, read as '0'-n = Value at POR'1' = Bit is set'0' = Bit is clearedx = Bit is unknown

bit 15-14 Unimplemented: Read as '0'

bit 13 ICSIDL: Input Capture x Stop in Idle Control bit

1 = Input capture module halts in CPU Idle mode

0 = Input capture module continues to operate in CPU Idle mode

bit 12-10 ICTSEL<2:0>: Input Capture x Timer Select bits

111 = System clock (Fosc/2)

110 = Reserved

101 = Reserved

100 = Timer1

011 = Timer5

010 = Timer4

001 = Timer2

000 = Timer3

bit 9-7 **Unimplemented:** Read as '0'

bit 6-5 ICI<1:0>: Select Number of Captures per Interrupt bits

11 = Interrupt on every fourth capture event

10 = Interrupt on every third capture event

01 = Interrupt on every second capture event

00 = Interrupt on every capture event

bit 4 **ICOV:** Input Capture x Overflow Status Flag bit (read-only)

1 = Input capture overflow has occurred

0 = No input capture overflow has occurred

bit 3 **ICBNE:** Input Capture x Buffer Empty Status bit (read-only)

1 = Input capture buffer is not empty, at least one more capture value can be read

0 = Input capture buffer is empty

bit 2-0 ICM<2:0>: Input Capture x Mode Select bits⁽¹⁾

111 = Interrupt mode: Input capture functions as an interrupt pin only when the device is in Sleep or Idle mode (rising edge detect only, all other control bits are not applicable)

110 = Unused (module is disabled)

101 = Prescaler Capture mode: Capture on every 16th rising edge

100 = Prescaler Capture mode: Capture on every 4th rising edge

011 = Simple Capture mode: Capture on every rising edge

010 = Simple Capture mode: Capture on every falling edge

001 = Edge Detect Capture mode: Capture on every edge (rising and falling); ICI<1:0> bits do not control interrupt generation for this mode

000 = Input capture module is turned off

Note 1: The ICx input must also be configured to an available RPn/RPIn pin. For more information, see Section 11.4 "Peripheral Pin Select (PPS)".

REGISTER 14-2: ICxCON2: INPUT CAPTURE x CONTROL REGISTER 2

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	
_	_	_	_	_	_	_	IC32	
bit 15 bit 8								

R/W-0	R/W-0, HS	U-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-1
ICTRIG	TRIGSTAT	_	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0
bit 7							bit 0

Legend: HS = Hardware Settable bit

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 15-9 Unimplemented: Read as '0'

bit 8 IC32: Cascade Two IC Modules Enable bit (32-bit operation)

1 = ICx and ICy operate in cascade as a 32-bit module (this bit must be set in both modules)

0 = ICx functions independently as a 16-bit module

bit 7 ICTRIG: Input Capture x Sync/Trigger Select bit

1 = Triggers ICx from the source designated by the SYNCSELx bits

0 = Synchronizes ICx with the source designated by the SYNCSELx bits

bit 6 TRIGSTAT: Timer Trigger Status bit

1 = Timer source has been triggered and is running (set in hardware, can be set in software)

0 = Timer source has not been triggered and is being held clear

bit 5 **Unimplemented:** Read as '0'

Note 1: Use these inputs as trigger sources only and never as sync sources.

2: Never use an IC module as its own trigger source by selecting this mode.

REGISTER 14-2: ICxCON2: INPUT CAPTURE x CONTROL REGISTER 2 (CONTINUED)

bit 4-0 SYNCSEL<4:0>: Synchronization/Trigger Source Selection bits

```
1111x = Reserved
11101 = Reserved
11100 = CTMU^{(1)}
11011 = A/D^{(1)}
11010 = Comparator 3<sup>(1)</sup>
11001 = Comparator 2<sup>(1)</sup>
11000 = Comparator 1<sup>(1)</sup>
10111 = Reserved
10110 = Reserved
10101 = Input Capture 6<sup>(2)</sup>
10100 = Input Capture 5<sup>(2)</sup>
10011 = Input Capture 4<sup>(2)</sup>
10010 = Input Capture 3<sup>(2)</sup>
10001 = Input Capture 2<sup>(2)</sup>
10000 = Input Capture 1<sup>(2)</sup>
01111 = Timer5
01110 = Timer4
01101 = Timer3
01100 = Timer2
01011 = Timer1
01010 = Reserved
01001 = Reserved
01000 = Reserved
00111 = Reserved
00110 = Output Compare 6
00101 = Output Compare 5
00100 = Output Compare 4
00011 = Output Compare 3
00010 = Output Compare 2
00001 = Output Compare 1
00000 = Not synchronized to any other module
```

- **Note 1:** Use these inputs as trigger sources only and never as sync sources.
 - 2: Never use an IC module as its own trigger source by selecting this mode.

NOTES:

15.0 OUTPUT COMPARE WITH DEDICATED TIMERS

Note:

This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the "dsPIC33/PIC24 Family Reference Manual", "Output Compare with Dedicated Timer" (DS70005159). The information in this data sheet supersedes the information in the FRM.

Devices in the PIC24FJ128GB204 family all feature six independent output compare modules. Each of these modules offers a wide range of configuration and operating options for generating pulse trains on internal device events, and can produce Pulse-Width Modulated (PWM) waveforms for driving power applications.

Key features of the output compare module include:

- Hardware-configurable for 32-bit operation in all modes by cascading two adjacent modules
- Synchronous and Trigger modes of output compare operation, with up to 31 user-selectable trigger/sync sources available
- Two separate Period registers (a main register, OCxR, and a secondary register, OCxRS) for greater flexibility in generating pulses of varying widths
- Configurable for single pulse or continuous pulse generation on an output event, or continuous PWM waveform generation
- Up to 6 clock sources available for each module, driving a separate internal 16-bit counter

15.1 General Operating Modes

15.1.1 SYNCHRONOUS AND TRIGGER MODES

When the output compare module operates in a Free-Running mode, the internal 16-bit counter, OCxTMR, runs counts up continuously, wrapping around from 0xFFFF to 0x0000 on each overflow. Its period is synchronized to the selected external clock source. Compare or PWM events are generated each time a match between the internal counter and one of the Period registers occurs.

In Synchronous mode, the module begins performing its compare or PWM operation as soon as its selected clock source is enabled. Whenever an event occurs on the selected sync source, the module's internal counter is reset. In Trigger mode, the module waits for a sync event from another internal module to occur before allowing the counter to run.

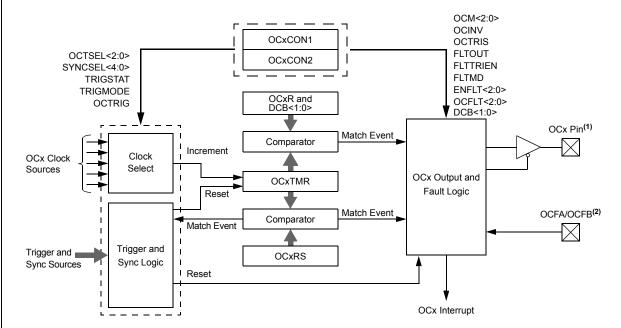
Free-Running mode is selected by default or any time that the SYNCSELx bits (OCxCON2<4:0>) are set to '00000'. Synchronous or Trigger modes are selected any time the SYNCSELx bits are set to any value except '00000'. The OCTRIG bit (OCxCON2<7>) selects either Synchronous or Trigger mode; setting the bit selects Trigger mode operation. In both modes, the SYNCSELx bits determine the sync/trigger source.

15.1.2 CASCADED (32-BIT) MODE

By default, each module operates independently with its own set of 16-bit Timer and Duty Cycle registers. To increase resolution, adjacent even and odd modules can be configured to function as a single 32-bit module. (For example, Modules 1 and 2 are paired, as are Modules 3 and 4, and so on.) The odd numbered module, Output Compare x (OCx), provides the Least Significant 16 bits of the 32-bit register pairs and the even numbered module, Output Compare y (OCy), provides the Most Significant 16 bits. Wrap arounds of the OCx registers cause an increment of their corresponding OCy registers.

Cascaded operation is configured in hardware by setting the OC32 bit (OCxCON2<8>) for both modules. For more information on cascading, refer to the "dsPIC33/PIC24 Family Reference Manual", "Output Compare with Dedicated Timer" (DS70005159).

FIGURE 15-1: OUTPUT COMPARE x BLOCK DIAGRAM (16-BIT MODE)



- Note 1: The OCx outputs must be assigned to an available RPn pin before use. For more information, see Section 11.4 "Peripheral Pin Select (PPS)".
 - 2: The OCFA/OCFB Fault inputs must be assigned to an available RPn/RPIn pin before use. For more information, see Section 11.4 "Peripheral Pin Select (PPS)".

15.2 Compare Operations

In Compare mode (Figure 15-1), the output compare module can be configured for single-shot or continuous pulse generation. It can also repeatedly toggle an output pin on each timer event.

To set up the module for compare operations:

- 1. Configure the OCx output for one of the available Peripheral Pin Select pins.
- Calculate the required values for the OCxR and (for Double Compare modes) OCxRS Duty Cycle registers:
 - Determine the instruction clock cycle time.
 Take into account the frequency of the external clock to the timer source (if one is used) and the timer prescaler settings.
 - b) Calculate the time to the rising edge of the output pulse relative to the timer start value (0000h).
 - c) Calculate the time to the falling edge of the pulse based on the desired pulse width and the time to the rising edge of the pulse.

- Write the rising edge value to OCxR and the falling edge value to OCxRS.
- Set the Timer Period register, PRy, to a value equal to or greater than the value in OCxRS.
- 5. Set the OCM<2:0> bits for the appropriate compare operation ('0xx').
- For Trigger mode operations, set OCTRIG to enable Trigger mode. Set or clear TRIGMODE to configure trigger operation and TRIGSTAT to select a hardware or software trigger. For Synchronous mode, clear OCTRIG.
- 7. Set the SYNCSEL<4:0> bits to configure the trigger or synchronization source. If free-running timer operation is required, set the SYNCSELx bits to '00000' (no sync/trigger source).
- 8. Select the time base source with the OCTSEL<2:0> bits. If necessary, set the TON bit for the selected timer, which enables the compare time base to count. Synchronous mode operation starts as soon as the time base is enabled; Trigger mode operation starts after a trigger source event occurs.

For 32-bit cascaded operation, these steps are also necessary:

- Set the OC32 bits for both registers (OCyCON2<8>) and (OCxCON2<8>). Enable the even numbered module first to ensure the modules will start functioning in unison.
- Clear the OCTRIG bit of the even module (OCyCON2<7>), so the module will run in Synchronous mode.
- Configure the desired output and Fault settings for OCy.
- 4. Force the output pin for OCx to the output state by clearing the OCTRIS bit.
- If Trigger mode operation is required, configure the trigger options in OCx by using the OCTRIG (OCxCON2<7>), TRIGMODE (OCxCON1<3>) and SYNCSELx (OCxCON2<4:0>) bits.
- 6. Configure the desired Compare or PWM mode of operation (OCM<2:0>) for OCy first, then for OCx.

Depending on the output mode selected, the module holds the OCx pin in its default state and forces a transition to the opposite state when OCxR matches the timer. In Double Compare modes, OCx is forced back to its default state when a match with OCxRS occurs. The OCxIF interrupt flag is set after an OCxR match in Single Compare modes and after each OCxRS match in Double Compare modes.

Single-shot pulse events only occur once, but may be repeated by simply rewriting the value of the OCxCON1 register. Continuous pulse events continue indefinitely until terminated.

15.3 Pulse-Width Modulation (PWM) Mode

In PWM mode, the output compare module can be configured for edge-aligned or center-aligned pulse waveform generation. All PWM operations are double-buffered (buffer registers are internal to the module and are not mapped into SFR space).

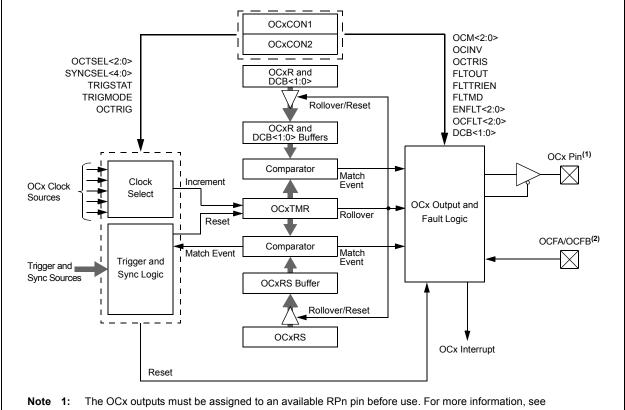
To configure the output compare module for PWM operation:

- 1. Configure the OCx output for one of the available Peripheral Pin Select pins.
- 2. Calculate the desired duty cycles and load them into the OCxR register.
- Calculate the desired period and load it into the OCxRS register.
- Select the current OCx as the synchronization source by writing '0x1F' to the SYNCSEL<4:0> bits (OCxCON2<4:0>) and '0' to the OCTRIG bit (OCxCON2<7>).
- 5. Select a clock source by writing to the OCTSEL<2:0> bits (OCxCON1<12:10>).
- Enable interrupts, if required, for the timer and output compare modules. The output compare interrupt is required for PWM Fault pin utilization.
- Select the desired PWM mode in the OCM<2:0> bits (OCxCON1<2:0>).
- Appropriate Fault inputs may be enabled by using the ENFLT<2:0> bits as described in Register 15-1.
- If a timer is selected as a clock source, set the selected timer prescale value. The selected timer's prescaler output is used as the clock input for the OCx timer and not the selected timer output.

Note:

This peripheral contains input and output functions that may need to be configured by the Peripheral Pin Select. For more information, see Section 11.4 "Peripheral Pin Select (PPS)".

FIGURE 15-2: OUTPUT COMPARE x BLOCK DIAGRAM (DOUBLE-BUFFERED, 16-BIT PWM MODE)



- Section 11.4 "Peripheral Pin Select (PPS)".
 - The OCFA/OCFB Fault inputs must be assigned to an available RPn/RPIn pin before use. For more information, see Section 11.4 "Peripheral Pin Select (PPS)".

15.3.1 **PWM PERIOD**

The PWM period is specified by writing to PRy, the Timer Period register. The PWM period can be calculated using Equation 15-1.

EQUATION 15-1: CALCULATING THE PWM PERIOD⁽¹⁾

 $PWM \ Period = [(PRy) + 1] \bullet TCY \bullet (Timer \ Prescale \ Value)$

where:

PWM Frequency = 1/[PWM Period]

Note 1: Based on Tcy = Tosc * 2; Doze mode and PLL are disabled.

A PRy value of N will produce a PWM period of N + 1 time base count cycles. For example, a value of Note: 7 written into the PRy register will yield a period consisting of 8 time base cycles.

15.3.2 PWM DUTY CYCLE

The PWM duty cycle is specified by writing to the OCxRS and OCxR registers. The OCxRS and OCxR registers can be written to at any time, but the duty cycle value is not latched until a match between PRy and TMRy occurs (i.e., the period is complete). This provides a double buffer for the PWM duty cycle and is essential for glitchless PWM operation.

Some important boundary parameters of the PWM duty cycle include:

- If OCxR, OCxRS and PRy are all loaded with 0000h, the OCx pin will remain low (0% duty cycle).
- If OCxRS is greater than PRy, the pin will remain high (100% duty cycle).

See Example 15-1 for PWM mode timing details. Table 15-1 and Table 15-2 show example PWM frequencies and resolutions for a device operating at 4 MIPS and 10 MIPS, respectively.

EQUATION 15-2: CALCULATION FOR MAXIMUM PWM RESOLUTION(1)

$$\label{eq:maximum_pwm} \begin{aligned} & log_{10} \left(\frac{FCY}{FPWM \bullet (Timer\ Prescale\ Value)} \right) \\ & log_{10} \end{aligned} \ bits$$

Note 1: Based on Fcy = Fosc/2; Doze mode and PLL are disabled.

EXAMPLE 15-1: PWM PERIOD AND DUTY CYCLE CALCULATIONS⁽¹⁾

1. Find the Timer Period register value for a desired PWM frequency of 52.08 kHz, where Fosc = 8 MHz with PLL (32 MHz device clock rate) and a Timer2 prescaler setting of 1:1.

19.2
$$\mu$$
s = (PR2 + 1) • 62.5 ns • 1

$$PR2 = 306$$

2. Find the maximum resolution of the duty cycle that can be used with a 52.08 kHz frequency and a 32 MHz device clock rate.

PWM Resolution = $log_{10}(FCY/FPWM)/log_{10}2)$ bits

 $= (\log_{10}(16 \text{ MHz}/52.08 \text{ kHz})/\log_{10}2) \text{ bits}$

= 8.3 bits

Note 1: Based on Tcy = 2 * Tosc; Doze mode and PLL are disabled.

TABLE 15-1: EXAMPLE PWM FREQUENCIES AND RESOLUTIONS AT 4 MIPS (FcY = 4 MHz)⁽¹⁾

PWM Frequency	7.6 Hz	61 Hz	122 Hz	977 Hz	3.9 kHz	31.3 kHz	125 kHz
Timer Prescaler Ratio	8	1	1	1	1	1	1
Period Register Value	FFFFh	FFFFh	7FFFh	0FFFh	03FFh	007Fh	001Fh
Resolution (bits)	16	16	15	12	10	7	5

Note 1: Based on Fcy = Fosc/2; Doze mode and PLL are disabled.

TABLE 15-2: EXAMPLE PWM FREQUENCIES AND RESOLUTIONS AT 16 MIPS (FcY = 16 MHz)⁽¹⁾

PWM Frequency	30.5 Hz	244 Hz	488 Hz	3.9 kHz	15.6 kHz	125 kHz	500 kHz
Timer Prescaler Ratio	8	1	1	1	1	1	1
Period Register Value	FFFFh	FFFFh	7FFFh	0FFFh	03FFh	007Fh	001Fh
Resolution (bits)	16	16	15	12	10	7	5

Note 1: Based on Fcy = Fosc/2; Doze mode and PLL are disabled.

REGISTER 15-1: OCxCON1: OUTPUT COMPARE x CONTROL REGISTER 1

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	OCSIDL	OCTSEL2	OCTSEL1	OCTSEL0	ENFLT2 ⁽²⁾	ENFLT1 ⁽²⁾
bit 15							bit 8

R/W-0	R/W-0, HSC	R/W-0, HSC	R/W-0, HSC	R/W-0	R/W-0	R/W-0	R/W-0
ENFLT0 ⁽²⁾	OCFLT2 ^(2,3)	OCFLT1 ^(2,4)	OCFLT0 ^(2,4)	TRIGMODE	OCM2 ⁽¹⁾	OCM1 ⁽¹⁾	OCM0 ⁽¹⁾
bit 7							bit 0

Legend:	HSC = Hardware Settable/Clearable bit				
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 15-14 **Unimplemented:** Read as '0'

bit 13 OCSIDL: Output Compare x Stop in Idle Mode Control bit

1 = Output Compare x halts in CPU Idle mode

0 = Output Compare x continues to operate in CPU Idle mode

bit 12-10 OCTSEL<2:0>: Output Compare x Timer Select bits

111 = Peripheral clock (FcY)

110 = Reserved

101 = Reserved

100 = Timer1 clock (only synchronous clock is supported)

011 = Timer5 clock 010 = Timer4 clock 001 = Timer3 clock

000 = Timer2 clock

bit 9 **ENFLT2:** Fault Input 2 Enable bit⁽²⁾

1 = Fault 2 (Comparator 1/2/3 out) is enabled(3)

0 = Fault 2 is disabled

bit 8 **ENFLT1:** Fault Input 1 Enable bit⁽²⁾

1 = Fault 1 (OCFB pin) is enabled(4)

0 = Fault 1 is disabled

bit 7 **ENFLT0**: Fault Input 0 Enable bit⁽²⁾

1 = Fault 0 (OCFA pin) is enabled (4)

0 = Fault 0 is disabled

bit 6 OCFLT2: Output Compare x PWM Fault 2 (Comparator 1/2/3) Condition Status bit(2,3)

1 = PWM Fault 2 has occurred0 = No PWM Fault 2 has occurred

bit 5 OCFLT1: Output Compare x PWM Fault 1 (OCFB pin) Condition Status bit(2,4)

1 = PWM Fault 1 has occurred

0 = No PWM Fault 1 has occurred

- Note 1: The OCx output must also be configured to an available RPn pin. For more information, see Section 11.4 "Peripheral Pin Select (PPS)".
 - 2: The Fault input enable and Fault status bits are valid when OCM<2:0> = 111 or 110.
 - **3:** The Comparator 1 output controls the OC1-OC2 channels; Comparator 2 output controls the OC3-OC4 channels; Comparator 3 output controls the OC5-OC6 channels.
 - 4: The OCFA/OCFB Fault input must also be configured to an available RPn/RPIn pin. For more information, see Section 11.4 "Peripheral Pin Select (PPS)".

REGISTER 15-1: OCXCON1: OUTPUT COMPARE x CONTROL REGISTER 1 (CONTINUED)

- bit 4 OCFLT0: Output Compare x PWM Fault 0 (OCFA pin) Condition Status bit^(2,4)
 - 1 = PWM Fault 0 has occurred
 - 0 = No PWM Fault 0 has occurred
- bit 3 TRIGMODE: Trigger Status Mode Select bit
 - 1 = TRIGSTAT (OCxCON2<6>) is cleared when OCxRS = OCxTMR or in software
 - 0 = TRIGSTAT is only cleared by software
- bit 2-0 OCM<2:0>: Output Compare x Mode Select bits⁽¹⁾
 - 111 = Center-Aligned PWM mode on OCx(2)
 - 110 = Edge-Aligned PWM mode on OCx(2)
 - 101 = Double Compare Continuous Pulse mode: Initializes the OCx pin low; toggles the OCx state continuously on alternate matches of OCxR and OCxRS
 - 100 = Double Compare Single-Shot mode: Initializes the OCx pin low; toggles the OCx state on matches of OCxR and OCxRS for one cycle
 - 011 = Single Compare Continuous Pulse mode: Compare events continuously toggle the OCx pin
 - 010 = Single Compare Single-Shot mode: Initializes OCx pin high; compare event forces the OCx pin low
 - 001 = Single Compare Single-Shot mode: Initializes OCx pin low; compare event forces the OCx pin high
 - 000 = Output compare channel is disabled
- Note 1: The OCx output must also be configured to an available RPn pin. For more information, see Section 11.4 "Peripheral Pin Select (PPS)".
 - 2: The Fault input enable and Fault status bits are valid when OCM<2:0> = 111 or 110.
 - **3:** The Comparator 1 output controls the OC1-OC2 channels; Comparator 2 output controls the OC3-OC4 channels; Comparator 3 output controls the OC5-OC6 channels.
 - 4: The OCFA/OCFB Fault input must also be configured to an available RPn/RPIn pin. For more information, see Section 11.4 "Peripheral Pin Select (PPS)".

REGISTER 15-2: OCxCON2: OUTPUT COMPARE x CONTROL REGISTER 2

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
FLTMD	FLTOUT	FLTTRIEN	OCINV	_	DCB1 ⁽³⁾	DCB0 ⁽³⁾	OC32
bit 15							bit 8

R/W-0	R/W-0, HS	R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0
OCTRIG	TRIGSTAT	OCTRIS	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0
bit 7							bit 0

Legend:HS = Hardware Settable bitR = Readable bitW = Writable bitU = Unimplemented bit, read as '0'-n = Value at POR'1' = Bit is set'0' = Bit is clearedx = Bit is unknown

bit 15 FLTMD: Fault Mode Select bit

1 = Fault mode is maintained until the Fault source is removed and the corresponding OCFLT0 bit is cleared in software

0 = Fault mode is maintained until the Fault source is removed and a new PWM period starts

bit 14 FLTOUT: Fault Out bit

1 = PWM output is driven high on a Fault

0 = PWM output is driven low on a Fault

bit 13 FLTTRIEN: Fault Output State Select bit

1 = Pin is forced to an output on a Fault condition

0 = Pin I/O condition is unaffected by a Fault

bit 12 OCINV: Output Compare x Invert bit

1 = OCx output is inverted

0 = OCx output is not inverted

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

bit 10-9 **DCB<1:0>:** PWM Duty Cycle Least Significant bits⁽³⁾

11 = Delays OCx falling edge by ¾ of the instruction cycle

10 = Delays OCx falling edge by ½ of the instruction cycle

01 = Delays OCx falling edge by 1/4 of the instruction cycle

00 = OCx falling edge occurs at the start of the instruction cycle

bit 8 OC32: Cascade Two OC Modules Enable bit (32-bit operation)

1 = Cascade module operation is enabled

0 = Cascade module operation is disabled

bit 7 OCTRIG: Output Compare x Trigger/Sync Select bit

1 = Triggers OCx from the source designated by the SYNCSELx bits

0 = Synchronizes OCx with the source designated by the SYNCSELx bits

bit 6 TRIGSTAT: Timer Trigger Status bit

1 = Timer source has been triggered and is running

0 = Timer source has not been triggered and is being held clear

bit 5 OCTRIS: Output Compare x Output Pin Direction Select bit

1 = OCx pin is tri-stated

0 = Output Compare Peripheral x is connected to an OCx pin

Note 1: Never use an OCx module as its own trigger source, either by selecting this mode or another equivalent SYNCSELx setting.

2: Use these inputs as trigger sources only and never as sync sources.

3: The DCB<1:0> bits are double-buffered in PWM modes only (OCM<2:0> (OCxCON1<2:0>) = 111, 110).

REGISTER 15-2: OCXCON2: OUTPUT COMPARE x CONTROL REGISTER 2 (CONTINUED)

```
bit 4-0
                SYNCSEL<4:0>: Trigger/Synchronization Source Selection bits
                11111 = This OC module<sup>(1)</sup>
                11110 = OCTRIG1 external input
                11101 = OCTRIG2 external input
                11100 = CTMU<sup>(2)</sup>
                11011 = A/D^{(2)}
                11010 = Comparator 3<sup>(2)</sup>
                11001 = Comparator 2<sup>(2)</sup>
                11000 = Comparator 1<sup>(2)</sup>
                10111 = Reserved
                10110 = Reserved
                10101 = Input Capture 6<sup>(2)</sup>
                10100 = Input Capture 5<sup>(2)</sup>
                10011 = Input Capture 4<sup>(2)</sup>
                10010 = Input Capture 3<sup>(2)</sup>
                10001 = Input Capture 2<sup>(2)</sup>
                10000 = Input Capture 1<sup>(2)</sup>
                01111 = Timer5
                01110 = Timer4
                01101 = Timer3
                01100 = Timer2
                01011 = Timer1
                01010 = Reserved
                01001 = Reserved
                01000 = Reserved
                00111 = Reserved
                00110 = Output Compare 6<sup>(1)</sup>
                00101 = Output Compare 5^{(1)}
                00100 = Output Compare 4(1)
                00011 = Output Compare 3<sup>(1)</sup>
                00010 = Output Compare 2<sup>(1)</sup>
                00001 = Output Compare 1<sup>(1)</sup>
                00000 = Not synchronized to any other module
```

- **Note 1:** Never use an OCx module as its own trigger source, either by selecting this mode or another equivalent SYNCSELx setting.
 - 2: Use these inputs as trigger sources only and never as sync sources.
 - 3: The DCB<1:0> bits are double-buffered in PWM modes only (OCM<2:0> (OCxCON1<2:0>) = 111, 110).

PICZ4FJ	PICZ4FJ IZOGBZU4 FAIVIIL I					
NOTES:						

16.0 SERIAL PERIPHERAL INTERFACE (SPI)

Note:

This data sheet summarizes the features of the PIC24FJ128GB204 family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to the "dsPIC33/PIC24 Family Reference Manual", "Serial Peripheral Interface (SPI) with Audio Codec Support" (DS70005136) which is available from the Microchip web site (www.microchip.com).

The Serial Peripheral Interface (SPI) module is a synchronous serial interface useful for communicating with other peripheral or microcontroller devices. These peripheral devices may be serial EEPROMs, shift registers, display drivers, A/D Converters, etc. The SPI module is compatible with the Motorola® SPI and SIOP interfaces. All devices in the PIC24FJ128GB204 family include three SPI modules.

The module supports operation in two Buffer modes. In Standard Buffer mode, data is shifted through a single serial buffer. In Enhanced Buffer mode, data is shifted through a FIFO buffer. The FIFO level depends on the configured mode.

Variable length data can be transmitted and received, from 2 to 32 bits.

Note:

Do not perform Read-Modify-Write operations (such as bit-oriented instructions) on the SPIxBUF register in either Standard or Enhanced Buffer mode.

The module also supports a basic framed SPI protocol while operating in either Master or Slave mode. A total of four framed SPI configurations are supported.

The module also supports Audio modes. Four different Audio modes are available.

- I²S mode
- · Left Justified
- · Right Justified
- · PCM/DSP

In each of these modes, the serial clock is free-running and audio data is always transferred.

If an audio protocol data transfer takes place between two devices, then usually one device is the master and the other is the slave. However, audio data can be transferred between two slaves. Because the audio protocols require free-running clocks, the master can be a third party controller. In either case, the master generates two free-running clocks: SCKx and LRC (Left, Right Channel Clock/SSx/FSYNC).

The SPI serial interface consists of four pins:

- · SDIx: Serial Data Input
- · SDOx: Serial Data Output
- SCKx: Shift Clock Input or Output
- SSx: Active-Low Slave Select or Frame Synchronization I/O Pulse

The SPI module can be configured to operate using 2, 3 or 4 pins. In the 3-pin mode, \overline{SSx} is not used. In the 2-pin mode, both SDOx and \overline{SSx} are not used.

The SPI module has the ability to generate three interrupts, reflecting the events that occur during the data communication. The following types of interrupts can be generated:

- Receive interrupts are signalled by SPIxRXIF. This event occurs when:
 - RX watermark interrupt
 - SPIROV = 1
 - SPIRBF = 1
 - **SPIRBE** = 1

provided the respective mask bits are enabled in SPIxIMSKL/H.

- Transmit interrupts are signalled by SPIxTXIF. This event occurs when:
 - TX watermark interrupt
 - SPITUR = 1
 - SPITBF = 1
 - SPITBE = 1

provided the respective mask bits are enabled in SPIxIMSKL/H.

- 3. General interrupts are signalled by SPIxIF. This event occurs when
 - FRMERR = 1
 - SPIBUSY = 1
 - SRMT = 1

provided the respective mask bits are enabled in SPIxIMSKL/H.

Block diagrams of the module in Standard and Enhanced modes are shown in Figure 16-1 and Figure 16-2.

Note

In this section, the SPI modules are referred to together as SPIx, or separately as SPI1, SPI2 or SPI3. Special Function Registers will follow a similar notation. For example, SPIxCON1L and SPIxCON1H refer to the control registers for any of the three SPI modules.

16.1 Standard Master Mode

To set up the SPIx module for the Standard Master mode of operation:

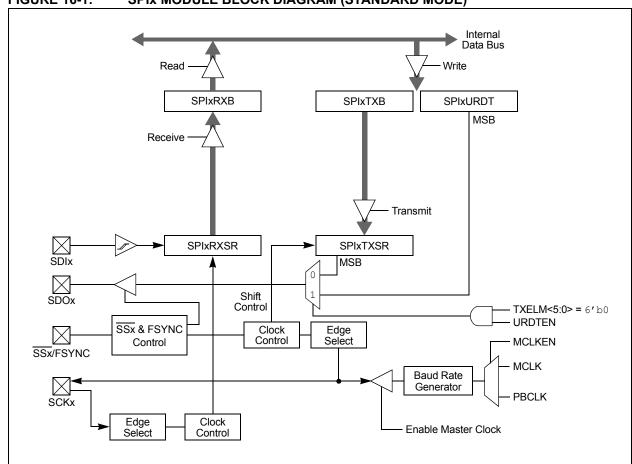
- 1. If using interrupts:
 - a) Clear the interrupt flag bits in the respective IFSx register.
 - Set the interrupt enable bits in the respective IECx register.
 - c) Write the SPIxIP<2:0> bits in the respective IPCx register to set the interrupt priority.
- Write the desired settings to the SPIxCON1L and SPIxCON1H registers with the MSTEN bit (SPIxCON1L<5>) = 1.
- 3. Clear the SPIROV bit (SPIxSTATL<6>).
- Enable SPIx operation by setting the SPIEN bit (SPIxCON1L<15>).
- Write the data to be transmitted to the SPIxBUFL and SPIxBUFH registers. Transmission (and reception) will start as soon as data is written to the SPIxBUFL and SPIxBUFH registers.

16.2 Standard Slave Mode

To set up the SPIx module for the Standard Slave mode of operation:

- 1. Clear the SPIxBUF registers.
- If using interrupts:
 - a) Clear the SPIxBUFL and SPIxBUFH registers.
 - b) Set the interrupt enable bits in the respective IECx register.
 - Write the SPIxIP<2:0> bits in the respective IPCx register to set the interrupt priority.
- Write the desired settings to the SPIxCON1L, SPIxCON1H and SPIxCON2L registers with the MSTEN bit (SPIxCON1L<5>) = 0.
- 4. Clear the SMP bit.
- If the CKE bit (SPIxCON1L<8>) is set, then the SSEN bit (SPIxCON1L<7>) must be set to enable the SSx pin.
- 6. Clear the SPIROV bit (SPIxSTATL<6>).
- 7. Enable SPIx operation by setting the SPIEN bit (SPIxCON1L<15>).

FIGURE 16-1: SPIX MODULE BLOCK DIAGRAM (STANDARD MODE)



16.3 Enhanced Slave Mode

To set up the SPIx module for the Enhanced Buffer Master mode of operation:

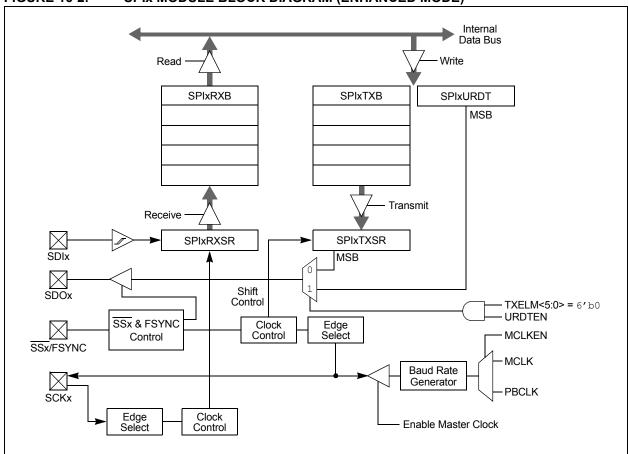
- 1. If using interrupts:
 - Clear the interrupt flag bits in the respective IFSx register.
 - b) Set the interrupt enable bits in the respective IECx register.
 - Write the SPIxIP<2:0> bits in the respective IPCx register.
- Write the desired settings to the SPIxCON1L, SPIxCON1H and SPIxCON2L registers with MSTEN (SPIxCON1L<5>) = 1.
- 3. Clear the SPIROV bit (SPIxSTATL<6>).
- Select Enhanced Buffer mode by setting the ENHBUF bit (SPIxCON1L<0>).
- Enable SPIx operation by setting the SPIEN bit (SPIxCON1L<15>).
- Write the data to be transmitted to the SPIxBUFL and SPIxBUFH registers. Transmission (and reception) will start as soon as data is written to the SPIxBUFL and SPIxBUFH registers.

16.4 Enhanced Master Mode

To set up the SPIx module for the Enhanced Buffer Slave mode of operation:

- 1. Clear the SPIxBUFL and SPIxBUFH registers.
- If using interrupts:
 - Clear the interrupt flag bits in the respective IFSx register.
 - Set the interrupt enable bits in the respective IECx register.
 - Write the SPIxIP<2:0> bits in the respective IPCx register to set the interrupt priority.
- Write the desired settings to the SPIxCON1L, SPIxCON1H and SPIxCON2L registers with the MSTEN bit (SPIxCON1L<5>) = 0.
- 4. Clear the SMP bit.
- 5. If the CKE bit is set, then the SSEN bit must be set, thus enabling the SSx pin.
- 6. Clear the SPIROV bit (SPIxSTATL<6>).
- 7. Select Enhanced Buffer mode by setting the ENHBUF bit (SPIxCON1L<0>).
- 8. Enable SPIx operation by setting the SPIEN bit (SPIxCON1L<15>).

FIGURE 16-2: SPIX MODULE BLOCK DIAGRAM (ENHANCED MODE)



16.5 Audio Mode

To set up the SPIx module for the Audio mode:

- 1. Clear the SPIxBUFL and SPIxBUFH registers.
- 2. If using interrupts:
 - Clear the interrupt flag bits in the respective IFSx register.
 - b) Set the interrupt enable bits in the respective IECx register.
 - a) Write the SPIxIP<2:0> bits in the respective IPCx register to set the interrupt priority.
- 3. Write the desired settings to the SPIxCON1L, SPIxCON1H and SPIxCON2L registers with AUDEN (SPIxCON1H<15>) = 1.
- 4. Clear the SPIROV bit (SPIxSTATL<6>).
- Enable SPIx operation by setting the SPIEN bit (SPIxCON1L<15>).
- 6. Write the data to be transmitted to the SPIxBUFL and SPIxBUFH registers. Transmission (and reception) will start as soon as data is written to the SPIxBUFL and SPIxBUFH registers.

16.6 Registers

The SPI module consists of the following Special Function Registers (SFRs):

- SPIxCON1L, SPIxCON1H and SPIxCON2L: SPIx Control Registers (Register 16-1 through Register 16-3)
- SPIxSTATL and SPIxSTATH: SPIx Status Registers (Register 16-4 and Register 16-5)
- · SPIxBUFL and SPIxBUFH: SPIx Buffer Registers
- · SPIxBRGL: SPIx Baud Rate Register
- SPIxIMSKL and SPIxIMSKH: SPIx Interrupt Mask Registers (Register 16-6 and Register 16-7)
- SPIxURDTL and SPIxURDTH: SPIx Underrun Data Registers

REGISTER 16-1: SPIxCON1L: SPIx CONTROL REGISTER 1 LOW

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SPIEN	_	SPISIDL	DISSDO	MODE32 ^(1,4)	MODE16 ^(1,4)	SMP	CKE ⁽¹⁾
bit 15							bit 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
SSEN ⁽²⁾	CKP	MSTEN	DISSDI	DISSCK	MCLKEN ⁽³⁾	SPIFE	ENHBUF
bit 7							bit 0

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

bit 15 SPIEN: SPIx On bit

1 = Enables module

0 = Turns off and resets module, disables clocks, disables interrupt event generation, allows SFR modifications

bit 14 Unimplemented: Read as '0'

bit 13 SPISIDL: SPIx Stop in Idle Mode bit

1 = Halts in CPU Idle mode

0 = Continues to operate in CPU Idle mode

bit 12 DISSDO: Disable SDOx Output Port bit

1 = SDOx pin is not used by the module; pin is controlled by the port function

0 = SDOx pin is controlled by the module

bit 11-10 MODE<32,16>: Serial Word Length bits^(1,4)

AUDEN = 0:

MODE32	MODE16	COMMUNICATION
1	X	32-Bit
0	1	16-Bit
0	0	8-Bit

AUDEN = 1:

MODE32	MODE16	COMMUNICATION
1	1	24-Bit Data, 32-Bit FIFO, 32-Bit Channel/64-Bit Frame
1	0	32-Bit Data, 32-Bit FIFO, 32-Bit Channel/64-Bit Frame
0	1	16-Bit Data, 16-Bit FIFO, 32-Bit Channel/64-Bit Frame
0	0	16-Bit Data 16-Bit FIFO 16-Bit Channel/32-Bit Frame

bit 9 SMP: SPIx Data Input Sample Phase bit

Master Mode:

- 1 = Input data is sampled at the end of data output time
- 0 = Input data is sampled at the middle of data output time

Slave Mode:

Input data is always sampled at the middle of data output time, regardless of the SMP bit setting.

bit 8 **CKE:** SPIx Clock Edge Select bit⁽¹⁾

- 1 = Transmit happens on transition from active clock state to Idle clock state
- 0 = Transmit happens on transition from Idle clock state to active clock state
- Note 1: When AUDEN = 1, this module functions as if CKE = 0, regardless of its actual value.
 - 2: When FRMEN = 1, SSEN is not used.
 - 3: MCLKEN can only be written when the SPIEN bit = 0.
 - **4:** This channel is not meaningful for DSP/PCM mode as LRC follows FRMSYPW.

REGISTER 16-1: SPIxCON1L: SPIx CONTROL REGISTER 1 LOW (CONTINUED)

SSEN: Slave Select Enable bit (Slave mode)(2) bit 7 $1 = \overline{SSx}$ pin is used by the macro in Slave mode; \overline{SSx} pin is used as the slave select input $0 = \overline{SSx}$ pin is not used by the macro (\overline{SSx} pin will be controlled by the port I/O) bit 6 **CKP:** Clock Polarity Select bit 1 = Idle state for clock is a high level; active state is a low level 0 = Idle state for clock is a low level; active state is a high level bit 5 MSTEN: Master Mode Enable bit 1 = Master mode 0 = Slave mode bit 4 **DISSDI:** Disable SDIx Input Port bit 1 = SDIx pin is not used by the module; pin is controlled by the port function 0 = SDIx pin is controlled by the module bit 3 **DISSCK:** Disable SCKx Output Port bit 1 = SCKx pin is not used by the module; pin is controlled by the port function 0 = SCKx pin is controlled by the module bit 2 MCLKEN: Master Clock Enable bit(3) 1 = MCLK is used by the BRG 0 = PBCLK is used by the BRG bit 1 SPIFE: Frame Sync Pulse Edge Select bit 1 = Frame Sync pulse (Idle-to-active edge) coincides with the first bit clock 0 = Frame Sync pulse (Idle-to-active edge) precedes the first bit clock bit 0 **ENHBUF:** Enhanced Buffer Mode Enable bit 1 = Enhanced Buffer mode is enabled 0 = Enhanced Buffer mode is disabled **Note 1:** When AUDEN = 1, this module functions as if CKE = 0, regardless of its actual value.

- 2: When FRMEN = 1, SSEN is not used.
- 3: MCLKEN can only be written when the SPIEN bit = 0.
- 4: This channel is not meaningful for DSP/PCM mode as LRC follows FRMSYPW.

REGISTER 16-2: SPIXCON1H: SPIX CONTROL REGISTER 1 HIGH

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
AUDEN ⁽¹⁾	SPISGNEXT	IGNROV	IGNTUR	AUDMONO ⁽²⁾	URDTEN ⁽³⁾	AUDMOD1 ⁽⁴⁾	AUDMOD0 ⁽⁴⁾
bit 15							bit 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
FRMEN	FRMSYNC	FRMPOL	MSSEN	FRMSYPW	FRMCNT2	FRMCNT1	FRMCNT0
bit 7							bit 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 15 AUDEN: Audio Codec Support Enable bit (1)
 - 1 = Audio protocol is enabled; MSTEN controls the direction of both SCKx and Frame (a.k.a. LRC), and this module functions as if FRMEN = 1, FRMSYNC = MSTEN, FRMCNT<2:0> = 001 and SMP = 0, regardless of their actual values
 - 0 = Audio protocol is disabled
- bit 14 SPISGNEXT: SPIx Sign-Extend RX FIFO Read Data Enable bit
 - 1 = Data from RX FIFO is sign-extended
 - 0 = Data from RX FIFO is not sign-extended
- bit 13 IGNROV: Ignore Receive Overflow bit
 - 1 = A Receive Overflow (ROV) is NOT a critical error; during ROV, data in the FIFO is not overwritten by the receive data
 - 0 = A ROV is a critical error that stops SPI operation
- bit 12 IGNTUR: Ignore Transmit Underrun bit
 - 1 = A Transmit Underrun (TUR) is NOT a critical error and data indicated by URDTEN is transmitted until the SPIxTXB is not empty
 - 0 = A TUR is a critical error that stops SPI operation
- bit 11 **AUDMONO:** Audio Data Format Transmit bit⁽²⁾
 - 1 = Audio data is mono (i.e., each data word is transmitted on both left and right channels)
 - 0 = Audio data is stereo
- bit 10 **URDTEN:** Transmit Underrun Data Enable bit⁽³⁾
 - 1 = Transmits data out of SPIxURDTL/H registers during Transmit Underrun (TUR) conditions
 - 0 = Transmits the last received data during Transmit Underrun conditions
- bit 9-8 **AUDMOD<1:0>:** Audio Protocol Mode Selection bits⁽⁴⁾
 - 11 = PCM/DSP mode
 - 10 = Right Justified mode: This module functions as if SPIFE = 1, regardless of its actual value
 - 01 = Left Justified mode: This module functions as if SPIFE = 1, regardless of its actual value
 - $00 = I^2S$ mode: This module functions as if SPIFE = 0, regardless of its actual value
- bit 7 FRMEN: Framed SPIx Support bit
 - 1 = Framed SPIx support is enabled (\overline{SSx} pin is used as the FSYNC input/output)
 - 0 = Framed SPIx support is disabled
- **Note 1:** AUDEN can only be written when the SPIEN bit = 0.
 - **2:** AUDMONO can only be written when the SPIEN bit = 0 and is only valid for AUDEN = 1.
 - 3: URDTEN is only valid when IGNTUR = 1.
 - **4:** AUDMOD<1:0> can only be written when the SPIEN bit = 0 and are only valid when AUDEN = 1. When NOT in PCM/DSP mode, this module functions as if FRMSYPW = 1, regardless of its actual value.

REGISTER 16-2: SPIXCON1H: SPIX CONTROL REGISTER 1 HIGH (CONTINUED)

- bit 6 FRMSYNC: Frame Sync Pulse Direction Control bit
 - 1 = Frame Sync pulse input (slave)
 - 0 = Frame Sync pulse output (master)
- bit 5 FRMPOL: Frame Sync/Slave Select Polarity bit
 - 1 = Frame Sync pulse/slave select is active-high
 - 0 = Frame Sync pulse/slave select is active-low
- bit 4 MSSEN: Master Mode Slave Select Enable bit
 - 1 = SPIx slave select support is enabled with polarity determined by FRMPOL (SSx pin is automatically driven during transmission in Master mode)
 - 0 = SPIx slave select support is disabled (SSx pin will be controlled by port I/O)
- bit 3 FRMSYPW: Frame Sync Pulse-Width bit
 - 1 = Frame Sync pulse is one serial word length wide (as defined by MODE<32,16>/WLENGTH<4:0>)
 - 0 = Frame Sync pulse is one clock (SCK) wide
- bit 2-0 FRMCNT<2:0>: Frame Sync Pulse Counter bits

Controls the number of serial words transmitted per Sync pulse.

- 111 = Reserved
- 110 = Reserved
- 101 = Generates a Frame Sync pulse on every 32 serial words
- 100 = Generates a Frame Sync pulse on every 16 serial words
- 011 = Generates a Frame Sync pulse on every 8 serial words
- 010 = Generates a Frame Sync pulse on every 4 serial words
- 001 = Generates a Frame Sync pulse on every 2 serial words (value used by audio protocols)
- 000 = Generates a Frame Sync pulse on each serial word
- **Note 1:** AUDEN can only be written when the SPIEN bit = 0.
 - 2: AUDMONO can only be written when the SPIEN bit = 0 and is only valid for AUDEN = 1.
 - **3:** URDTEN is only valid when IGNTUR = 1.
 - **4:** AUDMOD<1:0> can only be written when the SPIEN bit = 0 and are only valid when AUDEN = 1. When NOT in PCM/DSP mode, this module functions as if FRMSYPW = 1, regardless of its actual value.

REGISTER 16-3: SPIXCON2L: SPIX CONTROL REGISTER 2 LOW

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0		
_	_	_	_	_	_	_	_		
bit 15									

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0		
_	_	_	WLENGTH<4:0> ^(1,2)						
bit 7							bit 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 15-5 **Unimplemented:** Read as '0'

bit 4-0 WLENGTH<4:0>: Variable Word Length bits(1,2)

11111 = 32-bit data 11110 = 31-bit data 11101 = 30-bit data 11100 = 29-bit data 11011 = 28-bit data 11010 **= 27-bit data** 11001 = 26-bit data 11000 **= 25-bit data** 10111 = 24-bit data 10110 = 23-bit data 10101 = 22-bit data 10100 = 21-bit data 10011 **= 20-bit data** 10010 **= 19-bit data** 10001 = 18-bit data 10000 = 17-bit data 01111 = 16-bit data 01110 **= 15-bit data** 01101 **= 14-bit data** 01100 **= 13-bit data** 01011 = 12-bit data 01010 = 11-bit data 01001 = 10-bit data 01000 **= 9-bit data** 00111 **= 8-bit data** 00110 **= 7-bit data** 00101 **= 6-bit data** 00100 **= 5-bit data**

00000 = See MODE<32,16> bits in SPIxCON1L<11:10>

2: Varying the length by changing these bits does not affect the depth of the TX/RX FIFO.

Note 1: These bits are effective when AUDEN = 0 only.

00011 = 4-bit data 00010 = 3-bit data 00001 = 2-bit data

REGISTER 16-4: SPIXSTATL: SPIX STATUS REGISTER LOW

U-0	U-0	U-0	R/C-0, HS	R-0, HSC	U-0	U-0	R-0, HSC
_	_	_	FRMERR	SPIBUSY	_	_	SPITUR ⁽¹⁾
bit 15							bit 8

R-0, HSC	R/C-0, HS	R-1, HSC	U-0	R-1, HSC	U-0	R-0, HSC	R-0, HSC
SRMT	SPIROV	SPIRBE	_	SPITBE	_	SPITBF	SPIRBF
bit 7							bit 0

Legend:	C = Clearable bit	t HSC = Hardware Settable/Clearable bit		
R = Readable bit	x = Bit is unknown	U = Unimplemented bit, read as '0'		
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	HS = Hardware Settable bit	

bit 15-13 Unimplemented: Read as '0'

bit 12 FRMERR: SPIx Frame Error Status bit

1 = Frame error is detected0 = No frame error is detected

bit 11 SPIBUSY: SPIx Activity Status bit

1 = Module is currently busy with some transactions

0 = No ongoing transactions (at time of read)

bit 10-9 **Unimplemented:** Read as '0'

bit 8 SPITUR: SPIx Transmit Underrun (TUR) Status bit (1)

1 = Transmit buffer has encountered a Transmit Underrun condition
 0 = Transmit buffer does not have a Transmit Underrun condition

bit 7 SRMT: Shift Register Empty Status bit

1 = No current or pending transactions (i.e., neither SPIxTXB or SPIxTXSR contains data to transmit)

0 = Current or pending transactions

bit 6 SPIROV: SPIx Receive Overflow (ROV) Status bit

1 = A new byte/half-word/word has been completely received when the SPIxRXB was full

0 = No overflow

bit 5 SPIRBE: SPIx RX Buffer Empty Status bit

1 = RX buffer is empty 0 = RX buffer is not empty

Standard Buffer Mode:

Automatically set in hardware when SPIxBUF is read from, reading SPIxRXB. Automatically cleared in hardware when SPIx transfers data from SPIxRXSR to SPIxRXB.

Enhanced Buffer Mode:

Indicates RXELM<5:0> = 6'b000000.

bit 4 Unimplemented: Read as '0'

bit 3 SPITBE: SPIx Transmit Buffer Empty Status bit

1 = SPIxTXB is empty 0 = SPIxTXB is not empty Standard Buffer Mode:

Automatically set in hardware when SPIx transfers data from SPIxTXB to SPIxTXSR. Automatically cleared in hardware when SPIxBUF is written, loading SPIxTXB.

Enhanced Buffer Mode:

Indicates TXELM<5:0> = 6'b000000.

Note 1: SPITUR is cleared when SPIEN = 0. When IGNTUR = 1, SPITUR provides dynamic status of the underrun condition, but does not stop RX/TX operation and does not need to be cleared by software.

REGISTER 16-4: SPIXSTATL: SPIX STATUS REGISTER LOW (CONTINUED)

bit 2 Unimplemented: Read as '0'

bit 1 SPITBF: SPIx Transmit Buffer Full Status bit

1 = SPIxTXB is full 0 = SPIxTXB not full Standard Buffer Mode:

Automatically set in hardware when SPIxBUF is written, loading SPIxTXB. Automatically cleared in hardware when SPIx transfers data from SPIxTXB to SPIxTXSR.

Enhanced Buffer Mode:

Indicates TXELM<5:0> = 6'b111111.

bit 0 SPIRBF: SPIx Receive Buffer Full Status bit

1 = SPIxRXB is full 0 = SPIxRXB is not full Standard Buffer Mode:

Automatically set in hardware when SPIx transfers data from SPIxRXSR to SPIxRXB. Automatically cleared in hardware when SPIxBUF is read from, reading SPIxRXB.

Enhanced Buffer Mode:

Indicates RXELM<5:0> = 6'b111111.

Note 1: SPITUR is cleared when SPIEN = 0. When IGNTUR = 1, SPITUR provides dynamic status of the underrun condition, but does not stop RX/TX operation and does not need to be cleared by software.

REGISTER 16-5: SPIXSTATH: SPIX STATUS REGISTER HIGH

U-0	U-0	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC
_	_	RXELM5 ⁽³⁾	RXELM4 ⁽²⁾	RXELM3 ⁽¹⁾	RXELM2	RXELM1	RXELM0
bit 15							bit 8

U-0	U-0	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC
_	_	TXELM5 ⁽³⁾	TXELM4 ⁽²⁾	TXELM3 ⁽¹⁾	TXELM2	TXELM1	TXELM0
bit 7							bit 0

Legend: HSC = Hardware Settable/Clearable bit

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 15-14 Unimplemented: Read as '0'

bit 13-8 **RXELM<5:0>:** Receive Buffer Element Count bits (valid in Enhanced Buffer mode)^(1,2,3)

bit 7-6 Unimplemented: Read as '0'

bit 5-0 **TXELM<5:0>:** Transmit Buffer Element Count bits (valid in Enhanced Buffer mode)(1,2,3)

Note 1: RXELM3 and TXELM3 bits are only present when FIFODEPTH = 8 or higher.

2: RXELM4 and TXELM4 bits are only present when FIFODEPTH = 16 or higher.

3: RXELM5 and TXELM5 bits are only present when FIFODEPTH = 32.

REGISTER 16-6: SPIXIMSKL: SPIX INTERRUPT MASK REGISTER LOW

U-0	U-0	U-0	R/W-0	R/W-0	U-0	U-0	R/W-0
_	_	_	FRMERREN	BUSYEN	_	_	SPITUREN
bit 15							bit 8

R/W-0	R/W-0	R/W-0	U-0	R/W-0	U-0	R/W-0	R/W-0	
SRMTEN	SPIROVEN	SPIRBEN	_	SPITBEN	_	SPITBFEN	SPIRBFEN	
bit 7								

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 15-13 **Unimplemented:** Read as '0'

bit 12 FRMERREN: Enable Interrupt Events via FRMERR bit

1 = Frame error generates an interrupt event

0 = Frame error does not generate an interrupt event

bit 11 **BUSYEN:** Enable Interrupt Events via SPIBUSY bit

1 = SPIBUSY generates an interrupt event

0 = SPIBUSY does not generate an interrupt event

bit 10-9 **Unimplemented:** Read as '0'

bit 8 SPITUREN: Enable Interrupt Events via SPITUR bit

1 = Transmit Underrun (TUR) generates an interrupt event

0 = Transmit Underrun does not generate an interrupt event

bit 7 SRMTEN: Enable Interrupt Events via SRMT bit

1 = Shift Register Empty (SRMT) generates an interrupt events

0 = Shift Register Empty does not generate an interrupt events

bit 6 SPIROVEN: Enable Interrupt Events via SPIROV bit

1 = SPIx receive overflow generates an interrupt event

0 = SPIx receive overflow does not generate an interrupt event

bit 5 SPIRBEN: Enable Interrupt Events via SPIRBE bit

1 = SPIx RX buffer empty generates an interrupt event

0 = SPIx RX buffer empty does not generate an interrupt event

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

bit 3 SPITBEN: Enable Interrupt Events via SPITBE bit

1 = SPIx transmit buffer empty generates an interrupt event

0 = SPIx transmit buffer empty does not generate an interrupt event

bit 2 Unimplemented: Read as '0'

bit 1 SPITBFEN: Enable Interrupt Events via SPITBF bit

1 = SPIx transmit buffer full generates an interrupt event

0 = SPIx transmit buffer full does not generate an interrupt event

bit 0 SPIRBFEN: Enable Interrupt Events via SPIRBF bit

1 = SPIx receive buffer full generates an interrupt event

0 = SPIx receive buffer full does not generate an interrupt event

REGISTER 16-7: SPIXIMSKH: SPIX INTERRUPT MASK REGISTER HIGH

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
RXWIEN	_	RXMSK5 ⁽¹⁾	RXMSK4 ^(1,4)	RXMSK3 ^(1,3)	RXMSK2 ^(1,2)	RXMSK1 ⁽¹⁾	RXMSK0 ⁽¹⁾
bit 15							bit 8

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
TXWIEN	_	TXMSK5 ⁽¹⁾	TXMSK4 ^(1,4)	TXMSK3 ^(1,3)	TXMSK2 ^(1,2)	TXMSK1 ⁽¹⁾	TXMSK0 ⁽¹⁾
bit 7							bit 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 15 RXWIEN: Receive Watermark Interrupt Enable bit

1 = Triggers receive buffer element watermark interrupt when RXMSK<5:0> \leq RXELM<5:0>

0 = Disables receive buffer element watermark interrupt

bit 14 Unimplemented: Read as '0'

bit 13-8 **RXMSK<5:0>:** RX Buffer Mask bits^(1,2,3,4)

RX mask bits; used in conjunction with the RXWIEN bit.

bit 7 **TXWIEN:** Transmit Watermark Interrupt Enable bit

1 = Triggers transmit buffer element watermark interrupt when TXMSK<5:0> = TXELM<5:0>

0 = Disables transmit buffer element watermark interrupt

bit 6 Unimplemented: Read as '0'

bit 5-0 **TXMSK<5:0>:** TX Buffer Mask bits^(1,2,3,4)

TX mask bits; used in conjunction with the TXWIEN bit.

Note 1: Mask values higher than FIFODEPTH are not valid. The module will not trigger a match for any value in this case.

2: RXMSK2 and TXMSK2 bits are only present when FIFODEPTH = 8 or higher.

3: RXMSK3 and TXMSK3 bits are only present when FIFODEPTH = 16 or higher.

4: RXMSK4 and TXMSK4 bits are only present when FIFODEPTH = 32.

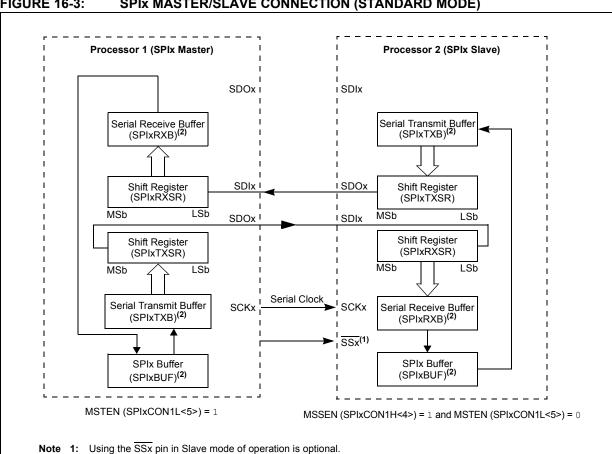
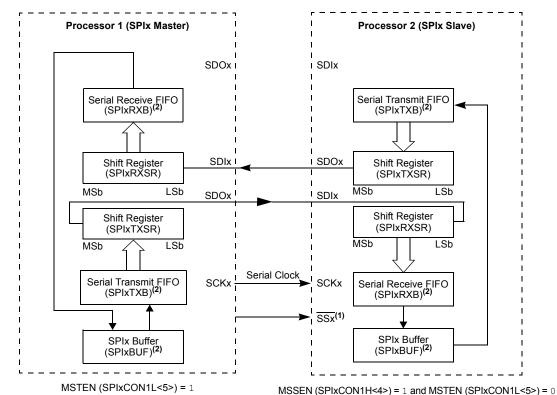


FIGURE 16-3: SPIX MASTER/SLAVE CONNECTION (STANDARD MODE)

2: User must write transmit data to read the received data from SPIxBUF. The SPIxTXB and SPIxRXB registers are memory-mapped to SPIxBUF.

FIGURE 16-4: SPIx MASTER/SLAVE CONNECTION (ENHANCED BUFFER MODES)



Note 1: Using the \overline{SSx} pin in Slave mode of operation is optional.

2: User must write transmit data to read the received data from SPIxBUF. The SPIxTXB and SPIxRXB registers are memory-mapped to SPIxBUF.



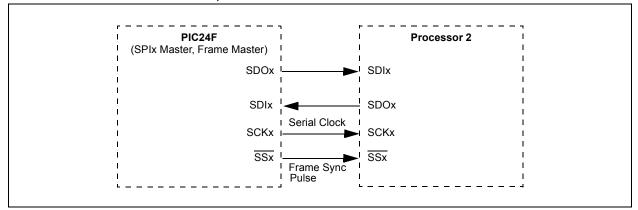


FIGURE 16-6: SPIX MASTER, FRAME SLAVE CONNECTION DIAGRAM

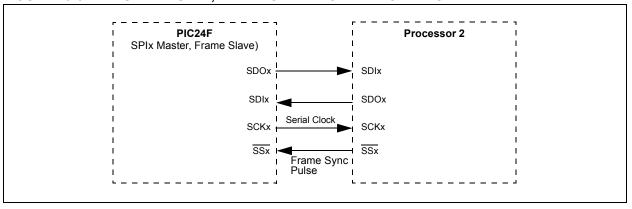


FIGURE 16-7: SPIX SLAVE, FRAME MASTER CONNECTION DIAGRAM

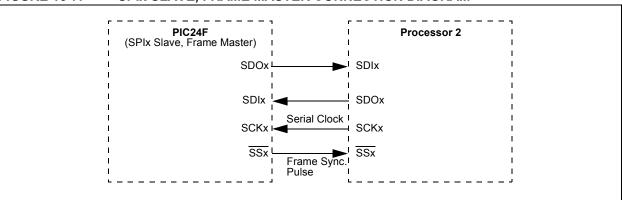
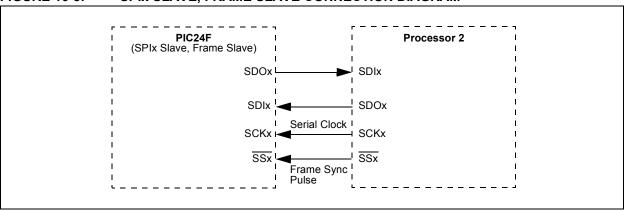


FIGURE 16-8: SPIX SLAVE, FRAME SLAVE CONNECTION DIAGRAM



EQUATION 16-1: RELATIONSHIP BETWEEN DEVICE AND SPIX CLOCK SPEED

$$Baud Rate = \frac{FPB}{(2 * (SPIxBRG + 1))}$$

Where:

FPB is the Peripheral Bus Clock Frequency.

Ν	0	T	Ε	S	

17.0 INTER-INTEGRATED CIRCUIT™ (I²C™)

Note:

This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the "dsPIC33/PIC24 Family Reference Manual", "Inter-Integrated Circuit™ (I²C™)" (DS70000195). The information in this data sheet supersedes the information in the FRM.

The Inter-Integrated CircuitTM (I^2C^{TM}) module is a serial interface useful for communicating with other peripheral or microcontroller devices. These peripheral devices may be serial EEPROMs, display drivers, A/D Converters, etc.

The I²C module supports these features:

- · Independent master and slave logic
- · 7-bit and 10-bit device addresses
- General call address as defined in the I²C protocol
- Clock stretching to provide delays for the processor to respond to a slave data request
- Both 100 kHz and 400 kHz bus specifications
- · Configurable address masking
- Multi-Master modes to prevent loss of messages in arbitration
- Bus Repeater mode, allowing the acceptance of all messages as a slave regardless of the address
- · Automatic SCL

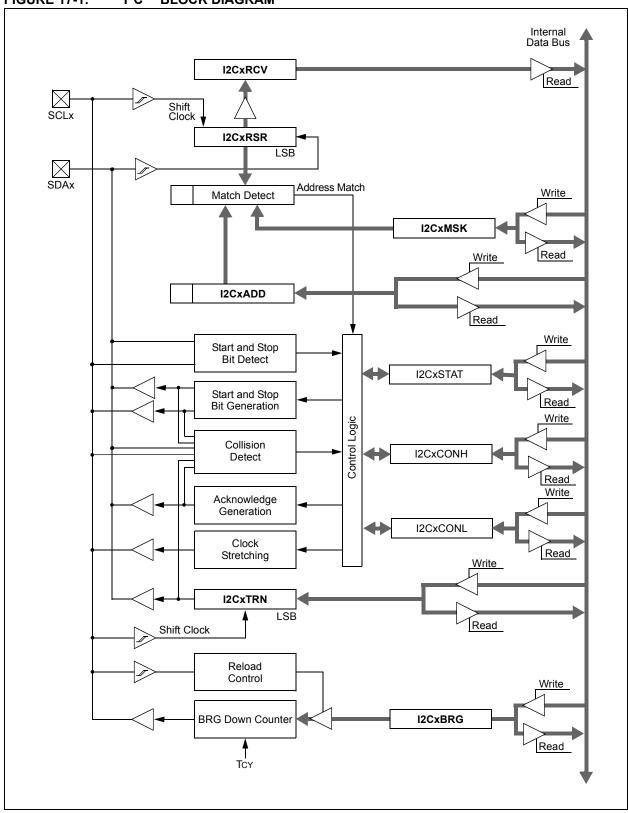
A block diagram of the module is shown in Figure 17-1.

17.1 Communicating as a Master in a Single Master Environment

The details of sending a message in Master mode depends on the communication protocols for the device being communicated with. Typically, the sequence of events is as follows:

- 1. Assert a Start condition on SDAx and SCLx.
- Send the I²C device address byte to the slave with a write indication.
- Wait for and verify an Acknowledge from the slave.
- 4. Send the first data byte (sometimes known as the command) to the slave.
- 5. Wait for and verify an Acknowledge from the
- Send the serial memory address low byte to the slave
- Repeat Steps 4 and 5 until all data bytes are sent
- Assert a Repeated Start condition on SDAx and SCLx.
- 9. Send the device address byte to the slave with a read indication.
- 10. Wait for and verify an Acknowledge from the
- Enable master reception to receive serial memory data.
- 12. Generate an ACK or NACK condition at the end of a received byte of data.
- 13. Generate a Stop condition on SDAx and SCLx.

FIGURE 17-1: I²C™ BLOCK DIAGRAM



17.2 Setting Baud Rate When Operating as a Bus Master

To compute the Baud Rate Generator reload value, use Equation 17-1.

EQUATION 17-1: COMPUTING BAUD RATE RELOAD VALUE⁽¹⁾

$$I2CxBRG = \left(\left(\frac{1}{FSCL} - PGDX\right) \times \frac{FCY}{2}\right) - 2$$

Note 1: Based on Fcy = Fosc/2; Doze mode and PLL are disabled.

17.3 Slave Address Masking

The I2CxMSK register (Register 17-4) designates address bit positions as "don't care" for both 7-Bit and 10-Bit Addressing modes. Setting a particular bit location (= 1) in the I2CxMSK register causes the slave module to respond, whether the corresponding address bit value is a '0' or a '1'. For example, when I2CxMSK is set to '00100000000', the slave module will detect both addresses, '00000000000' and '0010000000'.

To enable address masking, the Intelligent Peripheral Management Interface (IPMI) must be disabled by clearing the STRICT bit (I2CxCONL<11>).

Note: As a result of changes in the I²C™ protocol, the addresses in Table 17-1 are reserved and will not be Acknowledged in Slave mode. This includes any address mask settings that include any of these addresses.

TABLE 17-1: I²C™ RESERVED ADDRESSES⁽¹⁾

Slave Address	R/W Bit	Description					
0000 000	0	General Call Address ⁽²⁾					
0000 000	1	Start Byte					
0000 001	X	Cbus Address					
0000 01x	Х	Reserved					
0000 1xx	X	HS Mode Master Code					
1111 0xx	X	10-Bit Slave Upper Byte ⁽³⁾					
1111 1xx	Х	Reserved					

- Note 1: The address bits listed here will never cause an address match independent of address mask settings.
 - 2: The address will be Acknowledged only if GCEN = 1.
 - 3: A match on this address can only occur on the upper byte in 10-Bit Addressing mode.

REGISTER 17-1: I2CxCONL: I2Cx CONTROL REGISTER LOW

R/W-0	U-0	R/W-0, HC	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0
I2CEN	_	I2CSIDL	SCLREL ⁽¹⁾	STRICT	A10M	DISSLW	SMEN
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0, HC				
GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN
bit 7							bit 0

Legend: HC = Hardware Clearable bit

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 15 **I2CEN:** I2Cx Enable bit (writable from SW only)

1 = Enables the I²C[™] module, and configures the SDAx and SCLx pins as serial port pins

0 = Disables I²C module; all I²C pins are controlled by port functions

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

bit 13 I2CSIDL: I2Cx Stop in Idle Mode bit

1 = Discontinues module operation when device enters Idle mode

0 = Continues module operation in Idle mode

bit 12 SCLREL: SCLx Release Control bit (I²C Slave mode only)⁽¹⁾

Module resets and (I2CEN = 0) sets SCLREL = 1.

If STREN = 0:(2)

1 = Releases clock

0 = Forces clock low (clock stretch)

If STREN = 1:

1 = Releases clock

0 = Holds clock low (clock stretch); user may program this bit to '0', clock stretch is at the next SCLx low

bit 11 STRICT: I2Cx Strict Reserved Address Rule Enable bit

1 = Strict reserved addressing is enforced; for reserved addresses, refer to Table 17-1 (In Slave Mode) – The device doesn't respond to reserved address space and addresses falling in that category are NACKed.

(In Master Mode) – The device is allowed to generate addresses with reserved address space.

0 = Reserved addressing would be Acknowledged

(In Slave Mode) – The device will respond to an address falling in the reserved address space. When there is a match with any of the reserved addresses, the device will generate an ACK.

(In Master Mode) - Reserved.

bit 10 A10M: 10-Bit Slave Address Flag bit

1 = I2CxADD is a 10-bit slave address

0 = I2CADD is a 7-bit slave address

bit 9 DISSLW: Slew Rate Control Disable bit

1 = Slew rate control is disabled for Standard Speed mode (100 kHz, also disabled for 1 MHz mode)

0 = Slew rate control is enabled for High-Speed mode (400 kHz)

bit 8 SMEN: SMBus Input Levels Enable bit

1 = Enables input logic so thresholds are compliant with the SMBus specification

0 = Disables SMBus-specific inputs

Note 1: Automatically cleared to '0' at the beginning of slave transmission; automatically cleared to '0' at the end of slave reception.

2: Automatically cleared to '0' at the beginning of slave transmission.

REGISTER 17-1: I2CxCONL: I2Cx CONTROL REGISTER LOW (CONTINUED)

bit 7 GCEN: General Call Enable bit (I²C Slave mode only)

1 = Enables interrupt when a general call address is received in I2CxRSR; module is enabled for reception

0 = General call address is disabled

bit 6 STREN: SCLx Clock Stretch Enable bit

In I²C Slave mode only; used in conjunction with the SCLREL bit.

1 = Enables clock stretching

0 = Disables clock stretching

bit 5 ACKDT: Acknowledge Data bit

In I²C Master mode, during Master Receive mode – The value that will be transmitted when the user initiates an Acknowledge sequence at the end of a receive.

In I^2C Slave mode when AHEN = 1 or DHEN = 1 – The value that the slave will transmit when it initiates an Acknowledge sequence at the end of an address or data reception.

1 = A NACK is sent

0 = An ACK is sent

bit 4 ACKEN: Acknowledge Sequence Enable bit

In I²C Master mode only; applicable during Master Receive mode.

1 = Initiates Acknowledge sequence on SDAx and SCLx pins, and transmits ACKDT data bit

0 = Acknowledge sequence is Idle

bit 3 **RCEN:** Receive Enable bit (I²C Master mode only)

1 = Enables Receive mode for I²C, automatically cleared by hardware at the end of the 8-bit receive data byte

0 = Receive sequence is not in progress

bit 2 **PEN:** Stop Condition Enable bit (I²C Master mode only)

1 = Initiates Stop condition on the SDAx and SCLx pins

0 = Stop condition is Idle

bit 1 **RSEN:** Restart Condition Enable bit (I²C Master mode only)

1 = Initiates Restart condition on the SDAx and SCLx pins

0 = Restart condition is Idle

bit 0 **SEN:** Start Condition Enable bit (I²C Master mode only)

1 = Initiates Start condition on the SDAx and SCLx pins

0 = Start condition is Idle

Note 1: Automatically cleared to '0' at the beginning of slave transmission; automatically cleared to '0' at the end of slave reception.

2: Automatically cleared to '0' at the beginning of slave transmission.

REGISTER 17-2: I2CxCONH: I2Cx CONTROL REGISTER HIGH

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

U-0	R/W-0						
_	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN
bit 7							bit 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 15-7 **Unimplemented:** Read as '0'

bit 6 **PCIE**: Stop Condition Interrupt Enable bit (I²C[™] Slave mode only).

1 = Enables interrupt on detection of Stop condition

0 = Stop detection interrupts are disabled

bit 5 SCIE: Start Condition Interrupt Enable bit (I²C Slave mode only)

1 = Enables interrupt on detection of Start or Restart conditions

0 = Start detection interrupts are disabled

bit 4 **BOEN:** Buffer Overwrite Enable bit (I²C Slave mode only)

1 = I2CxRCV is updated and an ACK is generated for a received address/data byte, ignoring the state of the I2COV bit only if the RBF bit = 0

0 = I2CxRCV is only updated when I2COV is clear

bit 3 SDAHT: SDAx Hold Time Selection bit

1 = Minimum of 300 ns hold time on SDAx after the falling edge of SCLx

0 = Minimum of 100 ns hold time on SDAx after the falling edge of SCLx

bit 2 **SBCDE:** Slave Mode Bus Collision Detect Enable bit (I²C Slave mode only)

If, on the rising edge of SCLx, SDAx is sampled low when the module is outputting a high state, the BCL bit is set and the bus goes Idle. This Detection mode is only valid during data and ACK transmit sequences.

1 = Enables slave bus collision interrupts

0 = Slave bus collision interrupts are disabled

bit 1 AHEN: Address Hold Enable bit (I²C Slave mode only)

1 = Following the 8th falling edge of SCLx for a matching received address byte; SCLREL bit (I2CxCONL<12>) will be cleared and SCLx will be held low

0 = Address holding is disabled

bit 0 **DHEN:** Data Hold Enable bit (I²C Slave mode only)

1 = Following the 8th falling edge of SCLx for a received data byte; slave hardware clears the SCLREL bit (I2CxCONL<12>) and SCLx is held low

0 = Data holding is disabled

REGISTER 17-3: I2CxSTAT: I2Cx STATUS REGISTER

R-0, HSC	R-0, HSC	R-0, HSC	U-0	U-0	R/C-0, HSC	R-0, HSC	R-0, HSC
ACKSTAT	TRSTAT	ACKTIM	_	_	BCL	GCSTAT	ADD10
bit 15							bit 8

R/C-0, HS	R/C-0, HS	R-0, HSC	R/C-0, HSC	R/C-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC
IWCOL	I2COV	D/ A	Р	S	R/W	RBF	TBF
bit 7							bit 0

Legend: C = Clearable bit		HSC = Hardware Settable/Clearable bit		
R = Readable bit	HS = Hardware Settable bit	U = Unimplemented bit, read as '0'		
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown	

bit 15 ACKSTAT: Acknowledge Status bit (updated in all Master and Slave modes)

1 = Acknowledge was not received from slave

0 = Acknowledge was received from slave

bit 14 **TRSTAT:** Transmit Status bit (when operating as I²C[™] master; applicable to master transmit operation)

1 = Master transmit is in progress (8 bits + ACK)

0 = Master transmit is not in progress

bit 13 **ACKTIM**: Acknowledge Time Status bit (valid in I²C Slave mode only)

1 = Indicates I²C bus is in an Acknowledge sequence, set on 8th falling edge of SCLx clock

0 = Not an Acknowledge sequence, cleared on 9th rising edge of SCLx clock

bit 12-11 **Unimplemented:** Read as '0'

bit 10 BCL: Bus Collision Detect bit (Master/Slave mode; cleared when I²C module is disabled, I2CEN = 0)

1 = A bus collision has been detected during a master or slave transmit operation

0 = No bus collision has been detected

bit 9 GCSTAT: General Call Status bit (cleared after Stop detection)

1 = General call address was received

0 = General call address was not received

bit 8 **ADD10:** 10-Bit Address Status bit (cleared after Stop detection)

1 = 10-bit address was matched

0 = 10-bit address was not matched

bit 7 IWCOL: I2Cx Write Collision Detect bit

1 = An attempt to write to the I2CxTRN register failed because the I²C module is busy; must be cleared in software

0 = No collision

bit 6 I2COV: I2Cx Receive Overflow Flag bit

1 = A byte was received while the I2CxRCV register is still holding the previous byte; I2COV is a

"don't care" in Transmit mode, must be cleared in software

0 = No overflow

bit 5 **D/A:** Data/Address bit (when operating as I²C slave)

1 = Indicates that the last byte received was data

0 = Indicates that the last byte received or transmitted was an address

bit 4 **P:** I2Cx Stop bit

Updated when Start, Reset or Stop is detected; cleared when the I^2C module is disabled, I2CEN = 0.

1 = Indicates that a Stop bit has been detected last

0 = Stop bit was not detected last

REGISTER 17-3: I2CxSTAT: I2Cx STATUS REGISTER (CONTINUED)

bit 3 S: I2Cx Start bit

Updated when Start, Reset or Stop is detected; cleared when the I²C module is disabled, I2CEN = 0.

1 = Indicates that a Start (or Repeated Start) bit has been detected last

0 = Start bit was not detected last

bit 2 **R/W**: Read/Write Information bit (when operating as I²C slave)

1 = Read: Indicates the data transfer is output from the slave

0 = Write: Indicates the data transfer is input to the slave

bit 1 RBF: Receive Buffer Full Status bit

1 = Receive is complete, I2CxRCV is full

0 = Receive is not complete, I2CxRCV is empty

bit 0 TBF: Transmit Buffer Full Status bit

1 = Transmit is in progress, I2CxTRN is full (8 bits of data)

0 = Transmit is complete, I2CxTRN is empty

REGISTER 17-4: I2CxMSK: I2Cx SLAVE MODE ADDRESS MASK REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
_	_	_	_	_	_	MSK<9:8>	
bit 15							bit 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
MSK<7:0>							
bit 7							bit 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 15-10 Unimplemented: Read as '0'

bit 9-0 MSK<9:0>: I2Cx Mask for Address Bit x Select bits

1 = Enables masking for bit x of the incoming message address; bit match is not required in this position

0 = Disables masking for bit x; bit match is required in this position

18.0 UNIVERSAL ASYNCHRONOUS RECEIVER TRANSMITTER (UART)

Note:

This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the "dsPIC33/PIC24 Family Reference Manual", "Universal Asynchronous Receiver Transmitter (UART)" (DS70000582). The information in this data sheet supersedes the information in the FRM.

The Universal Asynchronous Receiver Transmitter (UART) module is one of the serial I/O modules available in the PIC24F device family. The UART is a full-duplex, asynchronous system that can communicate with peripheral devices, such as personal computers, LIN/J2602, RS-232 and RS-485 interfaces. The module also supports a hardware flow control option with the UxCTS and UxRTS pins. The UART module includes the ISO 7816 compliant Smart Card support and the IrDA® encoder/decoder unit.

The PIC24FJ128GB204 family devices are equipped with four UART modules, referred to as UART1, UART2, UART3 and UART4.

The primary features of the UARTx modules are:

- Full-Duplex, 8 or 9-Bit Data Transmission through the UxTX and UxRX Pins
- Even, Odd or No Parity Options (for 8-bit data)
- · One or Two Stop bits
- Hardware Flow Control Option with the UxCTS and UxRTS Pins
- Fully Integrated Baud Rate Generator with 16-Bit Prescaler
- Baud Rates Range from 15 bps to 1 Mbps at 16 MIPS in 16x mode

- Baud Rates Range from 61 bps to 4 Mbps at 16 MIPS in 4x mode
- 4-Deep, First-In-First-Out (FIFO) Transmit Data Buffer
- · 4-Deep FIFO Receive Data Buffer
- Parity, Framing and Buffer Overrun Error Detection
- Support for 9-Bit mode with Address Detect (9th bit = 1)
- Separate Transmit and Receive Interrupts
- · Loopback mode for Diagnostic Support
- · Polarity Control for Transmit and Receive Lines
- · Support for Sync and Break Characters
- · Supports Automatic Baud Rate Detection
- IrDA[®] Encoder and Decoder Logic
- · Includes DMA Support
- 16x Baud Clock Output for IrDA Support
- Smart Card ISO 7816 Support (UART1 and UART2 only):
 - T = 0 protocol with automatic error handling
 - T = 1 protocol
 - Dedicated Guard Time Counter (GTC)
 - Dedicated Waiting Time Counter (WTC)

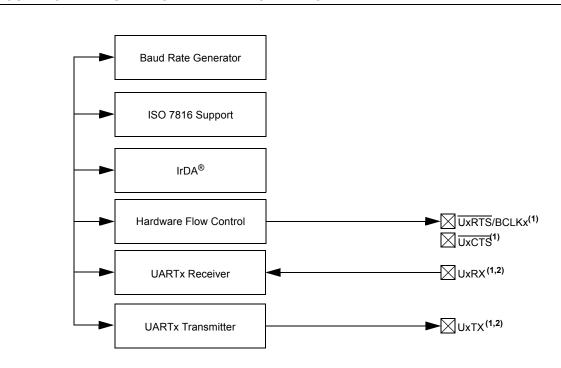
A simplified block diagram of the UARTx module is shown in Figure 18-1. The UARTx module consists of these key important hardware elements:

- · Baud Rate Generator
- · Asynchronous Transmitter
- · Asynchronous Receiver

or UART4.

Note: Throughout this section, references to register and bit names that may be associated with a specific UART module are referred to generically by the use of 'x' in place of the specific module number. Thus, "UxSTA" might refer to the Status register for either UART1, UART2, UART3

FIGURE 18-1: UARTX SIMPLIFIED BLOCK DIAGRAM



- Note 1: The UARTx inputs and outputs must all be assigned to available RPn/RPIn pins before use. See Section 11.4 "Peripheral Pin Select (PPS)" for more information.
 - 2: The UxTX and UxRX pins need to be shorted to be used for the Smart Card interface; this should be taken care of by the user.

18.1 UARTx Baud Rate Generator (BRG)

The UARTx module includes a dedicated, 16-bit Baud Rate Generator. The UxBRG register controls the period of a free-running, 16-bit timer. Equation 18-1 shows the formula for computation of the baud rate with BRGH = 0.

EQUATION 18-1: UARTX BAUD RATE WITH BRGH = $0^{(1,2)}$

$$Baud Rate = \frac{FCY}{16 \cdot (UxBRG + 1)}$$

$$UxBRG = \frac{FCY}{16 \cdot Baud Rate} - 1$$

Note 1: Fcy denotes the instruction cycle clock frequency (Fosc/2).

2: Based on Fcy = Fosc/2; Doze mode and PLL are disabled.

Example 18-1 shows the calculation of the baud rate error for the following conditions:

- Fcy = 4 MHz
- Desired Baud Rate = 9600

The maximum baud rate (BRGH = 0) possible is Fcy/16 (for UxBRG = 0) and the minimum baud rate possible is Fcy/(16 * 65536).

Equation 18-2 shows the formula for computation of the baud rate with BRGH = 1.

EQUATION 18-2: UARTX BAUD RATE WITH BRGH = $1^{(1,2)}$

$$Baud Rate = \frac{FCY}{4 \cdot (UxBRG + 1)}$$

$$UxBRG = \frac{FCY}{4 \cdot Baud Rate} - 1$$

Note 1: Fcy denotes the instruction cycle clock frequency.

2: Based on Fcy = Fosc/2; Doze mode and PLL are disabled.

The maximum baud rate (BRGH = 1) possible is Fcy/4 (for UxBRG = 0) and the minimum baud rate possible is Fcy/(4 * 65536).

Writing a new value to the UxBRG register causes the BRG timer to be reset (cleared). This ensures the BRG does not wait for a timer overflow before generating the new baud rate.

EXAMPLE 18-1: BAUD RATE ERROR CALCULATION (BRGH = 0)⁽¹⁾

Desired Baud Rate = FCY/(16 (UxBRG + 1))

Solving for UxBRG Value:

UxBRG = ((FCY/Desired Baud Rate)/16) - 1

UxBRG = ((4000000/9600)/16) - 1

UxBRG = 25

Calculated Baud Rate = 4000000/(16(25+1))

= 9615

Error = (Calculated Baud Rate – Desired Baud Rate)

Desired Baud Rate = (9615 – 9600)/9600

= 0.16%

Note 1: Based on FCY = FOSC/2; Doze mode and PLL are disabled.

18.2 Transmitting in 8-Bit Data Mode

- 1. Set up the UARTx:
 - a) Write appropriate values for data, parity and Stop bits.
 - b) Write appropriate baud rate value to the UxBRG register.
 - c) Set up transmit and receive interrupt enable and priority bits.
- 2. Enable the UARTx.
- Set the UTXEN bit (causes a transmit interrupt, two cycles after being set).
- Write a data byte to the lower byte of the UxTXREG word. The value will be immediately transferred to the Transmit Shift Register (TSR) and the serial bit stream will start shifting out with the next rising edge of the baud clock.
- 5. Alternatively, the data byte may be transferred while UTXEN = 0 and then, the user may set UTXEN. This will cause the serial bit stream to begin immediately because the baud clock will start from a cleared state.
- A transmit interrupt will be generated as per interrupt control bits, UTXISEL<1:0>.

18.3 Transmitting in 9-Bit Data Mode

- Set up the UARTx (as described in Section 18.2 "Transmitting in 8-Bit Data Mode").
- 2. Enable the UARTx.
- Set the UTXEN bit (causes a transmit interrupt).
- 4. Write UxTXREG as a 16-bit value only.
- A word write to UxTXREG triggers the transfer of the 9-bit data to the TSR. The serial bit stream will start shifting out with the first rising edge of the baud clock.
- 6. A transmit interrupt will be generated as per the setting of control bits, UTXISEL<1:0>.

18.4 Break and Sync Transmit Sequence

The following sequence will send a message frame header, made up of a Break, followed by an auto-baud Sync byte.

- 1. Configure the UARTx for the desired mode.
- Set UTXEN and UTXBRK to set up the Break character.
- 3. Load the UxTXREG with a dummy character to initiate transmission (value is ignored).
- 4. Write '55h' to UxTXREG; this loads the Sync character into the transmit FIFO.
- After the Break has been sent, the UTXBRK bit is reset by hardware. The Sync character now transmits.

18.5 Receiving in 8-Bit or 9-Bit Data Mode

- Set up the UARTx (as described in Section 18.2 "Transmitting in 8-Bit Data Mode").
- 2. Enable the UARTx.
- 3. Set the URXEN bit (UxSTA<12>).
- A receive interrupt will be generated when one or more data characters have been received as per interrupt control bits, URXISEL<1:0>.
- Read the OERR bit to determine if an overrun error has occurred. The OERR bit must be reset in software.
- 6. Read UxRXREG.

The act of reading the UxRXREG character will move the next character to the top of the receive FIFO, including a new set of PERR and FERR values.

18.6 Operation of UxCTS and UxRTS Control Pins

UARTx Clear-to-Send (UxCTS) and Request-to-Send (UxRTS) are the two hardware controlled pins that are associated with the UARTx modules. These two pins allow the UARTx to operate in Simplex and Flow Control mode. They are implemented to control the transmission and reception between the Data Terminal Equipment (DTE). The UEN<1:0> bits in the UxMODE register configure these pins.

18.7 Infrared Support

The UARTx module provides two types of infrared UART support: one is the IrDA clock output to support an external IrDA encoder and decoder device (legacy module support), and the other is the full implementation of the IrDA encoder and decoder. Note that because the IrDA modes require a 16x baud clock, they will only work when the BRGH bit (UxMODE<3>) is '0'.

18.7.1 IrDA CLOCK OUTPUT FOR EXTERNAL IrDA SUPPORT

To support external IrDA encoder and decoder devices, the BCLKx pin (same as the $\overline{\text{UxRTS}}$ pin) can be configured to generate the 16x baud clock. With UEN<1:0> = 11, the BCLKx pin will output the 16x baud clock if the UARTx module is enabled. It can be used to support the IrDA codec chip.

18.7.2 BUILT-IN IrDA ENCODER AND DECODER

The UARTx has full implementation of the IrDA encoder and decoder as part of the UARTx module. The built-in IrDA encoder and decoder functionality is enabled using the IREN bit (UxMODE<12>). When enabled (IREN = 1), the receive pin (UxRX) acts as the input from the infrared receiver. The transmit pin (UxTX) acts as the output to the infrared transmitter.

18.8 Smart Card ISO 7816 Support

Figure 18-2 shows a Smart Card subsystem using a PIC24F microcontroller with a UARTx module for Smart Card data communication. Vcc to power the Smart Card can be supplied through a terminal or an

external power supply. The terminal is also responsible for clocking and resetting the Smart Card. The TX and RX line of the PIC24F device has to be shorted externally, and then connected to the I/O line of the Smart Card.

There are two protocols which are widely used for Smart Card communication between terminal and Smart Card:

- T = 0 (asynchronous, half-duplex, byte-oriented protocol)
- T = 1 (asynchronous, half-duplex, block-oriented protocol)

The selection of the T = 0 or T = 1 protocol is done using the PTRCL bit in the UxSCCON register.

FIGURE 18-2: SMART CARD SUBSYSTEM CONNECTION **TERMINAL** SMART CARD PIC24F I/O UART_IRDA7816⁽²⁾ RX REFO CLK **GPO** RST ATTACHIE(1) Vcc **GND** Driven high upon card insertion. Note 1: Only UART1 and UART2 support Smart Card ISO 7816.

18.9 Control Registers

The UART module consists of the following Special Function Registers (SFRs):

- UxMODE: UARTx Mode Register (Register 18-1)
- UxSTA: UARTx Status and Control Register (Register 18-2)
- UxRXREG: UARTx Receive Register
- UxTXREG: UARTx Transmit Register (Write-Only) (Register 18-3)
- UxADMD: UARTx Address Mask Detect Register (Register 18-4)

- UxBRG: UARTx Baud Rate Register
- UxSCCON: UARTx Smart Card Control Register (Register 18-5)
- UxSCINT: UARTx Smart Card Interrupt Register (Register 18-6)
- · UxGTC: UARTx Guard Time Counter
- UxWTCL and UxWTCH: UARTx Waiting Time Counter Low/High

REGISTER 18-1: UxMODE: UARTX MODE REGISTER

R/W-0	U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
UARTEN ⁽¹⁾	_	USIDL	IREN ⁽²⁾	RTSMD	_	UEN1	UEN0
bit 15							bit 8

R/W-0, HC	R/W-0	R/W-0, HC	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
WAKE	LPBACK	ABAUD	URXINV	BRGH	PDSEL1	PDSEL0	STSEL
bit 7							bit 0

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

- bit 15 **UARTEN:** UARTx Enable bit⁽¹⁾
 - 1 = UARTx is enabled; all UARTx pins are controlled by UARTx as defined by UEN<1:0>
 - 0 = UARTx is disabled; all UARTx pins are controlled by port latches, UARTx power consumption is minimal
- bit 14 **Unimplemented:** Read as '0'
- bit 13 USIDL: UARTx Stop in Idle Mode bit
 - 1 = Discontinues module operation when device enters Idle mode
 - 0 = Continues module operation in Idle mode
- bit 12 IREN: IrDA® Encoder and Decoder Enable bit(2)
 - 1 = IrDA encoder and decoder are enabled
 - 0 = IrDA encoder and decoder are disabled
- bit 11 RTSMD: Mode Selection for UxRTS Pin bit
 - $1 = \overline{\text{UxRTS}}$ pin is in Simplex mode
 - $0 = \overline{\text{UxRTS}}$ pin is in Flow Control mode
- bit 10 **Unimplemented:** Read as '0'
- bit 9-8 **UEN<1:0>:** UARTx Enable bits
 - 11 = UxTX, UxRX and BCLKx pins are enabled and used; UxCTS pin is controlled by port latches
 - 10 = UxTX, UxRX, $\overline{\text{UxCTS}}$ and $\overline{\text{UxRTS}}$ pins are enabled and used
 - 01 = UxTX, UxRX and UxRTS pins are enabled and used; UxCTS pin is controlled by port latches
 - 00 = UxTX and UxRX pins are enabled and used; $\overline{\text{UxCTS}}$ and $\overline{\text{UxRTS}}/\text{BCLKx}$ pins are controlled by port latches
- Note 1: If UARTEN = 1, the peripheral inputs and outputs must be configured to an available RPn/RPIn pin. For more information, see Section 11.4 "Peripheral Pin Select (PPS)".
 - 2: This feature is only available for 16x BRG mode (BRGH = 0).

REGISTER 18-1: UxMODE: UARTx MODE REGISTER (CONTINUED)

- bit 7 WAKE: Wake-up on Start Bit Detect During Sleep Mode Enable bit
 - 1 = UARTx will continue to sample the UxRX pin; interrupt is generated on the falling edge, bit is cleared in hardware on the following rising edge
 - 0 = No wake-up is enabled
- bit 6 LPBACK: UARTx Loopback Mode Select bit
 - 1 = Enables Loopback mode
 - 0 = Loopback mode is disabled
- bit 5 ABAUD: Auto-Baud Enable bit
 - 1 = Enables baud rate measurement on the next character requires reception of a Sync field (55h); cleared in hardware upon completion
 - 0 = Baud rate measurement is disabled or has completed
- bit 4 URXINV: UARTx Receive Polarity Inversion bit
 - 1 = UxRX Idle state is '0'
 - 0 = UxRX Idle state is '1'
- bit 3 BRGH: High Baud Rate Enable bit
 - 1 = High-Speed mode (4 BRG clock cycles per bit)
 - 0 = Standard Speed mode (16 BRG clock cycles per bit)
- bit 2-1 **PDSEL<1:0>:** Parity and Data Selection bits
 - 11 = 9-bit data, no parity
 - 10 = 8-bit data, odd parity
 - 01 = 8-bit data, even parity
 - 00 = 8-bit data, no parity
- bit 0 STSEL: Stop Bit Selection bit
 - 1 = Two Stop bits
 - 0 = One Stop bit
- Note 1: If UARTEN = 1, the peripheral inputs and outputs must be configured to an available RPn/RPIn pin. For more information, see Section 11.4 "Peripheral Pin Select (PPS)".
 - 2: This feature is only available for 16x BRG mode (BRGH = 0).

REGISTER 18-2: UxSTA: UARTX STATUS AND CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0, HC	R/W-0	R-0, HSC	R-1, HSC
UTXISEL1	UTXINV ⁽¹⁾	UTXISEL0	URXEN	UTXBRK	UTXEN ⁽²⁾	UTXBF	TRMT
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R-1, HSC	R-0, HSC	R-0, HSC	R/C-0, HS	R-0, HSC
URXISEL1	URXISEL0	ADDEN	RIDLE	PERR	FERR	OERR	URXDA
bit 7							bit 0

Legend: C = Clearable bit HSC = Hardware Settable/Clearable bit

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

HS = Hardware Settable bit HC = Hardware Clearable bit

- bit 15,13 UTXISEL<1:0>: UARTx Transmission Interrupt Mode Selection bits
 - 11 = Reserved; do not use
 - 10 = Interrupt when a character is transferred to the Transmit Shift Register (TSR), and as a result, the transmit buffer becomes empty
 - 01 = Interrupt when the last character is shifted out of the Transmit Shift Register; all transmit operations are completed
 - 00 = Interrupt when a character is transferred to the Transmit Shift Register (this implies there is at least one character open in the transmit buffer)
- bit 14 UTXINV: IrDA® Encoder Transmit Polarity Inversion bit(1)

IREN = 0:

1 = UxTX is Idle ('0')

0 = UxTX is Idle ('1')

IREN = 1:

1 = UxTX is Idle ('1')

0 = UxTX is Idle ('0')

- bit 12 URXEN: UARTx Receive Enable bit
 - 1 = Receive is enabled, UxRX pin is controlled by UARTx
 - 0 = Receive is disabled, UxRX pin is controlled by the port
- bit 11 UTXBRK: UARTx Transmit Break bit
 - 1 = Sends Sync Break on next transmission Start bit, followed by twelve '0' bits, followed by Stop bit; cleared by hardware upon completion
 - 0 = Sync Break transmission is disabled or has completed
- bit 10 UTXEN: UARTx Transmit Enable bit⁽²⁾
 - 1 = Transmit is enabled, UxTX pin is controlled by UARTx
 - 0 = Transmit is disabled, any pending transmission is aborted and the buffer is reset; UxTX pin is controlled by the port
- bit 9 UTXBF: UARTx Transmit Buffer Full Status bit (read-only)
 - 1 = Transmit buffer is full
 - 0 = Transmit buffer is not full, at least one more character can be written
- bit 8 **TRMT:** Transmit Shift Register Empty bit (read-only)
 - 1 = Transmit Shift Register is empty and transmit buffer is empty (the last transmission has completed)
 - 0 = Transmit Shift Register is not empty, a transmission is in progress or queued
- **Note 1:** The value of this bit only affects the transmit properties of the module when the IrDA[®] encoder is enabled (IREN = 1).
 - 2: If UARTEN = 1, the peripheral inputs and outputs must be configured to an available RPn/RPIn pin. For more information, see Section 11.4 "Peripheral Pin Select (PPS)".

REGISTER 18-2: UXSTA: UARTX STATUS AND CONTROL REGISTER (CONTINUED)

- bit 7-6 URXISEL<1:0>: UARTx Receive Interrupt Mode Selection bits
 - 11 = Interrupt is set on an RSR transfer, making the receive buffer full (i.e., has 4 data characters)
 - 10 = Interrupt is set on an RSR transfer, making the receive buffer 3/4 full (i.e., has 3 data characters)
 - 0x = Interrupt is set when any character is received and transferred from the RSR to the receive buffer; receive buffer has one or more characters
- bit 5 **ADDEN:** Address Character Detect bit (bit 8 of received data = 1)
 - 1 = Address Detect mode is enabled (if 9-bit mode is not selected, this does not take effect)
 - 0 = Address Detect mode is disabled
- bit 4 RIDLE: Receiver Idle bit (read-only)
 - 1 = Receiver is Idle
 - 0 = Receiver is active
- bit 3 PERR: Parity Error Status bit (read-only)
 - 1 = Parity error has been detected for the current character (the character at the top of the receive FIFO)
 - 0 = Parity error has not been detected
- bit 2 **FERR:** Framing Error Status bit (read-only)
 - 1 = Framing error has been detected for the current character (the character at the top of the receive FIFO)
 - 0 = Framing error has not been detected
- bit 1 **OERR:** Receive Buffer Overrun Error Status bit (clear/read-only)
 - 1 = Receive buffer has overflowed
 - 0 = Receive buffer has not overflowed (clearing a previously set OERR bit (1 \rightarrow 0 transition); will reset the receive buffer and the RSR to the empty state)
- bit 0 **URXDA:** UARTX Receive Buffer Data Available bit (read-only)
 - 1 = Receive buffer has data, at least one more character can be read
 - 0 = Receive buffer is empty
- **Note 1:** The value of this bit only affects the transmit properties of the module when the IrDA® encoder is enabled (IREN = 1).
 - 2: If UARTEN = 1, the peripheral inputs and outputs must be configured to an available RPn/RPIn pin. For more information, see Section 11.4 "Peripheral Pin Select (PPS)".

REGISTER 18-3: UXTXREG: UARTX TRANSMIT REGISTER (NORMALLY WRITE-ONLY)

W-x	U-0	U-0	U-0	U-0	U-0	U-0	W-x
LAST ⁽¹⁾	_	_	_	_	_	_	UxTXREG8
bit 15							bit 8

W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x		
UxTXREG<7:0>									
bit 7							bit 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 15 Last Byte Indicator for Smart Card Support bit⁽¹⁾

bit 14-9 Unimplemented: Read as '0'

bit 8 **UxTXREG8:** UARTx Data of the Transmitted Character bit (in 9-bit mode)

bit 7-0 UxTXREG<7:0>: UARTx Data of the Transmitted Character bits

Note 1: This bit is only available for UART1 and UART2.

REGISTER 18-4: UXADMD: UARTX ADDRESS MATCH DETECT REGISTER

| R/W-0 |
|----------|----------|----------|----------|----------|----------|----------|----------|
| ADMMASK7 | ADMMASK6 | ADMMASK5 | ADMMASK4 | ADMMASK3 | ADMMASK2 | ADMMASK1 | ADMMASK0 |
| bit 15 | | | | | | | bit 8 |

| R/W-0 |
|----------|----------|----------|----------|----------|----------|----------|----------|
| ADMADDR7 | ADMADDR6 | ADMADDR5 | ADMADDR4 | ADMADDR3 | ADMADDR2 | ADMADDR1 | ADMADDR0 |
| bit 7 | | | | | | | bit 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 15-8 ADMMASK<7:0>: UARTx ADMADDR<7:0> (UxADMD<7:0>) Masking bits

For ADMMASK<x>:

1 = ADMADDR<x> is used to detect the address match

0 = ADMADDR<x> is not used to detect the address match

bit 7-0 ADMADDR<7:0>: UARTx Address Detect Task Off-Load bits

Used with the ADMMASK<7:0> bits (UxADMD<15:8>) to off-load the task of detecting the address character from the processor during Address Detect mode.

REGISTER 18-5: UXSCCON: UARTX SMART CARD CONTROL REGISTER(1)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	TXRPT1 ⁽²⁾	TXRPT0 ⁽²⁾	CONV	T0PD ⁽²⁾	PTRCL	SCEN
bit 7		•					bit 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 15-6 **Unimplemented:** Read as '0'

bit 5-4 TXRPT<1:0>: Transmit Repeat Selection bits⁽²⁾

11 = Retransmits the error byte four times

10 = Retransmits the error byte three times

01 = Retransmits the error byte twice

00 = Retransmits the error byte once

bit 3 CONV: Logic Convention Selection bit

1 = Inverse logic convention0 = Direct logic convention

bit 2 **TOPD:** Pull-Down Duration for T = 0 Error Handling bit⁽²⁾

1 = 2 ETU

0 = 1 ETU

bit 1 PTRCL: Smart Card Protocol Selection bit

1 = T = 1

0 = T = 0

bit 0 SCEN: Smart Card Mode Enable bit

1 = Smart Card mode is enabled if UARTEN (UxMODE<15>) = 1

0 = Smart Card mode is disabled

Note 1: This register is only available for UART1 and UART2.

2: These bits are applicable to T = 0 only, see PTRCL (UxSCCON<1>).

REGISTER 18-6: UXSCINT: UARTX SMART CARD INTERRUPT REGISTER(1)

U-0	U-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
_	_	RXRPTIF ⁽²⁾	TXRPTIF ⁽²⁾	_	_	WTCIF	GTCIF
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
_	PARIE ⁽²⁾	RXRPTIE ⁽²⁾	TXRPTIE ⁽²⁾	_	_	WTCIE	GTCIE
bit 7							bit 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 15-14 **Unimplemented:** Read as '0'

bit 13 **RXRPTIF:** Receive Repeat Interrupt Flag bit⁽²⁾

1 = Parity error has persisted after the same character has been received five times (four retransmits)

0 = Flag is cleared

bit 12 **TXRPTIF:** Transmit Repeat Interrupt Flag bit⁽²⁾

1 = Line error has been detected after the last retransmit per TXRPT<1:0> (see Register 18-5)

0 = Flag is cleared

bit 11-10 **Unimplemented:** Read as '0'

bit 9 WTCIF: Waiting Time Counter Interrupt Flag bit

1 = Waiting Time Counter has reached 0

0 = Waiting Time Counter has not reached 0

bit 8 **GTCIF:** Guard Time Counter Interrupt Flag bit

1 = Guard Time Counter has reached 0

0 = Guard Time Counter has not reached 0

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

bit 6 **PARIE:** Parity Interrupt Enable bit⁽²⁾

1 = An interrupt is invoked when a character is received with a parity error (see PERR (UxSTA<3>) in

Register 18-2 for the interrupt flag)

0 = Interrupt is disabled

bit 5 **RXRPTIE:** Receive Repeat Interrupt Enable bit⁽²⁾

1 = An interrupt is invoked when a parity error has persisted after the same character has been

received five times (four retransmits)

0 = Interrupt is disabled

bit 4 **TXRPTIE:** Transmit Repeat Interrupt Enable bit⁽²⁾

1 = An interrupt is invoked when a line error is detected after the last retransmit per TXRPT<1:0> has

been completed (see Register 18-5)

0 = Interrupt is disabled

bit 3-2 **Unimplemented:** Read as '0'

bit 1 WTCIE: Waiting Time Counter Interrupt Enable bit

1 = Waiting Time Counter interrupt is enabled

0 = Waiting Time Counter interrupt is disabled

GTCIE: Guard Time Counter Interrupt Enable bit

1 = Guard Time Counter interrupt is enabled0 = Guard Time Counter interrupt is disabled

Note 1: This register is only available for UART1 and UART2.

2: These bits are applicable to T = 0 only, see PTRCL (UxSCCON<1>).

bit 0

19.0 UNIVERSAL SERIAL BUS WITH ON-THE-GO SUPPORT (USB OTG)

Note:

This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the "dsPIC33/PIC24 Family Reference Manual", "USB On-The-Go (OTG)" (DS39721). The information in this data sheet supersedes the information in the FRM.

PIC24FJ128GB204 family devices contain a full-speed and low-speed compatible, On-The-Go (OTG) USB Serial Interface Engine (SIE). The OTG capability allows the device to act either as a USB peripheral device or as a USB embedded host with limited host capabilities. The OTG capability allows the device to dynamically switch from device to host operation using OTG's Host Negotiation Protocol (HNP).

For more details on OTG operation, refer to the "On-The-Go Supplement" to the "USB 2.0 Specification", published by the USB-IF. For more details on USB operation, refer to the "Universal Serial Bus Specification", v2.0.

The USB OTG module offers these features:

- USB functionality in Device and Host modes, and OTG capabilities for application-controlled mode switching
- Software-selectable module speeds of full speed (12 Mbps) or low speed (1.5 Mbps, available in Host mode only)
- Support for all four USB transfer types: control, interrupt, bulk and isochronous
- Sixteen bidirectional endpoints for a total of 32 unique endpoints
- DMA interface for data RAM access
- Queues up to sixteen unique endpoint transfers without servicing
- · Integrated, on-chip USB transceiver
- Integrated VBUS generation with on-chip comparators and boost generation
- Configurations for on-chip bus pull-up and pull-down resistors

A simplified block diagram of the USB OTG module is shown in Figure 19-1.

The USB OTG module can function as a USB peripheral device or as a USB host, and may dynamically switch between Device and Host modes under software control. In either mode, the same data paths and Buffer Descriptors (BDs) are used for the transmission and reception of data.

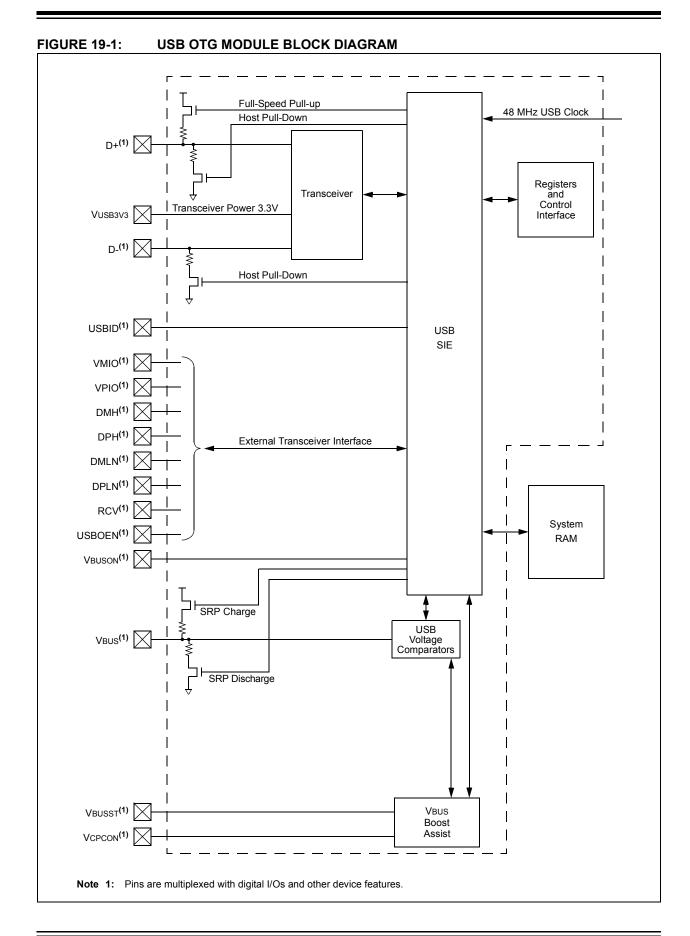
In discussing USB operation, this section will use a controller-centric nomenclature for describing the direction of the data transfer between the microcontroller and the USB. RX (Receive) will be used to describe transfers that move data from the USB to the microcontroller and TX (Transmit) will be used to describe transfers that move data from the microcontroller to the USB. Table 19-1 shows the relationship between data direction in this nomenclature and the USB tokens exchanged.

TABLE 19-1: CONTROLLER-CENTRIC
DATA DIRECTION FOR USB
HOST OR TARGET

USB Mode	Direction				
USB Wode	RX	TX			
Device	OUT or SETUP	IN			
Host	IN	OUT or SETUP			

This chapter presents the most basic operations needed to implement USB OTG functionality in an application. A complete and detailed discussion of the USB protocol and its OTG supplement are beyond the scope of this data sheet. It is assumed that the user already has a basic understanding of USB architecture and the latest version of the protocol.

Not all steps for proper USB operation (such as device enumeration) are presented here. It is recommended that application developers use an appropriate device driver to implement all of the necessary features. Microchip provides a number of application-specific resources, such as USB firmware and driver support. Refer to www.microchip.com/usb for the latest firmware and driver support.



19.1 Hardware Configuration

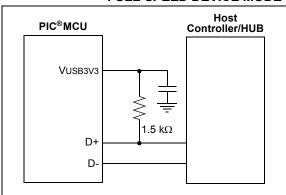
19.1.1 DEVICE MODE

19.1.1.1 D+ Pull-up Resistor

PIC24FJ128GB204 family devices have a built-in 1.5 k Ω resistor on the D+ line that is available when the microcontroller is operating in Device mode. This is used to signal an external host that the device is operating in Full-Speed Device mode. It is engaged by setting the USBEN bit (U1CON<0>). If the OTGEN bit (U1OTGCON<2>) is set, then the D+ pull-up is enabled through the DPPULUP bit (U1OTGCON<7>).

Alternatively, an external resistor may be used on D+, as shown in Figure 19-2.

FIGURE 19-2: EXTERNAL PULL-UP FOR FULL-SPEED DEVICE MODE



19.1.1.2 Power Modes

Many USB applications will likely have several different sets of power requirements and configuration. The most common power modes encountered are:

- Bus Power Only mode
- Self-Power Only mode
- · Dual Power with Self-Power Dominance mode

Bus Power Only mode (Figure 19-3) is effectively the simplest method. All power for the application is drawn from the USB.

To meet the inrush current requirements of the "USB 2.0 Specification", the total effective capacitance appearing across VBUs and ground must be no more than 10 $\mu\text{F}.$

In the USB Suspend mode, devices must consume no more than 2.5 mA from the 5V VBUS line of the USB cable. During the USB Suspend mode, the D+ or D-pull-up resistor must remain active, which will consume some of the allowed suspend current.

In Self-Power Only mode (Figure 19-4), the USB application provides its own power, with very little power being pulled from the USB. Note that an attach indication is added to indicate when the USB has been connected and the host is actively powering VBUS.

To meet compliance specifications, the USB module (and the D+ or D- pull-up resistor) should not be enabled until the host actively drives VBUS high. One of the 5.5V tolerant I/O pins may be used for this purpose.

The application should never source any current onto the 5V VBUS pin of the USB cable.

The Dual Power with Self-Power Dominance mode (Figure 19-5) allows the application to use internal power primarily, but switch to power from the USB when no internal power is available. Dual power devices must also meet all of the special requirements for inrush current and Suspend mode current previously described, and must not enable the USB module until VBUS is driven high.

FIGURE 19-3: BUS POWER ONLY

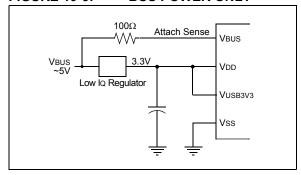


FIGURE 19-4: SELF-POWER ONLY

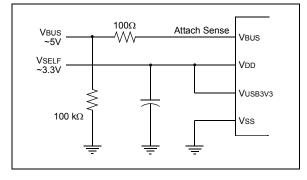
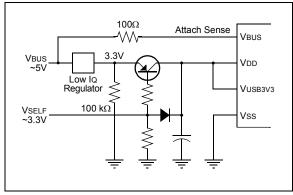


FIGURE 19-5: DUAL POWER EXAMPLE



19.1.2 HOST AND OTG MODES

19.1.2.1 D+ and D- Pull-Down Resistors

PIC24FJ128GB204 family devices have a built-in 15 kΩ pull-down resistor on the D+ and D- lines. These are used in tandem to signal to the bus that the microcontroller is operating in Host mode. They are engaged by setting the HOSTEN bit (U1CON<3>). If the OTGEN bit (U1OTGCON<2>) is set, then these pull-downs are enabled by setting the DPPULDWN and DMPULDWN bits (U1OTGCON<5:4>).

19.1.2.2 Power Configurations

In Host mode, as well as Host mode in On-The-Go operation, the "USB 2.0 Specification" requires that the host application should supply power on VBUS. Since the microcontroller is running below VBUS, and is not able to source sufficient current, a separate power supply must be provided.

When the application is always operating in Host mode, a simple circuit can be used to supply VBUS and regulate current on the bus (Figure 19-6). For OTG operation, it is necessary to be able to turn VBUS on or off as needed, as the microcontroller switches between Device and Host modes. A typical example using an external charge pump is shown in Figure 19-7.

FIGURE 19-6: USB HOST INTERFACE EXAMPLE

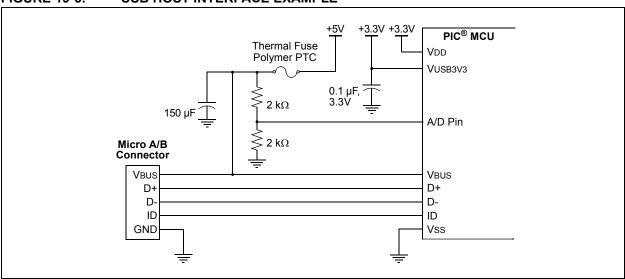
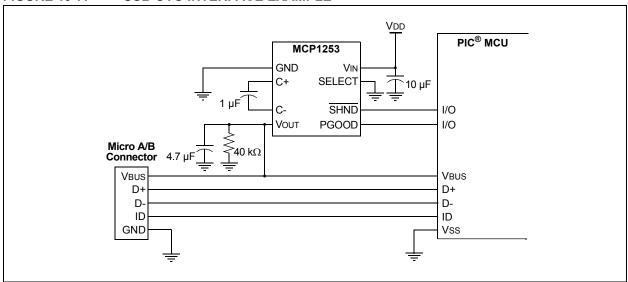


FIGURE 19-7: USB OTG INTERFACE EXAMPLE



19.1.2.3 VBUS Voltage Generation with External Devices

When operating as a USB host, either as an A-device in an OTG configuration or as an embedded host, VBUS must be supplied to the attached device. PIC24FJ128GB204 family devices have an internal VBUS boost assist to help generate the required 5V VBUS from the available voltages on the board. This is comprised of a simple PWM output to control a Switch mode power supply, and built-in comparators to monitor output voltage and limit current.

To enable voltage generation:

- 1. Verify that the USB module is powered (U1PWRC<0> = 1) and that the VBUS discharge is disabled (U1OTGCON<0> = 0).
- Select the desired target voltage using the VBUSCHG bit (U1OTGCON<1>).
- 3. Enable the VBUS generation circuit (U1OTGCON<3> = 1).

Note: This section describes the general process for VBUS voltage generation and control. Please refer to the "dsPIC33/PIC24 Family Reference Manual" for additional examples.

19.1.3 CALCULATING TRANSCEIVER POWER REQUIREMENTS

The USB transceiver consumes a variable amount of current depending on the characteristic impedance of the USB cable, the length of the cable, the VUSB3V3 supply voltage and the actual data patterns moving across the USB cable. Longer cables have larger capacitances and consume more total energy when switching output states. The total transceiver current consumption will be application-specific. Equation 19-1 can help estimate how much current actually may be required in full-speed applications.

Refer to the "dsPIC33/PIC24 Family Reference Manual", "USB On-The-Go (OTG)" (DS39721) for a complete discussion on transceiver power consumption.

EQUATION 19-1: ESTIMATING USB TRANSCEIVER CURRENT CONSUMPTION

$$IXCVR = \frac{40 \text{ } mA \cdot VUSB3V3 \cdot PZERO \cdot PIN \cdot LCABLE}{3.3V \cdot 5m} + IPULLUP$$

Legend: *VUSB3V3* – Voltage applied to the VUSB3V3 pin in volts (3.0V to 3.6V).

PZERO – Percentage (in decimal) of the IN traffic bits sent by the PIC® microcontroller that are a value of '0'.

PIN – Percentage (in decimal) of total bus bandwidth that is used for In traffic.

LCABLE – Length (in meters) of the USB cable. The "USB 2.0 Specification" requires that full-speed applications use cables no longer than 5m.

IPULLUP – Current which the nominal, 1.5 k Ω pull-up resistor (when enabled) must supply to the USB cable.

19.2 USB Buffer Descriptors and the BDT

Endpoint buffer control is handled through a structure called the Buffer Descriptor Table (BDT). This provides a flexible method for users to construct and control endpoint buffers of various lengths and configurations.

The BDT can be located in any available, 512-byte aligned block of data RAM. The BDT Pointer (U1BDTP1) contains the upper address byte of the BDT and sets the location of the BDT in RAM. The user must set this pointer to indicate the table's location.

The BDT is composed of Buffer Descriptors (BDs) which are used to define and control the actual buffers in the USB RAM space. Each BD consists of two, 16-bit "soft" (non-fixed address) registers, BDnSTAT and BDnADR, where n represents one of the 64 possible BDs (range of 0 to 63). BDnSTAT is the status register for BDn, while BDnADR specifies the starting address for the buffer associated with BDn.

Note: Since BDnADR is a 16-bit register, only the first 64 Kbytes of RAM can be

accessed by the USB module.

Depending on the endpoint buffering configuration used, there are up to 64 sets of Buffer Descriptors, for a total of 256 bytes. At a minimum, the BDT must be at least 8 bytes long. This is because the "USB 2.0 Specification" mandates that every device must have Endpoint 0 with both input and output for initial setup.

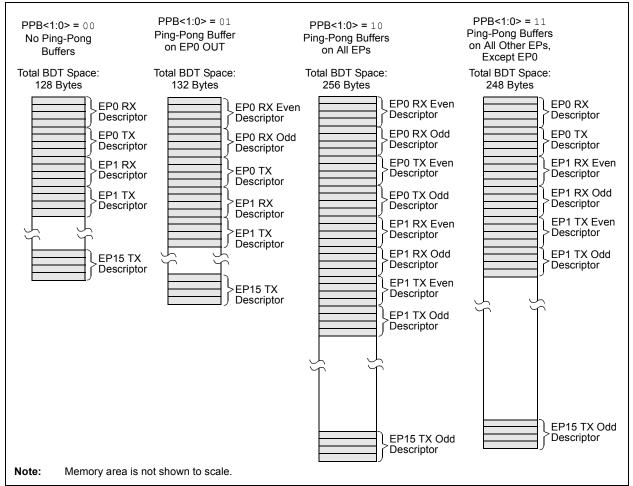
Endpoint mapping in the BDT is dependent on three variables:

- Endpoint number (0 to 15)
- Endpoint direction (RX or TX)
- Ping-pong settings (U1CNFG1<1:0>)

Figure 19-8 illustrates how these variables are used to map endpoints in the BDT.

In Host mode, only Endpoint 0 Buffer Descriptors are used. All transfers utilize the Endpoint 0 Buffer Descriptor and USB Endpoint Control register (U1EP0). For received packets, the attached device's source endpoint is indicated by the value of ENDPT<3:0> in the USB Status register (U1STAT<7:4>). For transmitted packets, the attached device's destination endpoint is indicated by the value written to the USB Token register (U1TOK).

FIGURE 19-8: BDT MAPPING FOR ENDPOINT BUFFERING MODES



BDs have a fixed relationship to a particular endpoint, depending on the buffering configuration. Table 19-2 provides the mapping of BDs to endpoints. This relationship also means that gaps may occur in the BDT if endpoints are not enabled contiguously. This, theoretically, means that the BDs for disabled endpoints could be used as buffer space. In practice, users should avoid using such spaces in the BDT unless a method of validating BD addresses is implemented.

19.2.1 BUFFER OWNERSHIP

Because the buffers and their BDs are shared between the CPU and the USB module, a simple semaphore mechanism is used to distinguish which is allowed to update the BD and associated buffers in memory. This is done by using the UOWN bit as a semaphore to distinguish which is allowed to update the BD and associated buffers in memory. UOWN is the only bit that is shared between the two configurations of BDnSTAT.

When UOWN is clear, the BD entry is "owned" by the microcontroller core. When the UOWN bit is set, the BD entry and the buffer memory are "owned" by the USB peripheral. The core should not modify the BD or its

corresponding data buffer during this time. Note that the microcontroller core can still read BDnSTAT while the SIE owns the buffer and vice versa.

The Buffer Descriptors have a different meaning based on the source of the register update. Register 19-1 and Register 19-2 show the differences in BDnSTAT depending on its current "ownership".

When UOWN is set, the user can no longer depend on the values that were written to the BDs. From this point, the USB module updates the BDs as necessary, overwriting the original BD values. The BDnSTAT register is updated by the SIE with the token PID and the transfer count is updated.

19.2.2 DMA INTERFACE

The USB OTG module uses a dedicated DMA to access both the BDT and the endpoint data buffers. Since part of the address space of the DMA is dedicated to the Buffer Descriptors, a portion of the memory connected to the DMA must comprise a contiguous address space, properly mapped for the access by the module.

TABLE 19-2: ASSIGNMENT OF BUFFER DESCRIPTORS FOR THE DIFFERENT BUFFERING MODES

				BDs Ass	signed to Endpoi	nt		
Endpoint	Mode 0 (No Ping-Pong)		Mode 1 (Ping-Pong on EP0 OUT)		Mod (Ping-Pong		Mode 3 (Ping-Pong on All Other EPs, Except EP0)	
	Out	In	Out	In	Out	In	Out	In
0	0	1	0 (E), 1 (O)	2	0 (E), 1 (O)	2 (E), 3 (O)	0	1
1	2	3	3	4	4 (E), 5 (O)	6 (E), 7 (O)	2 (E), 3 (O)	4 (E), 5 (O)
2	4	5	5	6	8 (E), 9 (O)	10 (E), 11 (O)	6 (E), 7 (O)	8 (E), 9 (O)
3	6	7	7	8	12 (E), 13 (O)	14 (E), 15 (O)	10 (E), 11 (O)	12 (E), 13 (O)
4	8	9	9	10	16 (E), 17 (O)	18 (E), 19 (O)	14 (E), 15 (O)	16 (E), 17 (O)
5	10	11	11	12	20 (E), 21 (O)	22 (E), 23 (O)	18 (E), 19 (O)	20 (E), 21 (O)
6	12	13	13	14	24 (E), 25 (O)	26 (E), 27 (O)	22 (E), 23 (O)	24 (E), 25 (O)
7	14	15	15	16	28 (E), 29 (O)	30 (E), 31 (O)	26 (E), 27 (O)	28 (E), 29 (O)
8	16	17	17	18	32 (E), 33 (O)	34 (E), 35 (O)	30 (E), 31 (O)	32 (E), 33 (O)
9	18	19	19	20	36 (E), 37 (O)	38 (E), 39 (O)	34 (E), 35 (O)	36 (E), 37 (O)
10	20	21	21	22	40 (E), 41 (O)	42 (E), 43 (O)	38 (E), 39 (O)	40 (E), 41 (O)
11	22	23	23	24	44 (E), 45 (O)	46 (E), 47 (O)	42 (E), 43 (O)	44 (E), 45 (O)
12	24	25	25	26	48 (E), 49 (O)	50 (E), 51 (O)	46 (E), 47 (O)	48 (E), 49 (O)
13	26	27	27	28	52 (E), 53 (O)	54 (E), 55 (O)	50 (E), 51 (O)	52 (E), 53 (O)
14	28	29	29	30	56 (E), 57 (O)	58 (E), 59 (O)	54 (E), 55 (O)	56 (E), 57 (O)
15	30	31	31	32	60 (E), 61 (O)	62 (E), 63 (O)	58 (E), 59 (O)	60 (E), 61 (O)

Legend: (E) = Even transaction buffer, (O) = Odd transaction buffer

REGISTER 19-1: BDnSTAT: BUFFER DESCRIPTOR n STATUS REGISTER PROTOTYPE, USB MODE (BD0STAT THROUGH BD63STAT)

R/W-x	R/W-x	R/W-x, HSC					
UOWN	DTS	PID3	PID2	PID1	PID0	BC9	BC8
bit 15							bit 8

| R/W-x, HSC |
|------------|------------|------------|------------|------------|------------|------------|------------|
| BC7 | BC6 | BC5 | BC4 | BC3 | BC2 | BC1 | BC0 |
| bit 7 | | | | | | | bit 0 |

Legend:	HSC = Hardware Settable/Clearable bit					
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 15 UOWN: USB Own bit

1 = The USB module owns the BD and its corresponding buffer; the CPU must not modify the BD or the buffer

bit 14 DTS: Data Toggle Packet bit

1 = Data 1 packet
0 = Data 0 packet

bit 13-10 **PID<3:0>:** Packet Identifier bits (written by the USB module)

In Device Mode:

Represents the PID of the received token during the last transfer.

In Host Mode:

Represents the last returned PID or the transfer status indicator.

bit 9-0 **BC<9:0>:** Byte Count bits

This represents the number of bytes to be transmitted or the maximum number of bytes to be received during a transfer. Upon completion, the byte count is updated by the USB module with the actual number of bytes transmitted or received.

REGISTER 19-2: BDnSTAT: BUFFER DESCRIPTOR n STATUS REGISTER PROTOTYPE, CPU MODE (BD0STAT THROUGH BD63STAT)

R/W-x	R/W-x	r-0	r-0	R/W-x	R/W-x	R/W-x, HSC	R/W-x, HSC
UOWN	DTS ⁽¹⁾	_	_	DTSEN	BSTALL	BC9	BC8
bit 15							bit 8

| R/W-x, HSC |
|------------|------------|------------|------------|------------|------------|------------|------------|
| BC7 | BC6 | BC5 | BC4 | BC3 | BC2 | BC1 | BC0 |
| bit 7 | | | | | | | bit 0 |

Legend:	r = Reserved bit	HSC = Hardware Settable/Clearable bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'		
-n = Value at POR	'1' = Bit is set	'r' = Reserved bit	x = Bit is unknown	

bit 15 UOWN: USB Own bit

0 = The microcontroller core owns the BD and its corresponding buffer; the USB module ignores all other fields in the BD

bit 14 DTS: Data Toggle Packet bit⁽¹⁾

1 = Data 1 packet 0 = Data 0 packet

bit 13-12 **Reserved:** Maintain as '0'

bit 11 DTSEN: Data Toggle Synchronization Enable bit

1 = Data toggle synchronization is enabled; data packets with incorrect Sync value will be ignored

0 = No data toggle synchronization is performed

bit 10 BSTALL: Buffer STALL Enable bit

1 = Buffer STALL is enabled; STALL handshake is issued if a token is received that would use the BD in the given location (UOWN bit remains set, BD value is unchanged); corresponding EPSTALL bit will get set on any STALL handshake

0 = Buffer STALL is disabled

bit 9-0 **BC<9:0>:** Byte Count bits

This represents the number of bytes to be transmitted or the maximum number of bytes to be received during a transfer. Upon completion, the byte count is updated by the USB module with the actual number of bytes transmitted or received.

Note 1: This bit is ignored unless DTSEN = 1.

19.3 USB Interrupts

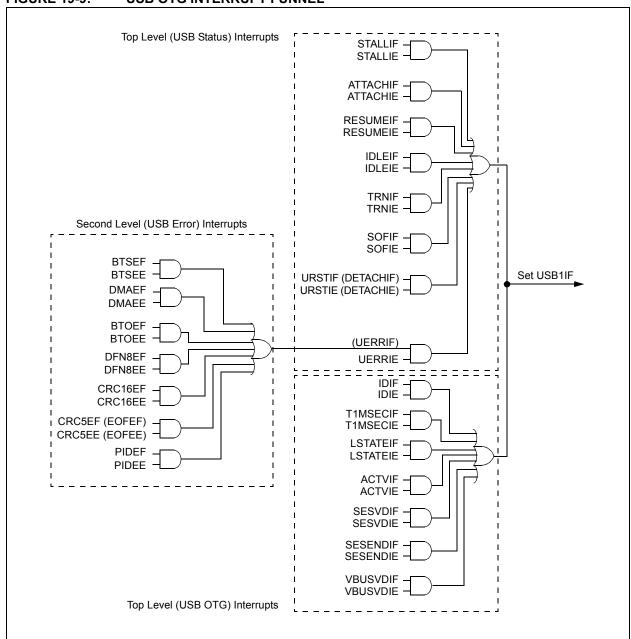
The USB OTG module has many conditions that can be configured to cause an interrupt. All interrupt sources use the same interrupt vector.

Figure 19-9 shows the interrupt logic for the USB module. There are two layers of interrupt registers in the USB module. The top level consists of overall USB status interrupts; these are enabled and flagged in the U1IE and U1IR registers, respectively. The second

level consists of USB error conditions, which are enabled and flagged in the U1EIR and U1EIE registers. An interrupt condition in any of these triggers a USB Error Interrupt Flag (UERRIF) in the top level.

Interrupts may be used to trap routine events in a USB transaction. Figure 19-10 provides some common events within a USB frame and their corresponding interrupts.

FIGURE 19-9: USB OTG INTERRUPT FUNNEL



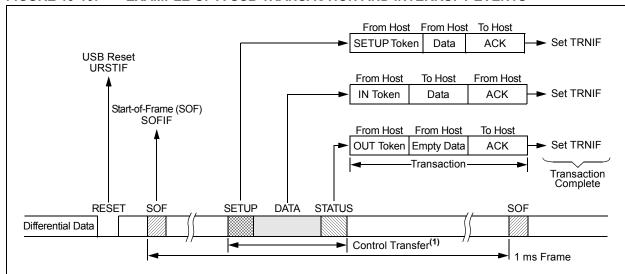
19.3.1 CLEARING USB OTG INTERRUPTS

Unlike device-level interrupts, the USB OTG interrupt status flags are not freely writable in software. All USB OTG flag bits are implemented as hardware settable only bits. Additionally, these bits can only be cleared in

software by writing a '1' to their locations (i.e., performing a MOV type instruction). Writing a '0' to a flag bit (i.e., a BCLR instruction) has no effect.

Note: Throughout this data sheet, a bit that can only be cleared by writing a '1' to its location is referred to as: "Write '1' to clear". In register descriptions, this function is indicated by the descriptor, "K".

FIGURE 19-10: EXAMPLE OF A USB TRANSACTION AND INTERRUPT EVENTS



Note 1: The control transfer shown here is only an example showing events that can occur for every transaction. Typical control transfers will spread across multiple frames.

19.4 Device Mode Operation

The following section describes how to perform a common Device mode task. In Device mode, USB transfers are performed at the transfer level. The USB module automatically performs the status phase of the transfer.

19.4.1 ENABLING DEVICE MODE

- Reset the Ping-Pong Buffer Pointers by setting, then clearing, the Ping-Pong Buffer Reset bit, PPBRST (U1CON<1>).
- 2. Disable all interrupts (U1IE and U1EIE = 00h).
- Clear any existing interrupt flags by writing FFh to U1IR and U1EIR.
- Verify that VBUs is present (non-OTG devices only).

- Enable the USB module by setting the USBEN bit (U1CON<0>).
- Set the OTGEN bit (U1OTGCON<2>) to enable OTG operation.
- 7. Enable the Endpoint 0 buffer to receive the first setup packet by setting the EPRXEN and EPHSHK bits for Endpoint 0 (U1EP0<3,0> = 1).
- Power up the USB module by setting the USBPWR bit (U1PWRC<0>).
- 9. Enable the D+ pull-up resistor to signal an attach by setting the DPPULUP bit (U1OTGCON<7>).

19.4.2 RECEIVING AN IN TOKEN IN DEVICE MODE

- Attach to a USB host and enumerate as described in Chapter 9 of the "USB 2.0 Specification".
- 2. Create a data buffer and populate it with the data to send to the host.
- In the appropriate (Even or Odd) TX BD for the desired endpoint:
 - Set up the status register (BDnSTAT) with the correct data toggle (DATA0/1) value and the byte count of the data buffer.
 - b) Set up the address register (BDnADR) with the starting address of the data buffer.
 - c) Set the UOWN bit of the status register to '1'.
- When the USB module receives an IN token, it automatically transmits the data in the buffer. Upon completion, the module updates the status register (BDnSTAT) and sets the Transfer Done Interrupt Flag, TRNIF (U1IR<3>).

19.4.3 RECEIVING AN OUT TOKEN IN DEVICE MODE

- Attach to a USB host and enumerate as described in Chapter 9 of the "USB 2.0 Specification".
- 2. Create a data buffer with the amount of data you are expecting from the host.
- In the appropriate (Even or Odd) TX BD for the desired endpoint:
 - Set up the status register (BDnSTAT) with the correct data toggle (DATA0/1) value and the byte count of the data buffer.
 - b) Set up the address register (BDnADR) with the starting address of the data buffer.
 - c) Set the UOWN bit of the status register to '1'.
- When the USB module receives an OUT token, it automatically receives the data sent by the host to the buffer. Upon completion, the module updates the status register (BDnSTAT) and sets the Transfer Done Interrupt Flag, TRNIF (U1IR<3>).

19.5 Host Mode Operation

The following sections describe how to perform common Host mode tasks. In Host mode, USB transfers are invoked explicitly by the host software. The host software is responsible for the Acknowledge portion of the transfer. Also, all transfers are performed using the USB Endpoint 0 Control register (U1EP0) and Buffer Descriptors.

19.5.1 ENABLE HOST MODE AND DISCOVER A CONNECTED DEVICE

- Enable Host mode by setting the HOSTEN bit (U1CON<3>). This causes the Host mode control bits in other USB OTG registers to become available.
- Enable the D+ and D- pull-down resistors by setting the DPPULDWN and DMPULDWN bits (U1OTGCON<5:4>). Disable the D+ and Dpull-up resistors by clearing the DPPULUP and DMPULUP bits (U1OTGCON<7:6>).
- At this point, Start-of-Frame (SOF) generation begins with the SOF counter loaded with 12,000. Eliminate noise on the USB by clearing the SOFEN bit (U1CON<0>) to disable Start-of-Frame packet generation.
- 4. Enable the device attached interrupt by setting the ATTACHIE bit (U1IE<6>).
- 5. Wait for the device attached interrupt (U1IR<6> = 1). This is signaled by the USB device changing the state of D+ or D- from '0' to '1' (SE0 to J-state). After it occurs, wait 100 ms for the device power to stabilize.
- 6. Check the state of the JSTATE and SE0 bits in U1CON. If the JSTATE bit (U1CON<7>) is '0', the connecting device is low speed. If the connecting device is low speed, set the LSPDEN and LSPD bits (U1ADDR<7> and U1EP0<7>, respectively) to enable low-speed operation.
- Reset the USB device by setting the USBRST bit (U1CON<4>) for at least 50 ms, sending Reset signaling on the bus. After 50 ms, terminate the Reset by clearing USBRST.
- 8. In order to keep the connected device from going into suspend, enable the SOF packet generation by setting the SOFEN bit.
- 9. Wait 10 ms for the device to recover from Reset.
- Perform enumeration as described by Chapter 9 of the "USB 2.0 Specification".

19.5.2 COMPLETE A CONTROL TRANSACTION TO A CONNECTED DEVICE

- Follow the procedure described in Section 19.5.1
 "Enable Host Mode and Discover a Connected Device" to discover a device.
- Set up the Endpoint Control register for bidirectional control transfers by writing 0Dh to U1EP0 (this sets the EPCONDIS, EPTXEN and EPHSHK bits).
- Place a copy of the device framework setup command in a memory buffer. See Chapter 9 of the "USB 2.0 Specification" for information on the device framework command set.
- Initialize the Buffer Descriptor (BD) for the current (Even or Odd) TX EP0 to transfer the eight bytes of command data for a device framework command (i.e., GET DEVICE DESCRIPTOR):
 - a) Set the BD Data Buffer Address (BD0ADR) to the starting address of the 8-byte memory buffer containing the command.
 - b) Write 8008h to BD0STAT (this sets the UOWN bit and sets a byte count of 8).
- Set the USB device address of the target device in the USB Address register (U1ADDR<6:0>).
 After a USB bus Reset, the device USB address will be zero. After enumeration, it will be set to another value, between 1 and 127.
- 6. Write D0h to U1TOK; this is a SETUP token to Endpoint 0, the target device's default control pipe. This initiates a SETUP token on the bus, followed by a data packet. The device handshake is returned in the PID field of BD0STAT after the packets are complete. When the USB module updates BD0STAT, a Transfer Done Interrupt Flag is asserted (the TRNIF flag is set). This completes the setup phase of the setup transaction, as referenced in Chapter 9 of the "USB 2.0 Specification".
- 7. To initiate the data phase of the setup transaction (i.e., get the data for the GET DEVICE DESCRIPTOR command), set up a buffer in memory to store the received data.

- 8. Initialize the current (Even or Odd) RX or TX (RX for IN, TX for OUT) EP0 BD to transfer the data.
 - a) Write C040h to BD0STAT. This sets the UOWN, configures Data Toggle Packet (DTS) to DATA1 and sets the byte count to the length of the data buffer (64 or 40h in this case).
 - Set BD0ADR to the starting address of the data buffer.
- 9. Write the USB Token register with the appropriate IN or OUT token to Endpoint 0, the target device's default control pipe (e.g., write 90h to U1TOK for an IN token for a GET DEVICE DESCRIPTOR command). This initiates an IN token on the bus, followed by a data packet from the device to the host. When the data packet completes, the BD0STAT is written and a Transfer Done Interrupt Flag is asserted (the TRNIF flag is set). For control transfers with a single packet data phase, this completes the data phase of the setup transaction, as referenced in Chapter 9 of the "USB 2.0 Specification". If more data needs to be transferred, return to Step 8.
- 10. To initiate the status phase of the setup transaction, set up a buffer in memory to receive or send the zero length status phase data packet.
- 11. Initialize the current (Even or Odd) TX EP0 BD to transfer the status data:
 - a) Set the BDT buffer address field to the starting address of the data buffer.
 - b) Write 8000h to BD0STAT (set UOWN bit, configure DTS to DATA0 and set byte count to 0).
- 12. Write the USB Token register with the appropriate IN or OUT token to Endpoint 0, the target device's default control pipe (e.g., write 01h to U1TOK for an OUT token for a GET DEVICE DESCRIPTOR command). This initiates an OUT token on the bus, followed by a zero length data packet from the host to the device. When the data packet completes, the BD is updated with the handshake from the device and a Transfer Done interrupt Flag is asserted (the TRNIF flag is set). This completes the status phase of the setup transaction as described in Chapter 9 of the "USB 2.0 Specification".

Note: Only one control transaction can be performed per frame.

19.5.3 SEND A FULL-SPEED BULK DATA TRANSFER TO A TARGET DEVICE

- Follow the procedure described in Section 19.5.1
 "Enable Host Mode and Discover a Connected
 Device" and Section 19.5.2 "Complete a Control Transaction to a Connected Device" to
 discover and configure a device.
- To enable transmit and receive transfers with handshaking enabled, write 1Dh to U1EP0. If the target device is a low-speed device, also set the LSPD (U1EP0<7>) bit. If you want the hardware to automatically retry indefinitely, if the target device asserts a NAK on the transfer, clear the Retry Disable bit, RETRYDIS (U1EP0<6>).
- 3. Set up the BD for the current (Even or Odd) TX EP0 to transfer up to 64 bytes.
- 4. Set the USB device address of the target device in the USB Address register (U1ADDR<6:0>).
- Write an OUT token to the desired endpoint to U1TOK. This triggers the module's transmit state machines to begin transmitting the token and the data.
- 6. Wait for the Transfer Done Interrupt Flag, TRNIF. This indicates that the BD has been released back to the microprocessor and the transfer has completed. If the Retry Disable bit (RETRYDIS) is set, the handshake (ACK, NAK, STALL or ERROR (0Fh)) is returned in the BD PID field. If a STALL interrupt occurs, the pending packet must be dequeued and the error condition in the target device cleared. If a detach interrupt occurs (SE0 for more than 2.5 μs), then the target has detached (U1IR<0> is set).
- 7. Once the Transfer Done Interrupt Flag occurs (TRNIF is set), the BD can be examined and the next data packet queued by returning to Step 2.

Note: USB speed, transceiver and pull-ups should only be configured during the module setup phase. It is not recommended to change these settings while the module is enabled.

19.6 OTG Operation

19.6.1 SESSION REQUEST PROTOCOL (SRP)

An OTG A-device may decide to power down the VBUS supply when it is not using the USB link through the Session Request Protocol (SRP). Software may do this by clearing VBUSON (U1OTGCON<3>). When the VBUS supply is powered down, the A-device is said to have ended a USB session.

An OTG A-device or embedded host may repower the VBUS supply at any time (initiate a new session). An OTG B-device may also request that the OTG A-device repower the VBUS supply (initiate a new session). This is accomplished via the Session Request Protocol (SRP).

Prior to requesting a new session, the B-device must first check that the previous session has definitely ended. To do this, the B-device must check for two conditions:

- 1. VBUS supply is below the session valid voltage.
- 2. Both D+ and D- have been low for at least 2 ms.

The B-device will be notified of Condition 1 by the SESENDIF (U10TGIR<2>) interrupt. Software will have to manually check for Condition 2.

Note: When the A-device powers down the VBUS supply, the B-device must disconnect its pull-up resistor from power. If the device is self-powered, it can do this by clearing DPPULUP (U10TGCON<7>) and DMPULUP (U10TGCON<6>).

The B-device may aid in achieving Condition 1 by discharging the VBUS supply through a resistor. Software may do this by setting VBUSDIS (U1OTGCON<0>).

After these initial conditions are met, the B-device may begin requesting the new session. The B-device begins by pulsing the D+ data line. Software should do this by setting DPPULUP (U1OTGCON<7>). The data line should be held high for 5 to 10 ms.

The B-device then proceeds by pulsing the VBUS supply. Software should do this by setting PUVBUS (U1CNFG2<4>). When an A-device detects SRP signaling (either via the ATTACHIF (U1IR<6>) interrupt or via the SESVDIF (U1OTGIR<3>) interrupt), the A-device must restore the VBUS supply by either setting VBUSON (U1OTGCON<3>) or by setting the I/O port controlling the external power source.

The B-device should not monitor the state of the VBUS supply while performing VBUS supply pulsing. When the B-device does detect that the VBUS supply has been restored (via the SESVDIF (U1OTGIR<3>) interrupt), the B-device must reconnect to the USB link by pulling up D+ or D- (via the DPPULUP or DMPULUP bits).

The A-device must complete the SRP by driving USB Reset signaling.

19.6.2 HOST NEGOTIATION PROTOCOL (HNP)

In USB OTG applications, a Dual Role Device (DRD) is a device that is capable of being either a host or a peripheral. Any OTG DRD must support Host Negotiation Protocol (HNP).

HNP allows an OTG B-device to temporarily become the USB host. The A-device must first enable the B-device to follow HNP. Refer to the "On-The-Go Supplement" to the "USB 2.0 Specification" for more information regarding HNP. HNP may only be initiated at full speed.

After being enabled for HNP by the A-device, the B-device requests being the host any time that the USB link is in suspend state by simply indicating a disconnect. This can be done in software by clearing DPPULUP and DMPULUP. When the A-device detects the disconnect condition (via the URSTIF (U1IR<0>) interrupt), the A-device may allow the B-device to take over as host. The A-device does this by signaling connect as a full-speed function. Software may accomplish this by setting DPPULUP.

If the A-device responds instead with resume signaling, the A-device remains as host. When the B-device detects the connect condition (via ATTACHIF), the B-device becomes host. The B-device drives Reset signaling prior to using the bus.

When the B-device has finished in its role as host, it stops all bus activity and turns on its D+ pull-up resistor by setting DPPULUP. When the A-device detects a suspend condition (Idle for 3 ms), the A-device turns off its D+ pull-up. The A-device may also power down the VBUS supply to end the session. When the A-device detects the connect condition (via ATTACHIF), the A-device resumes host operation and drives Reset signaling.

19.7 USB OTG Module Registers

There are a total of 37 memory-mapped registers associated with the USB OTG module. They can be divided into four general categories:

- · USB OTG Module Control (12)
- USB Interrupt (7)
- USB Endpoint Management (16)
- USB VBUS Power Control (2)

This total does not include the (up to) 128 BD registers in the BDT. Their prototypes, described in Register 19-1 and Register 19-2, are shown separately in Section 19.2 "USB Buffer Descriptors and the BDT".

All USB OTG registers are implemented in the Least Significant Byte of the register. Bits in the upper byte are unimplemented and have no function. Note that some registers are instantiated only in Host mode, while other registers have different bit instantiations and functions in Device and Host modes.

The registers described in the following sections are those that have bits with specific control and configuration features. The following registers are used for data or address values only:

- U1BDTP1: Specifies the 256-word page in data RAM used for the BDT; 8-bit value with bit 0 fixed as '0' for boundary alignment.
- U1FRML and U1FRMH: Contain the 11-bit byte counter for the current data frame.

19.7.1 USB OTG MODULE CONTROL REGISTERS

REGISTER 19-3: U10TGSTAT: USB OTG STATUS REGISTER (HOST MODE ONLY)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15		_	_		_		bit 8

R-0, HSC	U-0	R-0, HSC	U-0	R-0, HSC	R-0, HSC	U-0	R-0, HSC
ID	_	LSTATE	_	SESVD	SESEND	_	VBUSVD
bit 7							bit 0

Legend: U = Unimplemented bit, read as '0'

R = Readable bit W = Writable bit HSC = Hardware Settable/Clearable bit

-n = Value at POR '1' = Bit is set '0' = Bit is cleared x = Bit is unknown

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

bit 7 ID: ID Pin State Indicator bit

1 = No plug is attached or a Type B cable has been plugged into the USB receptacle

0 = A Type A plug has been plugged into the USB receptacle

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

bit 5 LSTATE: Line State Stable Indicator bit

1 = The USB line state (as defined by SE0 and JSTATE) has been stable for the previous 1 ms

0 = The USB line state has not been stable for the previous 1 ms

bit 4 Unimplemented: Read as '0'

bit 3 SESVD: Session Valid Indicator bit

1 = The VBUS voltage is above VA_SESS_VLD (as defined in the "USB 2.0 Specification") on the A or

B-device

0 = The VBUS voltage is below VA_SESS_VLD on the A or B-device

bit 2 SESEND: B Session End Indicator bit

1 = The VBUS voltage is below VB_SESS_END (as defined in the "USB 2.0 Specification") on the B-device

0 = The VBUS voltage is above VB SESS END on the B-device

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

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

1 = The VBUS voltage is above VA_VBUS_VLD (as defined in the "USB 2.0 Specification") on the A-device

0 = The VBUS voltage is below VA VBUS VLD on the A-device

REGISTER 19-4: U10TGCON: USB OTG CONTROL REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 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
DPPULUP	DMPULUP	DPPULDWN ⁽¹⁾	DMPULDWN ⁽¹⁾	VBUSON ⁽¹⁾	OTGEN ⁽¹⁾	VBUSCHG ⁽¹⁾	VBUSDIS ⁽¹⁾
bit 7							bit 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 15-8 **Unimplemented:** Read as '0' bit 7 **DPPULUP:** D+ Pull-up Enable bit

1 = D+ data line pull-up resistor is enabled 0 = D+ data line pull-up resistor is disabled

bit 6 **DMPULUP:** D- Pull-up Enable bit

1 = D- data line pull-up resistor is enabled 0 = D- data line pull-up resistor is disabled

bit 5 **DPPULDWN:** D+ Pull-Down Enable bit⁽¹⁾

1 = D+ data line pull-down resistor is enabled 0 = D+ data line pull-down resistor is disabled

bit 4 **DMPULDWN:** D- Pull-Down Enable bit⁽¹⁾

DIVIPOLDANIA. D-1 uli-Down Enable bit.

1 = D- data line pull-down resistor is enabled 0 = D- data line pull-down resistor is disabled

bit 3 **VBUSON:** VBUS Power-on bit⁽¹⁾

1 = VBUS line is powered

0 = VBUS line is not powered

bit 2 **OTGEN:** OTG Features Enable bit⁽¹⁾

1 = USB OTG is enabled; all D+/D- pull-up and pull-down bits are enabled

0 = USB OTG is disabled; D+/D- pull-up and pull-down bits are controlled in hardware by the settings of the HOSTEN and USBEN (U1CON<3,0>) bits

bit 1 VBUSCHG: VBUS Charge Select bit (1)

1 = VBUS line is set to charge to 3.3V

0 = VBUS line is set to charge to 5V

bit 0 VBUSDIS: VBUS Discharge Enable bit(1)

1 = VBUS line is discharged through a resistor

0 = VBUS line is not discharged

Note 1: These bits are only used in Host mode; do not use in Device mode.

REGISTER 19-5: U1PWRC: USB POWER CONTROL REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

R/W-0, HS	U-0	U-0	R/W-0	U-0	U-0	R/W-0, HC	R/W-0
UACTPND	_	_	USLPGRD	_	_	USUSPND	USBPWR
bit 7							bit 0

Legend:	HS = Hardware Settable bit	HC = Hardware Clearable bit		
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 15-8 **Unimplemented:** Read as '0'

bit 7 **UACTPND:** USB Activity Pending bit

1 = Module should not be suspended at the moment (requires the USLPGRD bit to be set)

0 = Module may be suspended or powered down

bit 6-5 **Unimplemented:** Read as '0'

bit 4 USLPGRD: USB Sleep/Suspend Guard bit

1 = Indicates to the USB module that it is about to be suspended or powered down

0 = No suspend

bit 3-2 **Unimplemented:** Read as '0'

bit 1 USUSPND: USB Suspend Mode Enable bit

1 = USB OTG module is in Suspend mode; USB clock is gated and the transceiver is placed in a

low-power state

0 = Normal USB OTG operation

bit 0 USBPWR: USB Operation Enable bit

1 = USB OTG module is enabled

0 = USB OTG module is disabled(1)

Note 1: Do not clear this bit unless the HOSTEN, USBEN and OTGEN bits (U1CON<3,0> and U1OTGCON<2>) are all cleared.

REGISTER 19-6: U1STAT: USB STATUS REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

R-0, HSC	U-0	U-0					
ENDPT3	ENDPT2	ENDPT1	ENDPT0	DIR	PPBI ⁽¹⁾	_	_
bit 7							bit 0

Legend:U = Unimplemented bit, read as '0'R = Readable bitW = Writable bitHSC = Hardware Settable/Clearable bit-n = Value at POR'1' = Bit is set'0' = Bit is clearedx = Bit is unknown

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

bit 7-4 **ENDPT<3:0>:** Number of the Last Endpoint Activity bits

(Represents the number of the BDT updated by the last USB transfer.)

1111 = Endpoint 15

1110 = Endpoint 14

•

0001 **= Endpoint 1**

0000 = Endpoint 0

bit 3 DIR: Last BD Direction Indicator bit

1 = The last transaction was a transmit transfer (TX)

0 = The last transaction was a receive transfer (RX)

bit 2 **PPBI:** Ping-Pong BD Pointer Indicator bit⁽¹⁾

1 = The last transaction was to the Odd BD bank

0 = The last transaction was to the Even BD bank

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

Note 1: This bit is only valid for endpoints with available Even and Odd BD registers.

REGISTER 19-7: U1CON: USB CONTROL REGISTER (DEVICE MODE)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

U-0	R-x, HSC	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
_	SE0	PKTDIS	_	HOSTEN	RESUME	PPBRST	USBEN
bit 7							bit 0

Legend: U = Unimplemented bit, read as '0'

R = Readable bit W = Writable bit HSC = Hardware Settable/Clearable bit

-n = Value at POR '1' = Bit is set '0' = Bit is cleared x = Bit is unknown

bit 15-7 **Unimplemented:** Read as '0'

bit 6 SE0: Live Single-Ended Zero Flag bit

1 = Single-ended zero is active on the USB bus

0 = No single-ended zero is detected

bit 5 **PKTDIS:** Packet Transfer Disable bit

1 = SIE token and packet processing are disabled; automatically set when a SETUP token is received

0 = SIE token and packet processing are enabled

bit 4 Unimplemented: Read as '0'

bit 3 **HOSTEN:** Host Mode Enable bit

1 = USB host capability is enabled; pull-downs on D+ and D- are activated in hardware

0 = USB host capability is disabled

bit 2 **RESUME:** Resume Signaling Enable bit

1 = Resume signaling is activated

0 = Resume signaling is disabled

bit 1 PPBRST: Ping-Pong Buffers Reset bit

1 = Resets all Ping-Pong Buffer Pointers to the Even BD banks

0 = Ping-Pong Buffer Pointers are not reset

bit 0 USBEN: USB Module Enable bit

1 = USB module and supporting circuitry are enabled (device attached); D+ pull-up is activated in hardware

0 = USB module and supporting circuitry are disabled (device detached)

REGISTER 19-8: U1CON: USB CONTROL REGISTER (HOST MODE ONLY)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15	•		•	•			bit 8

R-x, HSC	R-x, HSC	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
JSTATE	SE0	TOKBUSY	USBRST	HOSTEN	RESUME	PPBRST	SOFEN
bit 7							bit 0

Legend:U = Unimplemented bit, read as '0'R = Readable bitW = Writable bitHSC = Hardware Settable/Clearable bit-n = Value at POR'1' = Bit is set'0' = Bit is clearedx = Bit is unknown

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

bit 7 JSTATE: Live Differential Receiver J-State Flag bit

1 = J-state (differential '0' in low speed, differential '1' in full speed) is detected on the USB

0 = No J-state is detected

bit 6 **SE0:** Live Single-Ended Zero Flag bit

1 = Single-ended zero is active on the USB bus

0 = No single-ended zero is detected

bit 5 **TOKBUSY:** Token Busy Status bit

1 = Token is being executed by the USB module in the On-The-Go state

0 = No token is being executed

bit 4 USBRST: USB Module Reset bit

1 = USB Reset is generated for software Reset; application must set this bit for 50 ms, then clear it

0 = USB Reset is terminated

bit 3 **HOSTEN:** Host Mode Enable bit

1 = USB host capability is enabled; pull-downs on D+ and D- are activated in hardware

0 = USB host capability is disabled

bit 2 **RESUME:** Resume Signaling Enable bit

1 = Resume signaling is activated; software must set bit for 10 ms and then clear to enable remote

wake-up

0 = Resume signaling is disabled

bit 1 **PPBRST:** Ping-Pong Buffers Reset bit

1 = Resets all Ping-Pong Buffer Pointers to the Even BD banks

0 = Ping-Pong Buffer Pointers are not reset

bit 0 SOFEN: Start-of-Frame Enable bit

1 = Start-of-Frame token is sent every one 1 ms

0 = Start-of-Frame token is disabled

REGISTER 19-9: U1ADDR: USB ADDRESS REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_		_
bit 15							bit 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
LSPDEN ⁽¹⁾	ADDR6	ADDR5	ADDR4	ADDR3	ADDR2	ADDR1	ADDR0
bit 7		•	•	•	•		bit 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 15-8 **Unimplemented:** Read as '0'

bit 7 LSPDEN: Low-Speed Enable Indicator bit (1)

1 = USB module operates at low speed 0 = USB module operates at full speed

bit 6-0 ADDR<6:0>: USB Device Address bits

Note 1: Host mode only; in Device mode, this bit is unimplemented and read as '0'.

REGISTER 19-10: U1TOK: USB TOKEN REGISTER (HOST MODE ONLY)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

| R/W-0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| PID3 | PID2 | PID1 | PID0 | EP3 | EP2 | EP1 | EP0 |
| bit 7 | | | | | | | bit 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 15-8 **Unimplemented:** Read as '0'

bit 7-4 PID<3:0>: Token Type Identifier bits

1101 = SETUP (TX) token type transaction⁽¹⁾ 1001 = IN (RX) token type transaction⁽¹⁾ 0001 = OUT (TX) token type transaction⁽¹⁾

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

This value must specify a valid endpoint on the attached device.

Note 1: All other combinations are reserved and are not to be used.

REGISTER 19-11: U1SOF: USB OTG START-OF-FRAME COUNT REGISTER (HOST MODE ONLY)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 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	
CNT<7:0>								
bit 7								

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 15-8 **Unimplemented:** Read as '0' bit 7-0 **CNT<7:0>:** Start-of-Frame Size bits

Value represents 10 + (packet size of n bytes). For example:

0100 1010 = 64-byte packet 0010 1010 = 32-byte packet 0001 0010 = 8-byte packet

REGISTER 19-12: U1CNFG1: USB CONFIGURATION REGISTER 1

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	
_	_	_	_	_	_	_	_	
bit 15	bit 15 bit 8							

R/W-0	R/W-0	U-0	R/W-0	U-0	U-0	R/W-0	R/W-0
UTEYE	UOEMON ⁽¹⁾	_	USBSIDL	_	_	PPB1	PPB0
bit 7							bit 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 15-8 Unimplemented: Read as '0'

bit 7 UTEYE: USB Eye Pattern Test Enable bit

1 = Eye pattern test is enabled0 = Eye pattern test is disabled

bit 6 **UOEMON:** USB \overline{OE} Monitor Enable bit⁽¹⁾

1 = $\overline{\text{OE}}$ signal is active; it indicates intervals during which the D+/D- lines are driving

 $0 = \overline{OE}$ signal is inactive

bit 5 **Unimplemented:** Read as '0'

bit 4 USBSIDL: USB OTG Stop in Idle Mode bit

1 = Discontinues module operation when the device enters Idle mode

0 = Continues module operation in Idle mode

bit 3-2 **Unimplemented:** Read as '0'

bit 1-0 **PPB<1:0>:** Ping-Pong Buffers Configuration bits

11 = Even/Odd Ping-Pong Buffers are enabled for Endpoints 1 to 15

10 = Even/Odd Ping-Pong Buffers are enabled for all endpoints

01 = Even/Odd Ping-Pong Buffers are enabled for OUT Endpoint 0

00 = Even/Odd Ping-Pong Buffers are disabled

Note 1: This bit is only active when the UTRDIS bit (U1CNFG2<0>) is set.

REGISTER 19-13: U1CNFG2: USB CONFIGURATION REGISTER 2

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0
_	_	_	PUVBUS	EXTI2CEN	UVBUSDIS ⁽¹⁾	_	UTRDIS ⁽¹⁾
bit 7							bit 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 15-5 Unimplemented: Read as '0'
bit 4 PUVBUS: VBUS Pull-up Enable bit

1 = Pull-up on VBUS pin is enabled
0 = Pull-up on VBUS pin is disabled

bit 3 **EXTI2CEN:** I²C[™] Interface for External Module Control Enable bit

1 = External module(s) is controlled via the I²C interface 0 = External module(s) is controlled via the dedicated pins

bit 2 **UVBUSDIS:** USB On-Chip 5V Boost Regulator Builder Disable bit⁽¹⁾

1 = On-chip boost regulator builder is disabled; digital output control interface is enabled

0 = On-chip boost regulator builder is active

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

bit 0 UTRDIS: USB On-Chip Transceiver Disable bit(1)

1 = On-chip transceiver is disabled; digital transceiver interface is enabled

0 = On-chip transceiver is active

Note 1: Never change these bits while the USBPWR bit is set (U1PWRC<0> = 1).

19.7.2 USB INTERRUPT REGISTERS

REGISTER 19-14: U10TGIR: USB OTG INTERRUPT STATUS REGISTER (HOST MODE ONLY)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

R/K-0, HS	U-0	R/K-0, HS					
IDIF	T1MSECIF	LSTATEIF	ACTVIF	SESVDIF	SESENDIF	_	VBUSVDIF
bit 7							bit 0

Legend: U = Unimplemented bit, read as '0'

R = Readable bit K = Write '1' to Clear bit HS = Hardware Settable bit

-n = Value at POR '1' = Bit is set '0' = Bit is cleared x = Bit is unknown

bit 15-8 Unimplemented: Read as '0'

bit 7 IDIF: ID State Change Indicator bit

1 = Change in ID state is detected

0 = No ID state change is detected

bit 6 T1MSECIF: 1 Millisecond Timer bit

1 = The 1 millisecond timer has expired

 $_{
m 0}$ = The 1 millisecond timer has not expired

bit 5 LSTATEIF: Line State Stable Indicator bit

 $_{
m 1}$ = USB line state (as defined by the SE0 and JSTATE bits) has been stable for 1 ms, but different from

the last time

0 = USB line state has not been stable for 1 ms

bit 4 ACTVIF: Bus Activity Indicator bit

1 = Activity on the D+/D- lines or VBUS is detected

0 = No activity on the D+/D- lines or VBUS is detected

bit 3 SESVDIF: Session Valid Change Indicator bit

1 = VBUS has crossed VA SESS END (as defined in the "USB 2.0 Specification")(1)

0 = VBUS has not crossed VA SESS END

bit 2 SESENDIF: B-Device VBUS Change Indicator bit

1 = VBUS change on B-device is detected; VBUS has crossed VB_SESS_END (as defined in the "USB 2.0"

Specification")(1)

0 = VBUS has not crossed VB_SESS_END

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

bit 0 VBUSVDIF: A-Device VBUS Change Indicator bit

1 = VBUS change on A-device is detected; VBUS has crossed VA VBUS VLD (as defined in the

"USB 2.0 Specification")(1)

0 = No VBUS change on the A-device is detected

Note 1: VBUS threshold crossings may be either rising or falling.

Note: Individual bits can only be cleared by writing a '1' to the bit position as part of a word write operation on the entire register. Using Boolean instructions or bitwise operations to write to a single bit position will cause

all set bits, at the moment of the write, to become cleared.

REGISTER 19-15: U1OTGIE: USB OTG INTERRUPT ENABLE REGISTER (HOST MODE ONLY)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0
IDIE	T1MSECIE	LSTATEIE	ACTVIE	SESVDIE	SESENDIE	_	VBUSVDIE
bit 7							bit 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 15-8 Unimplemented: Read as '0' bit 7 IDIE: ID Interrupt Enable bit

1 = Interrupt is enabled0 = Interrupt is disabled

bit 6 T1MSECIE: 1 Millisecond Timer Interrupt Enable bit

1 = Interrupt is enabled0 = Interrupt is disabled

bit 5 LSTATEIE: Line State Stable Interrupt Enable bit

1 = Interrupt is enabled0 = Interrupt is disabled

bit 4 ACTVIE: Bus Activity Interrupt Enable bit

1 = Interrupt is enabled0 = Interrupt is disabled

bit 3 SESVDIE: Session Valid Interrupt Enable bit

1 = Interrupt is enabled0 = Interrupt is disabled

bit 2 SESENDIE: B-Device Session End Interrupt Enable bit

1 = Interrupt is enabled0 = Interrupt is disabled

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

bit 0 VBUSVDIE: A-Device VBUS Valid Interrupt Enable bit

1 = Interrupt is enabled0 = Interrupt is disabled

REGISTER 19-16: U1IR: USB INTERRUPT STATUS REGISTER (DEVICE MODE ONLY)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_		_		-	_
bit 15							bit 8

R/K-0, HS	U-0	R/K-0, HS	R/K-0, HS	R/K-0, HS	R/K-0, HS	R-0	R/K-0, HS
STALLIF	_	RESUMEIF	IDLEIF	TRNIF	SOFIF	UERRIF	URSTIF
bit 7							bit 0

Legend: U = Unimplemented bit, read as '0'

R = Readable bit K = Write '1' to Clear bit HS = Hardware Settable bit

-n = Value at POR '1' = Bit is set '0' = Bit is cleared x = Bit is unknown

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

bit 7 STALLIF: STALL Handshake Interrupt bit

1 = A STALL handshake was sent by the peripheral during the handshake phase of the transaction in Device mode

0 = A STALL handshake has not been sent

bit 6 Unimplemented: Read as '0'

bit 5 **RESUMEIF:** Resume Interrupt bit

1 = A K-state is observed on the D+ or D- pin for 2.5 μs (differential '1' for low speed, differential '0' for full speed)

0 = No K-state is observed

bit 4 **IDLEIF:** Idle Detect Interrupt bit

1 = Idle condition is detected (constant Idle state of 3 ms or more)

0 = No Idle condition is detected

bit 3 TRNIF: Token Processing Complete Interrupt bit

1 = Processing of the current token is complete; read the U1STAT register for endpoint information

0 = Processing of the current token is not complete; clear the U1STAT register or load the next token from U1STAT (clearing this bit causes the U1STAT FIFO to advance)

bit 2 **SOFIF:** Start-of-Frame Token Interrupt bit

1 = A Start-of-Frame token is received by the peripheral or the Start-of-Frame threshold is reached by the host

0 = No Start-of-Frame token is received or threshold reached

bit 1 **UERRIF**: USB Error Condition Interrupt bit (read-only)

1 = An unmasked error condition has occurred; only error states enabled in the U1EIE register can set this bit

0 = No unmasked error condition has occurred

bit 0 URSTIF: USB Reset Interrupt bit

 $_1$ = Valid USB Reset has occurred for at least 2.5 μ s; Reset state must be cleared before this bit can be reasserted

0 = No USB Reset has occurred. Individual bits can only be cleared by writing a '1' to the bit position as part of a word write operation on the entire register. Using Boolean instructions or bitwise operations to write to a single bit position will cause all set bits, at the moment of the write, to become cleared.

Note: Individual bits can only be cleared by writing a '1' to the bit position as part of a word write operation on the entire register. Using Boolean instructions or bitwise operations to write to a single bit position will cause all set bits, at the moment of the write, to become cleared.

REGISTER 19-17: U1IR: USB INTERRUPT STATUS REGISTER (HOST MODE ONLY)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

| R/K-0, HS |
|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| STALLIF | ATTACHIF | RESUMEIF | IDLEIF | TRNIF | SOFIF | UERRIF | DETACHIF |
| bit 7 | | | | | | | bit 0 |

Legend: U = Unimplemented bit, read as '0'

R = Readable bit K = Write '1' to Clear bit HS = Hardware Settable bit

-n = Value at POR '1' = Bit is set '0' = Bit is cleared x = Bit is unknown

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

bit 7 STALLIF: STALL Handshake Interrupt bit

- 1 = A STALL handshake was sent by the peripheral device during the handshake phase of the transaction in Device mode
- 0 = A STALL handshake has not been sent
- bit 6 ATTACHIF: Peripheral Attach Interrupt bit
 - 1 = A peripheral attachment has been detected by the module; it is set if the bus state is not SE0 and there has been no bus activity for $2.5~\mu s$
 - 0 = No peripheral attachment has been detected
- bit 5 **RESUMEIF:** Resume Interrupt bit
 - 1 = A K-state is observed on the D+ or D- pin for 2.5 μs (differential '1' for low speed, differential '0' for full speed)
 - 0 = No K-state is observed
- bit 4 **IDLEIF:** Idle Detect Interrupt bit
 - 1 = Idle condition is detected (constant Idle state of 3 ms or more)
 - 0 = No Idle condition is detected
- bit 3 **TRNIF:** Token Processing Complete Interrupt bit
 - 1 = Processing of the current token is complete; read the U1STAT register for endpoint information
 - 0 = Processing of the current token is not complete; clear the U1STAT register or load the next token from U1STAT
- bit 2 **SOFIF:** Start-of-Frame Token Interrupt bit
 - 1 = A Start-of-Frame token is received by the peripheral or the Start-of-Frame threshold is reached by the host
 - 0 = No Start-of-Frame token is received or threshold is reached
- bit 1 **UERRIF:** USB Error Condition Interrupt bit
 - 1 = An unmasked error condition has occurred; only error states enabled in the U1EIE register can set this bit
 - 0 = No unmasked error condition has occurred
- bit 0 **DETACHIF:** Detach Interrupt bit
 - 1 = A peripheral detachment has been detected by the module; Reset state must be cleared before this bit can be reasserted
 - 0 = No peripheral detachment is detected. Individual bits can only be cleared by writing a '1' to the bit position as part of a word write operation on the entire register. Using Boolean instructions or bitwise operations to write to a single bit position will cause all set bits, at the moment of the write, to become cleared.

Note: Individual bits can only be cleared by writing a '1' to the bit position as part of a word write operation on the entire register. Using Boolean instructions or bitwise operations to write to a single bit position will cause all set bits, at the moment of the write, to become cleared.

REGISTER 19-18: U1IE: USB INTERRUPT ENABLE REGISTER (ALL USB MODES)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

R/K-0, HS	R/K-0, HS	R/K-0, HS	R/K-0, HS	R/K-0, HS	R/K-0, HS	R/K-0, HS	R/K-0, HS
STALLIE	ATTACHIE ⁽¹⁾	RESUMEIE	IDLEIE	TRNIE	SOFIE	UERRIE	URSTIE
							DETACHIE ⁽¹⁾
bit 7							bit 0

Legend:	U = Unimplemented bit, rea	d as '0'	
R = Readable bit	K = Write '1' to Clear bit	HS = Hardware Settable bit	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

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

bit 7 STALLIE: STALL Handshake Interrupt Enable bit

1 = Interrupt is enabled0 = Interrupt is disabled

bit 6 ATTACHIE: Peripheral Attach Interrupt bit (Host mode only)⁽¹⁾

1 = Interrupt is enabled0 = Interrupt is disabled

bit 5 **RESUMEIE:** Resume Interrupt bit

1 = Interrupt is enabled0 = Interrupt is disabled

bit 4 IDLEIE: Idle Detect Interrupt bit

1 = Interrupt is enabled0 = Interrupt is disabled

bit 3 TRNIE: Token Processing Complete Interrupt bit

1 = Interrupt is enabled0 = Interrupt is disabled

bit 2 **SOFIE:** Start-of-Frame Token Interrupt bit

1 = Interrupt is enabled0 = Interrupt is disabled

bit 1 **UERRIE:** USB Error Condition Interrupt bit

1 = Interrupt is enabled0 = Interrupt is disabled

bit 0 URSTIE or DETACHIE: USB Reset Interrupt (Device mode) or

USB Detach Interrupt (Host mode) Enable bit(1)

1 = Interrupt is enabled0 = Interrupt is disabled

Note 1: The ATTACHIE and DETACHIE bits are unimplemented in Device mode, read as '0'.

REGISTER 19-19: U1EIR: USB ERROR INTERRUPT STATUS REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

R/K-0, HS	U-0	R/K-0, HS					
BTSEF	_	DMAEF	BTOEF	DFN8EF	CRC16EF	CRC5EF	PIDEF
						EOFEF	
bit 7							bit 0

Legend:U = Unimplemented bit, read as '0'R = Readable bitK = Write '1' to Clear bitHS = Hardware Settable bit-n = Value at POR'1' = Bit is set'0' = Bit is clearedx = Bit is unknown

bit 15-8 **Unimplemented:** Read as '0' bit 7 **BTSEF:** Bit Stuff Error Flag bit

1 = Bit stuff error has been detected 0 = No bit stuff error has been detected

bit 6 **Unimplemented:** Read as '0' bit 5 **DMAEF:** DMA Error Flag bit

1 = A USB DMA error condition is detected; the data size indicated by the BD byte count field is less than the number of received bytes, the received data is truncated

0 = No DMA error

bit 4 BTOEF: Bus Turnaround Time-out Error Flag bit

1 = Bus turnaround time-out has occurred

0 = No bus turnaround time-out

bit 3 **DFN8EF:** Data Field Size Error Flag bit

1 = Data field was not an integral number of bytes0 = Data field was an integral number of bytes

bit 2 CRC16EF: CRC16 Failure Flag bit

1 = CRC16 failed 0 = CRC16 passed

bit 1 For Device Mode:

CRC5EF: CRC5 Host Error Flag bit

1 = Token packet is rejected due to CRC5 error 0 = Token packet is accepted (no CRC5 error)

For Host Mode:

EOFEF: End-of-Frame Error Flag bit 1 = End-of-Frame error has occurred 0 = End-of-Frame interrupt is disabled

bit 0 PIDEF: PID Check Failure Flag bit

1 = PID check failed0 = PID check passed

Note: Individual bits can only be cleared by writing a '1' to the bit position as part of a word write operation on the entire register. Using Boolean instructions or bitwise operations to write to a single bit position will cause all set bits, at the moment of the write, to become cleared.

REGISTER 19-20: U1EIE: USB ERROR INTERRUPT ENABLE REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BTSEE	_	DMAEE	BTOEE	DFN8EE	CRC16EE	CRC5EE	PIDEE
						EOFEE	
bit 7							bit 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 15-8 Unimplemented: Read as '0'

bit 7 BTSEE: Bit Stuff Error Interrupt Enable bit

1 = Interrupt is enabled0 = Interrupt is disabled

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

bit 5 DMAEE: DMA Error Interrupt Enable bit

1 = Interrupt is enabled0 = Interrupt is disabled

bit 4 BTOEE: Bus Turnaround Time-out Error Interrupt Enable bit

1 = Interrupt is enabled0 = Interrupt is disabled

bit 3 **DFN8EE**: Data Field Size Error Interrupt Enable bit

1 = Interrupt is enabled0 = Interrupt is disabled

bit 2 CRC16EE: CRC16 Failure Interrupt Enable bit

1 = Interrupt is enabled0 = Interrupt is disabled

bit 1 For Device Mode:

CRC5EE: CRC5 Host Error Interrupt Enable bit

1 = Interrupt is enabled0 = Interrupt is disabled

For Host Mode:

EOFEE: End-of-Frame Error interrupt Enable bit

1 = Interrupt is enabled0 = Interrupt is disabled

bit 0 PIDEE: PID Check Failure Interrupt Enable bit

1 = Interrupt is enabled0 = Interrupt is disabled

19.7.3 USB ENDPOINT MANAGEMENT REGISTERS

REGISTER 19-21: U1EPn: USB ENDPOINT n CONTROL REGISTERS (n = 0 TO 15)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	1	1	1	_	_	_
bit 15							bit 8

R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
LSPD ⁽¹⁾	RETRYDIS ⁽¹⁾	_	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK
bit 7							bit 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 15-8 **Unimplemented:** Read as '0'

bit 7 LSPD: Low-Speed Direct Connection Enable bit (U1EP0 only)(1)

1 = Direct connection to a low-speed device is enabled

0 = Direct connection to a low-speed device is disabled

bit 6 **RETRYDIS:** Retry Disable bit (U1EP0 only)⁽¹⁾

1 = Retry NAK transactions are disabled

0 = Retry NAK transactions are enabled; retry is done in hardware

bit 5 **Unimplemented:** Read as '0'

bit 4 **EPCONDIS:** Bidirectional Endpoint Control bit

If EPTXEN and EPRXEN = 1:

1 = Disables Endpoint n from control transfers; only TX and RX transfers are allowed

0 = Enables Endpoint n for control (SETUP) transfers; TX and RX transfers are also allowed

For All Other Combinations of EPTXEN and EPRXEN:

This bit is ignored.

bit 3 EPRXEN: Endpoint Receive Enable bit

1 = Endpoint n receive is enabled

0 = Endpoint n receive is disabled

bit 2 **EPTXEN:** Endpoint Transmit Enable bit

1 = Endpoint n transmit is enabled

0 = Endpoint n transmit is disabled

bit 1 **EPSTALL:** Endpoint STALL Status bit

1 = Endpoint n was STALLed

0 = Endpoint n was not STALLed

bit 0 **EPHSHK:** Endpoint Handshake Enable bit

1 = Endpoint handshake is enabled

0 = Endpoint handshake is disabled (typically used for isochronous endpoints)

Note 1: These bits are available only for U1EP0 and only in Host mode. For all other U1EPn registers, these bits are always unimplemented and read as '0'.

NOTES:	
--------	--

20.0 DATA SIGNAL MODULATOR (DSM)

Note: This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the "dsPIC33/PIC24 Family Reference Manual", "Data Signal Modulator (DSM)" (DS39744). The information in this data

sheet supersedes the information in the

The Data Signal Modulator (DSM) allows the user to mix a digital data stream (the "modulator signal") with a carrier signal to produce a modulated output. Both the carrier and the modulator signals are supplied to the DSM module, either internally from the output of a peripheral, or externally through an input pin.

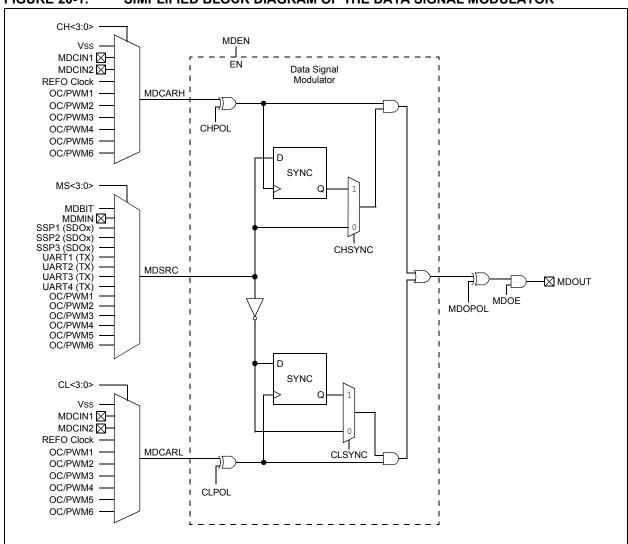
FRM.

The modulated output signal is generated by performing a logical AND operation of both the carrier and modulator signals, and then it is provided to the MDOUT pin. Using this method, the DSM can generate the following types of key modulation schemes:

- · Frequency Shift Keying (FSK)
- · Phase-Shift Keying (PSK)
- On-Off Keying (OOK)

Figure 20-1 shows a simplified block diagram of the Data Signal Modulator peripheral.

FIGURE 20-1: SIMPLIFIED BLOCK DIAGRAM OF THE DATA SIGNAL MODULATOR



REGISTER 20-1: MDCON: DATA SIGNAL MODULATOR CONTROL REGISTER

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
MDEN	_	MDSIDL	_	_	_	_	
bit 15							bit 8

U-0	R/W-0	R/W-1	R/W-0	U-0	U-0	U-0	R/W-0
_	MDOE	MDSLR	MDOPOL	_	_	_	MDBIT ⁽¹⁾
bit 7							bit 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 15 MDEN: DSM Module Enable bit

1 = Modulator module is enabled and mixing input signals

0 = Modulator module is disabled and has no output

bit 14 Unimplemented: Read as '0'

bit 13 MDSIDL: DSM Stop in Idle Mode bit

1 = Discontinues module operation when device enters Idle mode

0 = Continues module operation in Idle mode

bit 12-7 Unimplemented: Read as '0'

bit 6 MDOE: DSM Module Pin Output Enable bit

1 = Modulator pin output is enabled

0 = Modulator pin output is disabled

bit 5 MDSLR: MDOUT Pin Slew Rate Limiting bit

1 = MDOUT pin slew rate limiting is enabled0 = MDOUT pin slew rate limiting is disabled

bit 4 MDOPOL: DSM Output Polarity Select bit

1 = Modulator output signal is inverted

0 = Modulator output signal is not inverted

bit 3-1 **Unimplemented:** Read as '0'

bit 0 **MDBIT:** Manual Modulation Input bit⁽¹⁾

1 = Carrier is modulated

0 = Carrier is not modulated

Note 1: The MDBIT must be selected as the modulation source (MDSRC<3:0> = 0000).

REGISTER 20-2: MDSRC: DATA SIGNAL MODULATOR SOURCE CONTROL REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

R/W-x	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x
SODIS ⁽¹⁾	_	_	_	MS3 ⁽²⁾	MS2 ⁽²⁾	MS1 ⁽²⁾	MS0 ⁽²⁾
bit 7							bit 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 15-8 **Unimplemented:** Read as '0'

bit 7 **SODIS:** DSM Source Output Disable bit⁽¹⁾

1 = Output signal driving the peripheral output pin (selected by MS<3:0>) is disabled 0 = Output signal driving the peripheral output pin (selected by MS<3:0>) is enabled

bit 6-4 Unimplemented: Read as '0'

bit 3-0 MS<3:0> DSM Source Selection bits⁽²⁾

1111 = Unimplemented

1110 = SPI3 module output (SDO3)

1101 = Output Compare/PWM Module 6 output

1100 = Output Compare/PWM Module 5 output

1011 = Output Compare/PWM Module 4 output

1010 = Output Compare/PWM Module 3 output

1001 = Output Compare/PWM Module 2 output

1000 = Output Compare/PWM Module 1 output

0111 = UART4 TX output

0110 = UART3 TX output

0101 = UART2 TX output

0100 = UART1 TX output

0011 = SPI2 module output (SDO2)

0010 = SPI1 module output (SDO1)

0001 = Input on MDMIN pin

0000 = Manual modulation using MDBIT (MDCON<0>)

Note 1: This bit is only affected by a POR.

2: These bits are not affected by a POR.

REGISTER 20-3: MDCAR: DATA SIGNAL MODULATOR CARRIER CONTROL REGISTER

R/W-x	R/W-x	R/W-x	U-0	R/W-x	R/W-x	R/W-x	R/W-x
CHODIS	CHPOL	CHSYNC	_	CH3 ⁽¹⁾	CH2 ⁽¹⁾	CH1 ⁽¹⁾	CH0 ⁽¹⁾
bit 15							bit 8

R/W-0	R/W-x	R/W-x	U-0	R/W-x	R/W-x	R/W-x	R/W-x
CLODIS	CLPOL	CLSYNC	_	CL3 ⁽¹⁾	CL2 ⁽¹⁾	CL1 ⁽¹⁾	CL0 ⁽¹⁾
bit 7							bit 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 15 CHODIS: DSM High Carrier Output Disable bit

1 = Output signal driving the peripheral output pin (selected by CH<3:0>) is disabled

0 = Output signal driving the peripheral output pin is enabled

bit 14 CHPOL: DSM High Carrier Polarity Select bit

1 = Selected high carrier signal is inverted

0 = Selected high carrier signal is not inverted

bit 13 CHSYNC: DSM High Carrier Synchronization Enable bit

1 = Modulator waits for a falling edge on the high carrier before allowing a switch to the low carrier

0 = Modulator output is not synchronized to the high time carrier signal (1)

bit 12 **Unimplemented:** Read as '0'

bit 11-8 CH<3:0> DSM Data High Carrier Selection bits⁽¹⁾

1111

... = Reserved

1010

1001 = Output Compare/PWM Module 6 output

1000 = Output Compare/PWM Module 5 output

0111 = Output Compare/PWM Module 4 output

0110 = Output Compare/PWM Module 3 output

0101 = Output Compare/PWM Module 2 output

0100 = Output Compare/PWM Module 1 output

0011 = Reference Clock Output (REFO)

0010 = Input on MDCIN2 pin

0001 = Input on MDCIN1 pin

0000 = Vss

bit 7 CLODIS: DSM Low Carrier Output Disable bit

1 = Output signal driving the peripheral output pin (selected by CL<3:0>) is disabled

0 = Output signal driving the peripheral output pin is enabled

bit 6 **CLPOL:** DSM Low Carrier Polarity Select bit

1 = Selected low carrier signal is inverted

0 = Selected low carrier signal is not inverted

bit 5 CLSYNC: DSM Low Carrier Synchronization Enable bit

1 = Modulator waits for a falling edge on the low carrier before allowing a switch to the high carrier

= Modulator output is not synchronized to the low time carrier signal⁽¹⁾

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

bit 3-0 CL<3:0>: DSM Data Low Carrier Selection bits⁽¹⁾

Bit settings are identical to those for CH<3:0>.

Note 1: Narrowed carrier pulse widths or spurs may occur in the signal stream if the carrier is not synchronized.

21.0 ENHANCED PARALLEL MASTER PORT (EPMP)

Note:

This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the "dsPIC33/PIC24 Family Reference Manual", "Enhanced Parallel Master Port (EPMP)" (DS39730). The information in this data sheet supersedes the information in the FRM.

The Enhanced Parallel Master Port (EPMP) module provides a parallel, 4-bit (Master mode only) and 8-bit (Master and Slave modes) data bus interface to communicate with off-chip modules, such as memories, FIFOs, LCD controllers and other microcontrollers. This module can serve as either the master or the slave on the communication bus.

For EPMP Master modes, all external addresses are mapped into the internal Extended Data Space (EDS). This is done by allocating a region of the EDS for each chip select and then assigning each chip select to a particular external resource, such as a memory or external controller. This region should not be assigned to another device resource, such as RAM or SFRs. To perform a write or read on an external resource, the CPU simply performs a write or read within the address range assigned for the EPMP.

Key features of the EPMP module are:

- Extended Data Space (EDS) Interface allows Direct Access from the CPU
- Up to 10 Programmable Address Lines
- · Up to 2 Chip Select Lines
- Up to 1 Acknowledgment Line (one per chip select)
- · 4-Bit and 8-Bit Wide Data Bus
- · Programmable Strobe Options (per chip select)
 - Individual Read and Write Strobes or;
 - Read/Write Strobe with Enable Strobe
- Programmable Address/Data Multiplexing
- · Programmable Address Wait States
- Programmable Data Wait States (per chip select)
- Programmable Polarity on Control Signals (per chip select)
- · Legacy Parallel Slave Port (PSP) Support
- · Enhanced Parallel Slave Port Support
 - Address Support
 - 4-Byte Deep Auto-Incrementing Buffer

21.1 Memory Addressable in Different Modes

The memory space addressable by the device depends on the address/data multiplexing selection; it varies from 1K to 2 MB. Refer to Table 21-1 for different Memory-Addressable modes.

TABLE 21-1: MEMORY ADDRESSABLE IN DIFFERENT MODES

Data Port Size	PMA<9:8>	PMA<7:0>	PMD<7:4>	PMD<3:0>	Accessible Memory	
	Demultiplexed	Address (Al	ORMUX<1:0>	= 00)		
8-Bit (PTSZ<1:0> = 00)	Addr<9:8>	Addr<7:0>	Da	ata	1K	
4-Bit (PTSZ<1:0> = 01)	Addr<9:8>	Addr<7:0>	_	Data	1K	
	1 Address	Phase (ADRI	/IUX<1:0> = 0	1)		
8-Bit (PTSZ<1:0> = 00)	_	PMALL	Addr<7:	0> Data	1K	
4-Bit (PTSZ<1:0> = 01)	Addr<9:8>	PMALL	Addr<7:4>	Addr<3:0>	1K	
4-Dil (P132<1.0> - 01)	Auui \ 9.62	PIVIALL	_	Data (1)		
	2 Address I	Phases (ADR	MUX<1:0> = 1	L O)		
		PMALL	Addr	<7:0>		
8-Bit (PTSZ<1:0> = 00)	_	PMALH	Addr<15:8>		64K	
		_	Da	ata		
		PMALL	Addr<3:0>			
4-Bit (PTSZ<1:0> = 01)	Addr<9:8>	PMALH	Addr<7:4>		1K	
		_	Data			
	3 Address I	Phases (ADR	MUX<1:0> = 1	1)		
		PMALL	Addr	<7:0>		
8-Bit (PTSZ<1:0> = 00)		PMALH	Addr<	:15:8>	2 MB	
0-bit (F132<1.0> = 00)	_	PMALU	Addr<	22:16>	Z IVID	
		_	Da	ata		
		PMALL	Addr<3:0>			
4-Bit (PTSZ<1:0> = 01)	Addr<13:12>	PMALH	Addr	<7:4>	16K	
4-Dil (F132\1.0/ - 01)	Auui > 13.12>	PMALU	Addr<	:11:8>		
		_	Da	ata		

TABLE 21-2: ENHANCED PARALLEL MASTER PORT PIN DESCRIPTIONS

Pin Name (Alternate Function)	Туре	Description			
D	0	Address Bus bit 14			
PMA<14> (PMCS1)	I/O	Data Bus bit 14 (16-bit port with Multiplexed Addressing)			
(1 10001)	0	Chip Select 1 (alternate location)			
PMA<9:3>	0	Address Bus bits<9:3>			
PMA<2>	0	Address Bus bit 2			
(PMALU)	0	Address Latch Upper Strobe for Multiplexed Address			
PMA<1>	I/O	Address Bus bit 1			
(PMALH)	0	Address Latch High Strobe for Multiplexed Address			
PMA<0>	I/O	Address Bus bit 0			
(PMALL)	0	Address Latch Low Strobe for Multiplexed Address			
PMD<7:0>	I/O	Data Bus bits<7:0>, Data bits<15:8>			
PIVID~1.U>	0	Address Bus bits<7:0>			
PMCS1	I/O	Chip Select 1			
PMCS2	I/O	Chip Select 2			
PMWR	I/O	Write Strobe			
PMRD	I/O	Read Strobe			
PMBE1	0	Byte Indicator			
PMBE0	0	Nibble or Byte Indicator			
PMACK1	I	Acknowledgment Signal 1			

REGISTER 21-1: PMCON1: EPMP CONTROL REGISTER 1

R/W-0	U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
PMPEN	_	PSIDL	ADRMUX1	ADRMUX0	_	MODE1	MODE0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
CSF1	CSF0	ALP	ALMODE	_	BUSKEEP	IRQM1	IRQM0
bit 7							bit 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 15 **PMPEN:** Enhanced Parallel Master Port Enable bit

1 = EPMP is enabled 0 = EPMP is disabled

bit 14 Unimplemented: Read as '0'

bit 13 PSIDL: PMP Stop in Idle Mode bit

1 = Discontinues module operation when device enters Idle mode

0 = Continues module operation in Idle mode

bit 12-11 ADRMUX<1:0>: Address/Data Multiplexing Selection bits

11 = Lower address bits are multiplexed with data bits using 3 address phases

10 = Lower address bits are multiplexed with data bits using 2 address phases

01 = Lower address bits are multiplexed with data bits using 1 address phase

00 = Address and data appear on separate pins

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

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

11 = Master mode

10 = Enhanced PSP: Pins used are PMRD, PMWR, PMCS, PMD<7:0> and PMA<1:0>

01 = Buffered PSP: Pins used are PMRD, PMWR, PMCS and PMD<7:0>

00 = Legacy Parallel Slave Port: Pins used are PMRD, PMWR, PMCS and PMD<7:0>

bit 7-6 **CSF<1:0>:** Chip Select Function bits

11 = Reserved

10 = PMA<14> is used for Chip Select 1

01 = Reserved

00 = PMCS1 is used for Chip Select 1

bit 5 **ALP:** Address Latch Polarity bit

1 = Active-high (PMALL, PMALH and PMALU)

0 = Active-low (PMALL, PMALH and PMALU)

bit 4 ALMODE: Address Latch Strobe Mode bit

1 = Enables "smart" address strobes (each address phase is only present if the current access would cause a different address in the latch than the previous address)

0 = Disables "smart" address strobes

bit 3 Unimplemented: Read as '0'

bit 2 **BUSKEEP:** Bus Keeper bit

1 = Data bus keeps its last value when not actively being driven

0 = Data bus is in a high-impedance state when not actively being driven

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

11 = Interrupt is 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 PSP mode only)

10 = Reserved

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

00 = No interrupt is generated

REGISTER 21-2: PMCON2: EPMP CONTROL REGISTER 2

R-0, HSC	U-0	R/C-0, HS	R/C-0, HS	U-0	U-0	U-0	U-0
PMPBUSY	_	ERROR	TIMEOUT	_	_	_	_
bit 15							bit 8

| R/W-0 |
|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|
| RADDR23 ⁽¹⁾ | RADDR22 ⁽¹⁾ | RADDR21 ⁽¹⁾ | RADDR20 ⁽¹⁾ | RADDR19 ⁽¹⁾ | RADDR18 ⁽¹⁾ | RADDR17 ⁽¹⁾ | RADDR16 ⁽¹⁾ |
| bit 7 | | | | | | | bit 0 |

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

bit 15 **PMPBUSY:** PMP Busy bit (Master mode only)

1 = Port is busy0 = Port is not busy

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

bit 13 **ERROR:** PMP Error bit

1 = Transaction error (illegal transaction was requested)

0 = Transaction completed successfully

bit 12 **TIMEOUT:** PMP Time-out bit

1 = Transaction timed out

0 = Transaction completed successfully

bit 11-8 Unimplemented: Read as '0'

bit 7-0 RADDR<23:16>: Parallel Master Port Reserved Address Space bits⁽¹⁾

Note 1: If RADDR<23:16> = 000000000, then the last EDS address for Chip Select 2 will be FFFFFFh.

REGISTER 21-3: PMCON3: EPMP CONTROL REGISTER 3

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
PTWREN	PTRDEN	PTBE1EN	PTBE0EN	_	AWAITM1	AWAITM0	AWAITE
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 7							bit 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 15 PTWREN: PMP Write/Enable Strobe Port Enable bit

1 = PMWR port is enabled0 = PMWR port is disabled

bit 14 PTRDEN: PMP Read/Write Strobe Port Enable bit

1 = PMRD/PMWR port is enabled 0 = PMRD/PMWR port is disabled

bit 13 PTBE1EN: PMP High Nibble/Byte Enable Port Enable bit

1 = PMBE1 port is enabled 0 = PMBE1 port is disabled

bit 12 PTBE0EN: PMP Low Nibble/Byte Enable Port Enable bit

1 = PMBE0 port is enabled 0 = PMBE0 port is disabled

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

bit 10-9 **AWAITM<1:0>:** Address Latch Strobe Wait State bits

11 = Wait of 3½ Tcy 10 = Wait of 2½ Tcy 01 = Wait of 1½ Tcy 00 = Wait of ½ Tcy

bit 8 AWAITE: Address Hold After Address Latch Strobe Wait State bit

1 = Wait of 11/4 Tcy 0 = Wait of 1/4 Tcy

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

REGISTER 21-4: PMCON4: EPMP CONTROL REGISTER 4

U-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
_	PTEN14	_	_	_	_	PTEN<9:8>	
bit 15							bit 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
		PTEN<7:3>				PTEN<2:0>	
bit 7							bit 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 15 **Unimplemented:** Read as '0' bit 14 **PTEN14:** PMA14 Port Enable bit

1 = PMA14 functions as either Address Line 14 or Chip Select 1

0 = PMA14 functions as port I/O

bit 13-10 **Unimplemented:** Read as '0'

bit 9-3 PTEN<9:3>: EPMP Address Port Enable bits

1 = PMA<9:3> function as EPMP address lines

0 = PMA<9:3> function as port I/Os

bit 2-0 PTEN<2:0>: PMALU/PMALH/PMALL Strobe Enable bits

1 = PMA<2:0> function as either address lines or address latch strobes

0 = PMA<2:0> function as port I/Os

REGISTER 21-5: PMCSxCF: EPMP CHIP SELECT x CONFIGURATION REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
CSDIS	CSP	CSPTEN	BEP	_	WRSP	RDSP	SM
bit 15							bit 8

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0
ACKP	PTSZ1	PTSZ0	_	_	_	_	_
bit 7							bit 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 15 CSDIS: Chip Select x Disable bit

1 = Disables the Chip Select x functionality

0 = Enables the Chip Select x functionality

bit 14 CSP: Chip Select x Polarity bit

1 = Active-high (PMCSx)

 $0 = Active-low (\overline{PMCSx})$

bit 13 CSPTEN: PMCSx Port Enable bit

1 = PMCSx port is enabled

0 = PMCSx port is disabled

bit 12 **BEP:** Chip Select x Nibble/Byte Enable Polarity bit

1 = Nibble/byte enable is active-high (PMBE0, PMBE1)

0 = Nibble/byte enable is active-low (PMBE0, PMBE1)

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

bit 10 **WRSP:** Chip Select x Write Strobe Polarity bit

For Slave Modes and Master Mode When SM = 0:

1 = Write strobe is active-high (PMWR)

0 = Write strobe is active-low (PMWR)

For Master Mode When SM = 1:

1 = Enable strobe is active-high

0 = Enable strobe is active-low

bit 9 RDSP: Chip Select x Read Strobe Polarity bit

For Slave Modes and Master Mode When SM = 0:

1 = Read strobe is active-high (PMRD)

0 = Read strobe is active-low (PMRD)

For Master Mode When SM = 1:

1 = Read/write strobe is active-high (PMRD/ \overline{PMWR})

0 = Read/write strobe is active-low (PMRD/PMWR)

bit 8 **SM:** Chip Select x Strobe Mode bit

1 = Read/write and enable strobes (PMRD/PMWR)

0 = Read and write strobes (PMRD and PMWR)

bit 7 ACKP: Chip Select x Acknowledge Polarity bit

1 = ACK is active-high (PMACK1)

 $0 = ACK \text{ is active-low } (\overline{PMACK1})$

bit 6-5 PTSZ<1:0>: Chip Select x Port Size bits

11 = Reserved

10 = Reserved

01 = 4-bit port size (PMD<3:0>)

00 = 8-bit port size (PMD<7:0>)

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

REGISTER 21-6: PMCSxBS: EPMP CHIP SELECT x BASE ADDRESS REGISTER (2)

R/W ⁽¹⁾								
BASE<23:16>								
bit 15							bit 8	

R/W ⁽¹⁾	U-0	U-0	U-0	R/W ⁽¹⁾	U-0	U-0	U-0
BASE15	_	_	_	BASE11	_	_	_
bit 7							bit 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 15-7 BASE<23:15>: Chip Select x Base Address bits⁽¹⁾

bit 6-4 Unimplemented: Read as '0'

bit 3 BASE11: Chip Select x Base Address bit⁽¹⁾

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

Note 1: The value at POR is 0080h for PMCS1BS and 0880h for PMCS2BS.

2: If the whole PMCS2BS register is written together as 0x0000, then the last EDS address for Chip Select 1 will be FFFFFh. In this case, Chip Select 2 should not be used. PMCS1BS has no such feature.

REGISTER 21-7: PMCSxMD: EPMP CHIP SELECT x MODE REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
ACKM1	ACKM0	AMWAIT2	AMWAIT1	AMWAIT0	_	_	_
bit 15							bit 8

| R/W-0 |
|---------|---------|---------|---------|---------|---------|---------|---------|
| DWAITB1 | DWAITB0 | DWAITM3 | DWAITM2 | DWAITM1 | DWAITM0 | DWAITE1 | DWAITE0 |
| bit 7 | | | | | | | bit 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 15-14 ACKM<1:0>: Chip Select x Acknowledge Mode bits

11 = Reserved

10 = PMACKx is used to determine when a read/write operation is complete

01 = PMACKx is used to determine when a read/write operation is complete with time-out (If DWAITM<3:0> = 0000, the maximum time-out is 255 Tcy or else it is DWAITM<3:0> cycles.)

00 = PMACKx is not used

bit 13-11 AMWAIT<2:0>: Chip Select x Alternate Master Wait State bits

111 = Wait of 10 alternate master cycles

Ī

001 = Wait of 4 alternate master cycles

000 = Wait of 3 alternate master cycles

bit 10-8 **Unimplemented:** Read as '0'

bit 7-6 **DWAITB<1:0>:** Chip Select x Data Setup Before Read/Write Strobe Wait State bits

11 = Wait of 31/4 TcY

10 = Wait of 21/4 TCY

01 = Wait of 11/4 TcY

00 = Wait of 1/4 TcY

bit 5-2 **DWAITM<3:0>:** Chip Select x Data Read/Write Strobe Wait State bits

For Write Operations:

1111 = Wait of 151/2 TcY

•

•

0001 = Wait of 11/2 Tcy

0000 = Wait of 1/2 TCY

For Read Operations:

1111 = Wait of 153/4 TcY

.

•

•

0001 = Wait of 13/4 TcY

0000 = Wait of 3/4 TcY

REGISTER 21-7: PMCSxMD: EPMP CHIP SELECT x MODE REGISTER (CONTINUED)

bit 1-0 **DWAITE<1:0>:** Chip Select x Data Hold After Read/Write Strobe Wait State bits

For Write Operations:

11 = Wait of 31/4 TCY

10 = Wait of 21/4 TCY

01 = Wait of 11/4 TcY

00 = Wait of 1/4 Tcy

For Read Operations:

11 = Wait of 3 Tcy

10 = Wait of 2 Tcy

01 = Wait of 1 Tcy

00 = Wait of 0 Tcy

REGISTER 21-8: PMSTAT: EPMP STATUS REGISTER (SLAVE MODE ONLY)

R-0, HSC	R/W-0, HS	U-0	U-0	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC
IBF	IBOV	_	_	IB3F ⁽¹⁾	IB2F ⁽¹⁾	IB1F ⁽¹⁾	IB0F ⁽¹⁾
bit 15							bit 8

R-1, HSC	R/W-0, HS	U-0	U-0	R-1, HSC	R-1, HSC	R-1, HSC	R-1, HSC
OBE	OBUF	_	_	OB3E	OB2E	OB1E	OB0E
bit 7							bit 0

Legend:	HS = Hardware Settable bit HSC = Hardware Settable/Clearable bit			
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 15 IBF: Input Buffer Full Status bit

1 = All writable Input Buffer registers are full

0 = Some or all of the writable Input Buffer registers are empty

bit 14 IBOV: Input Buffer Overflow Status bit

1 = A write attempt to a full Input Buffer register occurred (must be cleared in software)

0 = No overflow occurred

bit 13-12 **Unimplemented:** Read as '0'

bit 11-8 **IB3F:IB0F:** Input Buffer x Status Full bits⁽¹⁾

1 = Input buffer contains unread data (reading the buffer will clear this bit)

0 = Input buffer does not contain unread data

bit 7 **OBE:** Output Buffer Empty Status bit

1 = All readable Output Buffer registers are empty

0 = Some or all of the readable Output Buffer registers are full

bit 6 **OBUF:** Output Buffer Underflow Status bit

1 = A read occurred from an empty Output Buffer register (must be cleared in software)

0 = No underflow occurred

bit 5-4 **Unimplemented:** Read as '0'

bit 3-0 **OB3E:OB0E:** Output Buffer x Status Empty bits

1 = Output Buffer register is empty (writing data to the buffer will clear this bit)

0 = Output Buffer register contains untransmitted data

Note 1: Even though an individual bit represents the byte in the buffer, the bits corresponding to the word (Byte 0 and 1, or Byte 2 and 3) get cleared, even on byte reading.

REGISTER 21-9: PADCFG1: PAD CONFIGURATION CONTROL REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
_	_	_	_	_	_	_	PMPTTL
bit 7							bit 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 15-1 Unimplemented: Read as '0'

bit 0 PMPTTL: EPMP Module TTL Input Buffer Select bit

1 = EPMP module inputs (PMDx, PMCS1) use TTL input buffers

0 = EPMP module inputs use Schmitt Trigger input buffers

22.0 REAL-TIME CLOCK AND CALENDAR (RTCC)

Note:

This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information on the Real-Time Clock and Calendar, refer to the "dsPIC33/PIC24 Family Reference Manual", "RTCC with External Power Control" (DS39745).

The RTCC provides the user with a Real-Time Clock and Calendar (RTCC) function that can be calibrated.

Key features of the RTCC module are:

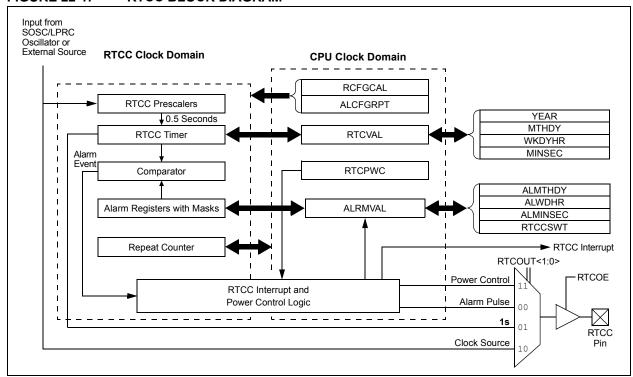
- · Operates in Deep Sleep mode
- · Selectable clock source
- Provides hours, minutes and seconds using 24-hour format
- · Visibility of one half second period
- Provides calendar weekday, date, month and year
- Alarm-configurable for half a second, one second, 10 seconds, one minute, 10 minutes, one hour, one day, one week, one month or one year
- · Alarm repeat with decrementing counter
- · Alarm with indefinite repeat chime
- · Year 2000 to 2099 leap year correction

- · BCD format for smaller software overhead
- · Optimized for long-term battery operation
- User calibration of the 32.768 kHz clock crystal/32K INTRC frequency with periodic auto-adjust
- Optimized for long-term battery operation
- · Fractional second synchronization
- Calibration to within ±2.64 seconds error per month
- · Calibrates up to 260 ppm of crystal error
- Ability to periodically wake up external devices without CPU intervention (external power control)
- · Power control output for external circuit control
- · Calibration takes effect every 15 seconds
- · Runs from any one of the following:
- External Real-Time Clock (RTC) of 32.768 kHz
- Internal 31.25 kHz LPRC clock
- 50 Hz or 60 Hz external input

22.1 RTCC Source Clock

The user can select between the SOSC crystal oscillator, LPRC internal oscillator or an external 50 Hz/60 Hz power line input as the clock reference for the RTCC module. This gives the user an option to trade off system cost, accuracy and power consumption, based on the overall system needs.

FIGURE 22-1: RTCC BLOCK DIAGRAM



22.2 RTCC Module Registers

The RTCC module registers are organized into three categories:

- · RTCC Control Registers
- · RTCC Value Registers
- · Alarm Value Registers

22.2.1 REGISTER MAPPING

To limit the register interface, the RTCC Timer and Alarm Time registers are accessed through corresponding register pointers. The RTCC Value register window (RTCVALH and RTCVALL) uses the RTCPTRx bits (RCFGCAL<9:8>) to select the desired Timer register pair (see Table 22-1).

By writing the RTCVALH byte, the RTCPTR<1:0> bits (the RTCC Pointer value) decrement by one until they reach '00'. Once they reach '00', the MINUTES and SECONDS value will be accessible through RTCVALH and RTCVALL until the pointer value is manually changed.

TABLE 22-1: RTCVAL REGISTER MAPPING

RTCPTR<1:0>	RTCC Value Register Window					
KICPIK 1.02	RTCVAL<15:8>	RTCVAL<7:0>				
0.0	MINUTES	SECONDS				
01	WEEKDAY	HOURS				
10	MONTH	DAY				
11	1	YEAR				

The Alarm Value register window (ALRMVALH and ALRMVALL) uses the ALRMPTRx bits (ALCFGRPT<9:8>) to select the desired Alarm register pair (see Table 22-2).

By writing the ALRMVALH byte, the ALRMPTR<1:0> bits (the Alarm Pointer value) decrement by one until they reach '00'. Once they reach '00', the ALRMMIN and ALRMSEC value will be accessible through ALRMVALH and ALRMVALL until the pointer value is manually changed.

TABLE 22-2: ALRMVAL REGISTER MAPPING

ALRMPTR	Alarm Value Register Window				
<1:0>	ALRMVAL<15:8>	ALRMVAL<7:0>			
00	ALRMMIN	ALRMSEC			
01	ALRMWD	ALRMHR			
10	ALRMMNTH	ALRMDAY			
11	_	_			

Considering that the 16-bit core does not distinguish between 8-bit and 16-bit read operations, the user must be aware that when reading either the ALRMVALH or ALRMVALL bytes, the ALRMPTR<1:0> value will be decremented. The same applies to the RTCVALH or RTCVALL bytes with the RTCPTR<1:0> being decremented.

Note: This only applies to read operations and not write operations.

22.2.2 WRITE LOCK

In order to perform a write to any of the RTCC Timer registers, the RTCWREN bit (RCFGCAL<13>) must be set (see Example 22-1).

Note: To avoid accidental writes to the timer, it is recommended that the RTCWREN bit (RCFGCAL<13>) is kept clear at any other time. For the RTCWREN bit to be set, there is only one instruction cycle time window allowed between the 55h/AA sequence and the setting of RTCWREN; therefore, it is recommended that code follow the procedure in Example 22-1.

22.2.3 SELECTING RTCC CLOCK SOURCE

The clock source for the RTCC module can be selected using the RTCLK<1:0> bits in the RTCPWC register. When the bits are set to '00', the Secondary Oscillator (SOSC) is used as the reference clock and when the bits are '01', LPRC is used as the reference clock. When RTCLK<1:0> = 10 and 11, the external power line (50 Hz and 60 Hz) is used as the clock source.

EXAMPLE 22-1: SETTING THE RTCWREN BIT

```
asm volatile("push w7");
asm volatile("push w8");
asm volatile("disi #5");
asm volatile("mov #0x55, w7");
asm volatile("mov w7, _NVMKEY");
asm volatile("mov #0xAA, w8");
asm volatile("mov w8, _NVMKEY");
asm volatile("mov w8, _NVMKEY");
asm volatile("bset _RCFGCAL, #13");  //set the RTCWREN bit
asm volatile("pop w8");
asm volatile("pop w7");
```

22.3 Registers

22.3.1 RTCC CONTROL REGISTERS

REGISTER 22-1: RCFGCAL: RTCC CALIBRATION/CONFIGURATION REGISTER⁽¹⁾

R/W-0	U-0	R/W-0	R-0, HSC	R-0, HSC	R/W-0	R/W-0	R/W-0
RTCEN ⁽²⁾	_	RTCWREN	RTCSYNC	HALFSEC ⁽³⁾	RTCOE	RTCPTR1	RTCPTR0
bit 15							bit 8

| R/W-0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| CAL7 | CAL6 | CAL5 | CAL4 | CAL3 | CAL2 | CAL1 | CAL0 |
| bit 7 | | | | | | | bit 0 |

Legend:	HSC = Hardware Settable/C	HSC = Hardware Settable/Clearable bit				
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 15 RTCEN: RTCC Enable bit⁽²⁾

1 = RTCC module is enabled 0 = RTCC module is disabled

bit 14 Unimplemented: Read as '0'

bit 13 RTCWREN: RTCC Value Registers Write Enable bit

1 = RTCVALH and RTCVALL registers can be written to by the user

0 = RTCVALH and RTCVALL registers are locked out from being written to by the user

bit 12 RTCSYNC: RTCC Value Registers Read Synchronization bit

1 = RTCVALH, RTCVALL and ALCFGRPT registers can change while reading due to a rollover ripple resulting in an invalid data read. If the register is read twice and results in the same data, the data can be assumed to be valid.

0 = RTCVALH, RTCVALL or ALCFGRPT registers can be read without concern over a rollover ripple

bit 11 HALFSEC: Half Second Status bit (3)

1 = Second half period of a second

0 = First half period of a second

bit 10 RTCOE: RTCC Output Enable bit

1 = RTCC output is enabled

0 = RTCC output is disabled

bit 9-8 RTCPTR<1:0>: RTCC Value Register Window Pointer bits

Points to the corresponding RTCC Value registers when reading the RTCVALH and RTCVALL registers. The RTCPTR<1:0> value decrements on every read or write of RTCVALH until it reaches '00'.

RTCVAL<15:8>:

11 = Reserved

10 = MONTH

01 = WEEKDAY

00 = MINUTES

RTCVAL<7:0>:

11 = YEAR

10 **= DAY**

01 = HOURS

00 = SECONDS

Note 1: The RCFGCAL register is only affected by a POR.

2: A write to the RTCEN bit is only allowed when RTCWREN = 1.

3: This bit is read-only; it is cleared to '0' on a write to the lower half of the MINSEC register.

REGISTER 22-1: RCFGCAL: RTCC CALIBRATION/CONFIGURATION REGISTER⁽¹⁾ (CONTINUED)

bit 7-0

CAL<7:0>: RTC Drift Calibration bits

01111111 = Maximum positive adjustment; adds 127 RTC clock pulses every 15 seconds

.

00000001 = Minimum positive adjustment; adds 1 RTC clock pulse every 15 seconds

00000000 = No adjustment

11111111 = Minimum negative adjustment; subtracts 1 RTC clock pulse every 15 seconds

.

10000000 = Maximum negative adjustment; subtracts 128 RTC clock pulses every 15 seconds

- Note 1: The RCFGCAL register is only affected by a POR.
 - 2: A write to the RTCEN bit is only allowed when RTCWREN = 1.
 - 3: This bit is read-only; it is cleared to '0' on a write to the lower half of the MINSEC register.

REGISTER 22-2: RTCPWC: RTCC POWER CONTROL REGISTER(1)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PWCEN	PWCPOL	PWCPRE	PWSPRE	RTCLK1 ⁽²⁾	RTCLK0 ⁽²⁾	RTCOUT1	RTCOUT0
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 7							bit 0

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

bit 15 **PWCEN:** Power Control Enable bit

1 = Power control is enabled

0 = Power control is disabled

bit 14 **PWCPOL:** Power Control Polarity bit

1 = Power control output is active-high

0 = Power control output is active-low

bit 13 **PWCPRE:** Power Control/Stability Prescaler bit

1 = PWC stability window clock is divide-by-2 of source RTCC clock

0 = PWC stability window clock is divide-by-1 of source RTCC clock

bit 12 **PWSPRE:** Power Control Sample Prescaler bit

1 = PWC sample window clock is divide-by-2 of source RTCC clock

0 = PWC sample window clock is divide-by-1 of source RTCC clock

bit 11-10 RTCLK<1:0>: RTCC Clock Source Select bits⁽²⁾

11 = External power line (60 Hz)

10 = External power line source (50 Hz)

01 = Internal LPRC Oscillator

00 = External Secondary Oscillator (SOSC)

bit 9-8 RTCOUT<1:0>: RTCC Output Source Select bits

11 = Power control

10 = RTCC clock

01 = RTCC seconds clock

00 = RTCC alarm pulse

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

Note 1: The RTCPWC register is only affected by a POR.

When a new value is written to these register bits, the lower half of the MINSEC register should also be written to properly reset the clock prescalers in the RTCC.

REGISTER 22-3: ALCFGRPT: ALARM CONFIGURATION REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ALRMEN	CHIME	AMASK3	AMASK2	AMASK1	AMASK0	ALRMPTR1	ALRMPTR0
bit 15							bit 8

| R/W-0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| ARPT7 | ARPT6 | ARPT5 | ARPT4 | ARPT3 | ARPT2 | ARPT1 | ARPT0 |
| bit 7 | | | | | | | bit 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 15 ALRMEN: Alarm Enable bit

1 = Alarm is enabled (cleared automatically after an alarm event whenever ARPT<7:0> = 00h and CHIME = 0)

0 = Alarm is disabled

bit 14 **CHIME:** Chime Enable bit

1 = Chime is enabled; ARPT<7:0> bits are allowed to roll over from 00h to FFh

0 = Chime is disabled; ARPT<7:0> bits stop once they reach 00h

bit 13-10 AMASK<3:0>: Alarm Mask Configuration bits

0000 = Every half second

0001 = Every second

0010 = Every 10 seconds

0011 = Every minute

0100 = Every 10 minutes

0101 = Every hour

0110 = Once a day

0111 = Once a week

1000 = Once a month

1001 = Once a year (except when configured for February 29th, once every 4 years)

101x = Reserved – do not use

11xx = Reserved – do not use

bit 9-8 **ALRMPTR<1:0>:** Alarm Value Register Window Pointer bits

Points to the corresponding Alarm Value registers when reading the ALRMVALH and ALRMVALL registers. The ALRMPTR<1:0> value decrements on every read or write of ALRMVALH until it reaches '00'.

ALRMVAL<15:8>:

00 = ALRMMIN

01 = ALRMWD

10 = ALRMMNTH

11 = PWCSTAB

ALRMVAL<7:0>:

00 = ALRMSEC

01 = ALRMHR

10 = ALRMDAY

11 = PWCSAMP

bit 7-0 ARPT<7:0>: Alarm Repeat Counter Value bits

11111111 = Alarm will repeat 255 more times

•

00000000 = Alarm will not repeat

The counter decrements on any alarm event; it is prevented from rolling over from 00h to FFh unless CHIME = 1.

22.3.2 RTCVAL REGISTER MAPPINGS

REGISTER 22-4: YEAR: YEAR VALUE REGISTER(1)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

| R/W-x |
|--------|--------|--------|--------|--------|--------|--------|--------|
| YRTEN3 | YRTEN2 | YRTEN1 | YRTEN0 | YRONE3 | YRONE2 | YRONE1 | YRONE0 |
| bit 7 | | | | | | | bit 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 15-8 Unimplemented: Read as '0'

bit 7-4 YRTEN<3:0>: Binary Coded Decimal Value of Year's Tens Digit bits

Contains a value from 0 to 9.

bit 3-0 YRONE<3:0>: Binary Coded Decimal Value of Year's Ones Digit bits

Contains a value from 0 to 9.

Note 1: A write to the YEAR register is only allowed when RTCWREN = 1.

REGISTER 22-5: MTHDY: MONTH AND DAY VALUE REGISTER(1)

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
_	_	_	MTHTEN0	MTHONE3	MTHONE2	MTHONE1	MTHONE0
bit 15							bit 8

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
_	_	DAYTEN1	DAYTEN0	DAYONE3	DAYONE2	DAYONE1	DAYONE0
bit 7							bit 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 15-13 **Unimplemented:** Read as '0'

bit 12 MTHTEN0: Binary Coded Decimal Value of Month's Tens Digit bit

Contains a value of '0' or '1'.

bit 11-8 MTHONE<3:0>: Binary Coded Decimal Value of Month's Ones Digit bits

Contains a value from 0 to 9.

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

bit 5-4 DAYTEN<1:0>: Binary Coded Decimal Value of Day's Tens Digit bits

Contains a value from 0 to 3.

bit 3-0 DAYONE<3:0>: Binary Coded Decimal Value of Day's Ones Digit bits

Contains a value from 0 to 9.

Note 1: A write to this register is only allowed when RTCWREN = 1.

REGISTER 22-6: WKDYHR: WEEKDAY AND HOURS VALUE REGISTER⁽¹⁾

U-0	U-0	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x
_	_	_	_	_	WDAY2	WDAY1	WDAY0
bit 15							bit 8

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
_	_	HRTEN1	HRTEN0	HRONE3	HRONE2	HRONE1	HRONE0
bit 7							bit 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 15-11 **Unimplemented:** Read as '0'

bit 10-8 WDAY<2:0>: Binary Coded Decimal Value of Weekday Digit bits

Contains a value from 0 to 6.

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

bit 5-4 HRTEN<1:0>: Binary Coded Decimal Value of Hour's Tens Digit bits

Contains a value from 0 to 2.

bit 3-0 HRONE<3:0>: Binary Coded Decimal Value of Hour's Ones Digit bits

Contains a value from 0 to 9.

Note 1: A write to this register is only allowed when RTCWREN = 1.

REGISTER 22-7: MINSEC: MINUTES AND SECONDS VALUE REGISTER

U-0	R/W-x						
_	MINTEN2	MINTEN1	MINTEN0	MINONE3	MINONE2	MINONE1	MINONE0
bit 15							bit 8

U-0	R/W-x						
_	SECTEN2	SECTEN1	SECTEN0	SECONE3	SECONE2	SECONE1	SECONE0
bit 7							bit 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 15 **Unimplemented:** Read as '0'

bit 14-12 MINTEN<2:0>: Binary Coded Decimal Value of Minute's Tens Digit bits

Contains a value from 0 to 5.

bit 11-8 MINONE<3:0>: Binary Coded Decimal Value of Minute's Ones Digit bits

Contains a value from 0 to 9.

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

bit 6-4 SECTEN<2:0>: Binary Coded Decimal Value of Second's Tens Digit bits

Contains a value from 0 to 5.

bit 3-0 SECONE<3:0>: Binary Coded Decimal Value of Second's Ones Digit bits

Contains a value from 0 to 9.

22.3.3 ALRMVAL REGISTER MAPPINGS

REGISTER 22-8: ALMTHDY: ALARM MONTH AND DAY VALUE REGISTER(1)

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
_	_	_	MTHTEN0	MTHONE3	MTHONE2	MTHONE1	MTHONE0
bit 15							bit 8

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
_	_	DAYTEN1	DAYTEN0	DAYONE3	DAYONE2	DAYONE1	DAYONE0
bit 7							bit 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 15-13 **Unimplemented:** Read as '0'

bit 12 MTHTEN0: Binary Coded Decimal Value of Month's Tens Digit bit

Contains a value of '0' or '1'.

bit 11-8 MTHONE<3:0>: Binary Coded Decimal Value of Month's Ones Digit bits

Contains a value from 0 to 9.

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

bit 5-4 DAYTEN<1:0>: Binary Coded Decimal Value of Day's Tens Digit bits

Contains a value from 0 to 3.

bit 3-0 DAYONE<3:0>: Binary Coded Decimal Value of Day's Ones Digit bits

Contains a value from 0 to 9.

Note 1: A write to this register is only allowed when RTCWREN = 1.

REGISTER 22-9: ALWDHR: ALARM WEEKDAY AND HOURS VALUE REGISTER(1)

U-0	U-0	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x
_	_	_	_	_	WDAY2	WDAY1	WDAY0
bit 15							bit 8

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
_	_	HRTEN1	HRTEN0	HRONE3	HRONE2	HRONE1	HRONE0
bit 7							bit 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 15-11 **Unimplemented:** Read as '0'

bit 10-8 WDAY<2:0>: Binary Coded Decimal Value of Weekday Digit bits

Contains a value from 0 to 6.

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

bit 5-4 HRTEN<1:0>: Binary Coded Decimal Value of Hour's Tens Digit bits

Contains a value from 0 to 2.

bit 3-0 **HRONE<3:0>:** Binary Coded Decimal Value of Hour's Ones Digit bits

Contains a value from 0 to 9.

Note 1: A write to this register is only allowed when RTCWREN = 1.

REGISTER 22-10: ALMINSEC: ALARM MINUTES AND SECONDS VALUE REGISTER

U-0	R/W-x						
_	MINTEN2	MINTEN1	MINTEN0	MINONE3	MINONE2	MINONE1	MINONE0
bit 15							bit 8

U-0	R/W-x						
_	SECTEN2	SECTEN1	SECTEN0	SECONE3	SECONE2	SECONE1	SECONE0
bit 7							bit 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 15 **Unimplemented:** Read as '0'

bit 14-12 MINTEN<2:0>: Binary Coded Decimal Value of Minute's Tens Digit bits

Contains a value from 0 to 5.

bit 11-8 MINONE<3:0>: Binary Coded Decimal Value of Minute's Ones Digit bits

Contains a value from 0 to 9.

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

bit 6-4 SECTEN<2:0>: Binary Coded Decimal Value of Second's Tens Digit bits

Contains a value from 0 to 5.

bit 3-0 SECONE<3:0>: Binary Coded Decimal Value of Second's Ones Digit bits

Contains a value from 0 to 9.

REGISTER 22-11: RTCCSWT: RTCC POWER CONTROL AND SAMPLE WINDOW TIMER REGISTER⁽¹⁾

| R/W-x |
|----------|----------|----------|----------|----------|----------|----------|----------|
| PWCSTAB7 | PWCSTAB6 | PWCSTAB5 | PWCSTAB4 | PWCSTAB3 | PWCSTAB2 | PWCSTAB1 | PWCSTAB0 |
| bit 15 | | | | | | | bit 8 |

	R/W-x							
Ī	PWCSAMP7 ⁽²⁾	PWCSAMP6 ⁽²⁾	PWCSAMP5 ⁽²⁾	PWCSAMP4 ⁽²⁾	PWCSAMP3 ⁽²⁾	PWCSAMP2 ⁽²⁾	PWCSAMP1 ⁽²⁾	PWCSAMP0 ⁽²⁾
	bit 7							bit 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		

- **Note 1:** A write to this register is only allowed when RTCWREN = 1.
 - 2: The sample window always starts when the stability window timer expires, except when its initial value is 00h.

22.4 Calibration

The real-time crystal input can be calibrated using the periodic auto-adjust feature. When properly calibrated, the RTCC can provide an error of less than 3 seconds per month. This is accomplished by finding the number of error clock pulses and storing the value into the lower half of the RCFGCAL register. The 8-bit signed value loaded into the lower half of RCFGCAL is multiplied by four and will either be added or subtracted from the RTCC timer, once every minute. Refer to the steps below for RTCC calibration:

- Using another timer resource on the device, the user must find the error of the 32.768 kHz crystal.
- 2. Once the error is known, it must be converted to the number of error clock pulses per minute.
- a) If the oscillator is faster than ideal (negative result form Step 2), the RCFGCAL register value must be negative. This causes the specified number of clock pulses to be subtracted from the timer counter, once every minute.
 - b) If the oscillator is slower than ideal (positive result from Step 2), the RCFGCAL register value must be positive. This causes the specified number of clock pulses to be subtracted from the timer counter, once every minute.

EQUATION 22-1:

(Ideal Frequency† – Measured Frequency) * 60 = Clocks per Minute

† Ideal Frequency = 32,768 Hz

Writes to the lower half of the RCFGCAL register should only occur when the timer is turned off, or immediately after the rising edge of the seconds pulse, except when SECONDS = 00, 15, 30 or 45. This is due to the auto-adjust of the RTCC at 15 second intervals.

Note: It is up to the user to include, in the error value, the initial error of the crystal: drift due to temperature and drift due to crystal aging.

22.5 Alarm

- · Configurable from half second to one year
- Enabled using the ALRMEN bit (ALCFGRPT<15>)
- One-time alarm and repeat alarm options available

22.5.1 CONFIGURING THE ALARM

The alarm feature is enabled using the ALRMEN bit. This bit is cleared when an alarm is issued. Writes to ALRMVAL should only take place when ALRMEN = 0.

As shown in Figure 22-2, the interval selection of the alarm is configured through the AMASKx bits (ALCFGRPT<13:10>). These bits determine which and how many digits of the alarm must match the clock value for the alarm to occur.

The alarm can also be configured to repeat based on a preconfigured interval. The amount of times this occurs, once the alarm is enabled, is stored in the ARPT<7:0> bits (ALCFGRPT<7:0>). When the value of the ARPTx bits equals 00h and the CHIME bit (ALCFGRPT<14>) is cleared, the repeat function is disabled and only a single alarm will occur. The alarm can be repeated, up to 255 times, by loading ARPT<7:0> with FFh.

After each alarm is issued, the value of the ARPTx bits is decremented by one. Once the value has reached 00h, the alarm will be issued one last time, after which, the ALRMEN bit will be cleared automatically and the alarm will turn off.

Indefinite repetition of the alarm can occur if the CHIME bit = 1. Instead of the alarm being disabled when the value of the ARPTx bits reaches 00h, it rolls over to FFh and continues counting indefinitely while CHIME is set.

22.5.2 ALARM INTERRUPT

At every alarm event, an interrupt is generated. In addition, an alarm pulse output is provided that operates at half the frequency of the alarm. This output is completely synchronous to the RTCC clock and can be used as a trigger clock to other peripherals.

Note: Changing any of the registers, other than the RCFGCAL and ALCFGRPT registers, and the CHIME bit while the alarm is enabled (ALRMEN = 1), can result in a false alarm event leading to a false alarm interrupt. To avoid a false alarm event, the timer and alarm values should only be changed while the alarm is disabled (ALRMEN = 0). It is recommended that the ALCFGRPT register and CHIME bit be changed when RTCSYNC = 0.

FIGURE 22-2: ALARM MASK SETTINGS

Alarm Mask Setting (AMASK<3:0>)	Day of the Week	Month Day	Hours Minutes Seconds
0000 - Every half second 0001 - Every second			
0010 - Every 10 seconds			: s
0011 - Every minute			: s s
0100 - Every 10 minutes			m:ss
0101 - Every hour			; m m; s s
0110 - Every day			h h : m m : s s
0111 - Every week	d		h h m m sss
1000 - Every month		/ d d	h h m m sss
1001 - Every year ⁽¹⁾		m m / d d	h h m m sss
Note 1: Annually, except when co	nfigured fo	or February 29.	

22.6 Power Control

The RTCC includes a power control feature that allows the device to periodically wake-up an external device, wait for the device to be stable before sampling wake-up events from that device and then shut down the external device. This can be done completely autonomously by the RTCC, without the need to wake from the current lower power mode (Sleep, Deep Sleep, etc.).

To use this feature:

- 1. Enable the RTCC (RTCEN = 1).
- 2. Set the PWCEN bit (RTCPWC<15>).
- 3. Configure the RTCC pin to drive the PWC control signal (RTCOE = 1 and RTCOUT<1:0> = 11).

The polarity of the PWC control signal may be chosen using the PWCPOL bit (RTCPWC<14>). An active-low or active-high signal may be used with the appropriate external switch to turn on or off the power to one or

more external devices. The active-low setting may also be used in conjunction with an open-drain setting on the RTCC pin, in order to drive the ground pin(s) of the external device directly (with the appropriate external VDD pull-up device), without the need for external switches. Finally, the CHIME bit should be set to enable the PWC periodicity.

22.7 RTCC VBAT Operation

The RTCC can operate in VBAT mode when there is a power loss on the VDD pin. The RTCC will continue to operate if the VBAT pin is powered on (it is usually connected to the battery).

Note: It is recommended to connect the VBAT pin to VDD if the VBAT mode is not used (not connected to the battery).

PICZ4FJ	IZOGDA	204 FA	IVIIL Y		
NOTES:					

23.0 CRYPTOGRAPHIC ENGINE

Note:

This data sheet summarizes the features of the PIC24FJ128GB204 family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to the "dsPIC33/PIC24 Family Reference Manual", "Cryptographic Engine" (DS70005133), which is available from the Microchip web site (www.microchip.com).

The Cryptographic Engine provides a new set of data security options. Using its own free-standing state machines, the engine can independently perform NIS standard encryption and decryption of data independently of the CPU. This eliminates the concerns of excessive CPU or program memory overhead that encryption and decryption would otherwise require, while enhancing the application's security.

The primary features of the Cryptographic Engine are:

- Memory-mapped 128-bit and 256-bit memory spaces for encryption/decryption data
- Multiple options for key storage, selection and management

- · Support for internal context saving
- · Session key encryption and loading
- · Half-duplex operation
- DES and Triple DES (3DES) encryption and decryption (64-bit block size):
 - Supports 64-bit keys and 2-key or 3-key Triple DES
- AES encryption and decryption (128-bit block size):
 - Supports key sizes of 128, 192 or 256 bits
- Supports ECB, CBC, CFB, OFB and CTR modes for both DES and AES standards
- · Programmatically secure key storage:
 - 512-bit OTP array for key storage, not readable from other memory spaces
 - 32-bit Configuration Page
 - Simple in-module programming interface
 - Supports Key Encryption Key (KEK)
- Support for True and Psuedorandom Number Generation (PRNG), NIST SP800-90 compliant

A simplified block diagram of the Cryptographic Engine is shown in Figure 23-1.

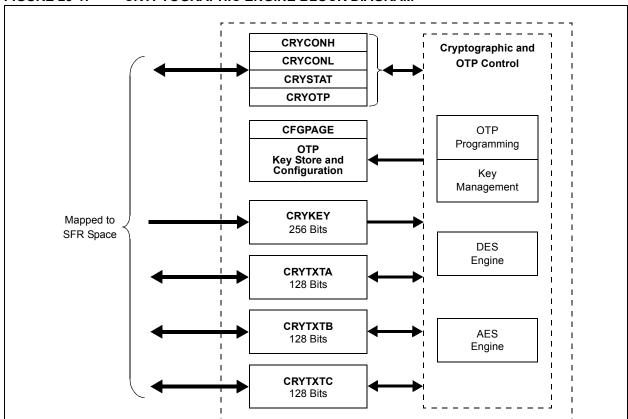


FIGURE 23-1: CRYPTOGRAPHIC ENGINE BLOCK DIAGRAM

23.1 Data Register Spaces

There are four register spaces used for cryptographic data and key storage:

- CRYTXTA
- CRYTXTB
- CRYTXTC
- CRYKEY

Although mapped into the SFR space, all of these Data Spaces are actually implemented as 128-bit or 256-bit wide arrays, rather than groups of 16-bit wide Data registers. Reads and writes to and from these arrays are automatically handled as if they were any other register in the SFR space.

CRYTXTA through CRYTXTC are 128-bit wide spaces; they are used for writing data to and reading from the Cryptographic Engine. Additionally, they are also used for storing intermediate results of the encryption/decryption operation. None of these registers may be written to when the module is performing an operation (CRYGO = 1).

CRYTXTA and CRYTXTB normally serve as inputs to the encryption/decryption process.

CRYTXTA usually contains the initial plaintext or ciphertext to be encrypted or decrypted. Depending on the mode of operation, CRYTXTB may contain the ciphertext output or intermediate cipher data. It may also function as a programmable length counter in certain operations.

CRYTXTC is primarily used to store the final output of an encryption/decryption operation. It is also used as the input register for data to be programmed to the secure OTP array.

CRYKEY is a 256-bit wide space, used to store cryptographic keys for the selected operation. It is writable from both the SFR space and the secure OTP array. Although mapped into the SFR space, it is a write-only memory area; any data placed here, regardless of its source, cannot be read back by any run-time operations. This feature helps to ensure the security of any key data.

23.2 Modes of Operation

The Cryptographic Engine supports the following modes of operation, determined by the OPMOD<3:0> bits:

- · Block Encryption
- · Block Decryption
- AES Decryption Key Expansion
- Random Number Generation
- · Session Key Generation
- · Session Key Encryption
- · Session Key Loading

The OPMODx bits may be changed while CRYON is set. They should only be changed when a cryptographic operation is not being done (CRYGO = 0).

Once the encryption operation, and the appropriate and valid key configuration is selected, the operation is performed by setting the CRYGO bit. This bit is automatically

cleared by hardware when the operation is complete. The CRYGO bit can also be manually cleared by software; this causes any operation in progress to terminate immediately. Clearing this bit in software also sets the CRYABRT bit (CRYSTAT<5>).

For most operations, CRYGO can only be set when an OTP operation is not being performed and there are no other error conditions. CRYREAD, CRYWR, CRYABRT, ROLLOVR, MODFAIL and KEYFAIL must all be '0'.

Setting CRYWR and CRYGO simultaneously will not initiate an OTP programming operation or any other operation. Setting CRYGO when the module is disabled (CRYON = 0) also has no effect.

23.3 Enabling the Engine

The Cryptographic Engine is enabled by setting the CRYON bit. Clearing this bit disables both the DES and AES engines, as well as causing the following register bits to be held in Reset:

- CRYGO (CRYCONL<8>)
- TXTABSY (CRYSTAT<6>)
- CRYWR (CRYOTP<0>)

All other register bits and registers may be read and written while CRYON = 0.

23.4 Operation During Sleep and Idle Modes

23.4.1 OPERATION DURING SLEEP MODES

Whenever the device enters any Sleep or Deep Sleep mode, all operation engine state machines are reset. This feature helps to preserve the integrity, or any data being encrypted or decrypted, by discarding any intermediate text that might be used to break the key.

Any OTP programming operations under way when a Sleep mode is entered are also halted. Depending on what is being programmed, this may result in permanent loss of a memory location or potentially, the use of the entire secure OTP array. Users are advised to perform OTP programming only when entry into power-saving modes is disabled.

Note: OTP programming errors, regardless of the source, are not recoverable errors. Users should ensure that all foreseeable interruptions to the programming operation, including device interrupts and entry into power-saving modes, are disabled.

23.4.2 OPERATION DURING IDLE MODE

When the CRYSIDL bit (CRYCONL<13>) is '0', the engine will continue any ongoing operations without interruption when the device enters Idle mode.

When CRYSIDL is '1', the module behaves as in Sleep modes.

23.5 Specific Cryptographic Operations

This section provides the step-wise details for each operation type that is available with the Cryptographic Engine.

23.5.1 ENCRYPTING DATA

- 1. If not already set, set the CRYON bit.
- 2. Configure the CPHRSEL, CPHRMODx, KEYMODx and KEYSRCx bits as desired to select the proper mode and key length.
- 3. Set OPMOD<3:0> to '0000'.
- 4. If a software key is being used, write it to the CRYKEY register. It is only necessary to write the lowest n bits of CRYKEY for a key length of n, as all unused CRYKEY bits are ignored.
- Read the KEYFAIL bit. If this bit is '1', an illegal configuration has been selected and the encrypt operation will NOT be performed.
- 6. Write the data to be encrypted to the appropriate CRYTXT register. For a single DES encrypt operation, it is only necessary to write the lowest 64 bits. However, for data less than the block size (64 bits for DES, 128 bits for AES), it is the responsibility of the software to properly pad the upper bits within the block.
- 7. Set the CRYGO bit.
- 8. In ECB and CBC modes, set the FREEIE bit (CRYCONL<10>) to enable the optional CRYTXTA interrupt to indicate when the next plaintext block can be loaded.
- Poll the CRYGO bit until it is cleared or wait for the CRYDNIF module interrupt (DONEIE must be set). If other Cryptographic Engine interrupts are enabled, it will be necessary to poll the CRYGO bit to verify the interrupt source.
- Read the encrypted block from the appropriate CRYTXT register.
- 11. Repeat Steps 5 through 8 to encrypt further blocks in the message with the same key.

23.5.2 DECRYPTING DATA

- 1. If not already set, set the CRYON bit.
- 2. Configure the CPHRSEL, CPHRMODx, KEYMODx and KEYSRCx bits as desired to select the proper mode and key length.
- 3. Set OPMOD<3:0> to '0001'.
- 4. If a software key is being used, write the CRYKEY register. It is only necessary to write the lowest n bits of CRYKEY for a key length of n, as all unused CRYKEY bits are ignored.
- 5. If an AES-ECB or AES-CBC mode decryption is being performed, you must first perform an AES decryption key expansion operation.
- Read the KEYFAIL status bit. If this bit is '1', an illegal configuration has been selected and the encrypt operation will not be performed.
- Write the data to be decrypted into the appropriate text/data register. For a DES decrypt operation, it is only necessary to write the lowest 64 bits of CRYTXTB.
- 8. Set the CRYGO bit.
- 9. If this is the first decrypt operation after a Reset, or if a key storage program operation was performed after the last decrypt operation, or if the KEYMODx or KEYSRCx fields are changed, the engine will perform a new key expansion operation. This will result in extra clock cycles for the decrypt operation, but will otherwise be transparent to the application (i.e., the CRYGO bit will be cleared only after the key expansion and the decrypt operation have completed).
- In ECB and CBC modes, set the FREEIE bit (CRYCONL<10>) to enable the optional CRYTXTA interrupt to indicate when the next plaintext block can be loaded.
- 11. Poll the CRYGO bit until it is cleared or wait for the CRYDNIF module interrupt (DONEIE must be set). If other Cryptographic Engine interrupts are enabled, it will be necessary to poll the CRYGO bit to verify the interrupt source.
- 12. Read the decrypted block out of the appropriate text/data register.
- 13. Repeat Steps 6 through 10 to encrypt further blocks in the message with the same key.

23.5.3 ENCRYPTING A SESSION KEY

Note: ECB and CBC modes are restricted to 128-bit session keys only.

- 1. If not already set, set the CRYON bit.
- 2. If not already programmed, program the SKEYEN bit to '1'.

Note: Setting SKEYEN permanently makes Key #1 available as a Key Encryption Key only. It cannot be used for other encryption or decryption operations after that.

- 3. Set OPMOD<3:0> to '1110'.
- Configure the CPHRSEL, CPHRMOD<2:0> and KEYMOD<1:0> register bit fields as desired, set SKEYSEL to '0'.
- 5. Read the KEYFAIL status bit. If this bit is '1', an illegal configuration has been selected, and the encrypt operation will not be performed.
- Write the software generated session key into the CRYKEY register or generate a random key into the CRYKEY register. It is only necessary to write the lowest n bits of CRYKEY for a key length of n, as all unused key bits are ignored.
- Set the CRYGO bit. Poll the bit until it is cleared by hardware; alternatively, set the DONEIE bit (CRYCONL<11>) to generate an interrupt when the encryption is done.
- 8. Read the encrypted session key out of the appropriate CRYTXT register.
- 9. For total key lengths of more than 128 bits, set SKEYSEL to '1' and repeat Steps 6 and 7.
- 10. Set KEYSRC<3:0> to '0000' to use the session key to encrypt data.

23.5.4 RECEIVING A SESSION KEY

Note: ECB and CBC modes are restricted to 128-bit session keys only.

- 1. If not already set, set the CRYON bit.
- 2. If not already programmed, program the SKEYEN bit to '1'.

Note: Setting SKEYEN permanently makes Key #1 available as a Key Encryption Key only. It cannot be used for other encryption or decryption operations after that. It also permanently disables the ability of software to decrypt the session key into the CRYTXTA register, thereby breaking programmatic security (i.e., software can read the unencrypted key).

- 3. Set OPMOD<3:0> to '1111'.
- Configure the CPHRSEL, CPHRMOD
 and KEYMOD
 register bit fields as desired, set SKEYSEL to '0'.
- Read the KEYFAIL status bit. If this bit is '1', an illegal configuration has been selected and the encrypt operation will NOT be performed.
- 6. Write the encrypted session key received into the appropriate CRYTXT register.
- Set the CRYGO bit. Poll the bit until it is cleared by hardware; alternatively, set the DONEIE bit (CRYCONL<11>) to generate an interrupt when the process is done.
- 8. For total key lengths of more than 128 bits, set SKEYSEL to '1' and repeat Steps 6 and 7.
- Set KEYSRC<3:0> to '0000' to use the newly generated session key to encrypt and decrypt data.

23.5.5 GENERATING A PSEUDORANDOM NUMBER (PRN)

For operations that require a Pseudorandom Number (PRN), the method outlined in NIST SP800-90 can be adapted for efficient use with the Cryptographic Engine. This method uses the AES algorithm in CTR mode to create PRNs with minimal CPU overhead. PRNs generated in this manner can be used for cryptographic purposes or any other purpose that the host application may require.

The random numbers used as initial seeds can be taken from any source convenient to the user's application. If possible, a non-deterministic random number source should be used.

Note: PRN generation is not available when software keys are disabled (SWKYDIS = 1).

To perform the initial reseeding operation, and subsequent reseedings after the reseeding interval has expired:

- 1. Store a random number (128 bits) in CRYTXTA.
- 2. For the initial generation ONLY, use a key value of 0h (128 bits), and a counter value of 0h.
- Configure the engine for AES encryption, CTR mode (OPMOD<3:0> = 0000, CPHRSEL = 1, CPHMOD<2:0> = 100).
- Perform an encrypt operation by setting CRYGO.
- 5. Move the results in CRYTXTC to RAM. This is the new key value (NEW KEY).
- Store another random number (128 bits) in CRYTXTA.
- 7. Configure the module for encryption as in Step 3.
- Perform an encrypt operation by setting CRYGO.
- Store this value in RAM. This is the new counter value (NEW CTR).
- For subsequent reseeding operations, use NEW_KEY and NEW_CTR for the starting key and counter values.

To generate the pseudorandom number:

- 1. Load NEW KEY value from RAM into CRYKEY.
- 2. Load NEW CTR value from RAM into CRYTXTB.
- 3. Load CRYTXTA with 0h (128 bits).
- 4. Configure the engine for AES encryption, CTR mode (OPMOD<3:0> = 0000, CPHRSEL = 1, CPHMOD<2:0> = 100).
- 5. Perform an encrypt operation by setting CRYGO.
- 6. Copy the generated PRN in CRYTXTC (PRNG_VALUE) to RAM.
- 7. Repeat the encrypt operation.
- 8. Store the value of CRYTXTC from this round as the new value of NEW KEY.
- 9. Repeat the encrypt operation.
- Store the value of CRYTXTC from this round as the new value of NEW CTR.

Subsequent PRNs can be generated by repeating this procedure until the reseeding interval has expired. At that point, the reseeding operation is performed using the stored values of NEW KEY and NEW CTR.

23.5.6 GENERATING A RANDOM NUMBER

- Enable the Cryptographic mode (CRYON (CRYCONL<15>) = 1).
- 2. Set the OPMOD<3:0> bits to '1010'.
- 3. Start the request by setting the CRYGO bit (CRYCONL<8>) to '1'.
- Wait for the CRYGO bit to be cleared to '0' by the hardware.
- Read the random number from the CRYTXTA register.

23.5.7 TESTING THE KEY SOURCE CONFIGURATION

The validity of the key source configuration can always be tested by writing the appropriate register bits and then reading the KEYFAIL register bit. No operation needs to be started to perform this check; the module does not even need to be enabled.

23.5.8 PROGRAMMING CFGPAGE (PAGE 0) CONFIGURATION BITS

- If not already set, set the CRYON bit. Set KEYPG<3:0> to '0000'.
- Read the PGMFAIL status bit. If this bit is '1', an illegal configuration has been selected and the programming operation will not be performed.
- Write the data to be programmed into the Configuration Page into CRYTXTC<31:0>. Any bits that are set ('1') will be permanently programmed, while any bits that are cleared ('0') will not be programmed and may be programmed at a later time.
- 4. Set the CRYWR bit. Poll the bit until it is cleared; alternatively, set the OTPIE bit (CRYOTP<6>) to enable the optional OTP done interrupt.
- Once all programming has completed, set the CRYREAD bit to reload the values from the on-chip storage. A read operation must be performed to complete programming.

Note: Do not clear the CRYON bit while the CRYREAD bit is set; this will result in an incomplete read operation and unavailable key data. To recover, set CRYON and CRYREAD, and allow the read operation to fully complete.

- 6. Poll the CRYREAD bit until it is cleared; alternatively, set the OTPIE bit (CRYOTP<6>) to enable the optional OTP done interrupt.
- For production programming, the TSTPGM bit can be set to indicate a successful programming operation. When TSTPGM is set, the PGMTST bit (CRYOTP<7>) will also be set, allowing users to see the OTP array status with performing a read operation on the array.

Note: If the device enters Sleep mode during OTP programming, the contents of the OTP array may become corrupted. This is not a recoverable error. Users must ensure that entry into power-saving modes is disabled before OTP programming is performed.

23.5.9 PROGRAMMING KEYS

- 1. If not already set, set the CRYON bit.
- Configure KEYPG<3:0> to the page you want to program.
- 3. Read the PGMFAIL status bit. If this bit is '1', an illegal configuration has been selected and the programming operation will not be performed.
- 4. Write the data to be programmed into the Configuration Page into CRYTXTC<63:0>. Any bits that are set ('1') will be permanently programmed, while any bits that are cleared ('0') will not be programmed and may be programmed at a later time.
- 5. Set the CRYWR bit. Poll the bit until it is cleared; alternatively, set the OTPIE bit (CRYOTP<6>) to enable the optional OTP done interrupt.
- 6. Repeat Steps 2 through 5 for each OTP array page to be programmed.
- Once all programming has completed, set the CRYREAD bit to reload the values from the on-chip storage. A read operation must be performed to complete programming.

Note: Do not clear the CRYON bit while the CRYREAD bit is set; this will result in an incomplete read operation and unavailable key data. To recover, set CRYON and CRYREAD, and allow the read operation to fully complete.

- Poll the CRYREAD bit until it is cleared; alternatively, set the OTPIE bit (CRYOTP<6>) to enable the optional OTP done interrupt.
- For production programming, the TSTPGM bit can be set to indicate a successful programming operation. When TSTPGM is set, the PGMTST bit (CRYOTP<7>) will also be set, allowing users to see the OTP array status with performing a read operation on the array.

Note: If the device enters Sleep Mode during OTP programming, the contents of the OTP array may become corrupted. This is not a recoverable error. Users must ensure that entry into power-saving modes is disabled before OTP programming is performed.

23.5.10 VERIFYING PROGRAMMED KEYS

To maintain key security, the secure OTP array has no provision to read back its data to any user-accessible memory space in any operating mode. Therefore, there is no way to directly verify programmed data. The only method for verifying that they have been programmed correctly is to perform an encryption operation with a known plaintext/ciphertext pair for each programmed key.

23.6 Control Registers

REGISTER 23-1: CRYCONL: CRYPTOGRAPHIC CONTROL LOW REGISTER

R/W-0	U-0	R/W-0	R/W-0 ⁽¹⁾	R/W-0 ⁽¹⁾	R/W-0 ⁽¹⁾	U-0	R/W-0, HC ⁽¹⁾
CRYON	_	CRYSIDL ⁽³⁾	ROLLIE	DONEIE	FREEIE	_	CRYGO
bit 15							bit 8

R/W-0 ⁽¹⁾	R/W-0 ⁽¹⁾	R/W-0 ⁽¹⁾	R/W-0 ⁽¹⁾				
OPMOD3 ⁽²⁾	OPMOD2 ⁽²⁾	OPMOD1 ⁽²⁾	OPMOD0 ⁽²⁾	CPHRSEL ⁽²⁾	CPHRMOD2 ⁽²⁾	CPHRMOD1 ⁽²⁾	CPHRMOD0 ⁽²⁾
bit 7							bit 0

Legend:	HC = Hardware Clearable bit					
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 15 CRYON: Cryptographic Enable bit

1 = Module is enabled

0 = Module is disabled

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

bit 13 CRYSIDL: Cryptographic Stop in Idle Control bit (3)

1 = Stops module operation in Idle mode

0 = Continues module operation in Idle mode

bit 12 **ROLLIE:** CRYTXTB Rollover Interrupt Enable bit (1)

1 = Generates an interrupt event when the counter portion of CRYTXTB rolls over to '0'

0 = Does not generate an interrupt event when the counter portion of CRYTXTB rolls over to '0'

bit 11 **DONEIE:** Operation Done Interrupt Enable bit⁽¹⁾

1 = Generates an interrupt event when the current cryptographic operation completes

0 = Does not generate an interrupt event when the current cryptographic operation completes; software must poll the CRYGO or CRYBSY bit to determine when current cryptographic operation is complete

bit 10 FREEIE: Input Text Interrupt Enable bit (1)

1 = Generates an interrupt event when the input text (plaintext or ciphertext) is consumed during the current cryptographic operation

0 = Does not generate an interrupt event when the input text is consumed

bit 9 **Unimplemented:** Read as '0'

bit 8 **CRYGO:** Cryptographic Engine Start bit⁽¹⁾

1 = Starts the operation specified by OPMOD<3:0> (cleared automatically when operation is done)

0 = Stops the current operation (when cleared by software); also indicates the current operation has completed (when cleared by hardware)

Note 1: These bits are reset on system Resets or whenever the CRYMD bit is set.

2: Writes to these bit fields are locked out whenever an operation is in progress (CRYGO bit is set).

3: If the device enters Idle mode when CRYSIDL = 1, the module will stop its current operation. Entering into Idle mode while an OTP write operation is in process can result in irreversible corruption of the OTP.

REGISTER 23-1: CRYCONL: CRYPTOGRAPHIC CONTROL LOW REGISTER (CONTINUED)

```
OPMOD<3:0>: Cryptographic Operating Mode Selection bits(1,2)
bit 7-4
            1111 = Loads session key (decrypt session key in CRYTXTA/CRYTXTB using the Key Encryption Key
                    and write to CRYKEY)
            1110 = Encrypts session key (encrypt session key in CRYKEY using the Key Encryption Key and write
                    to CRYTXTA/CRYTXTB)
            1011 = Reserved
            1010 = Generates a random number
            1001 =
            1000 =
                  = Reserved
            0011 =
            0010 = AES decryption key expansion
            0001 = Decryption
            0000 = Encryption
bit 3
            CPHRSEL: Cipher Engine Select bit(1,2)
            1 = AES engine
            0 = DES engine
            CPHRMOD<2:0>: Cipher Mode bits<sup>(1,2)</sup>
bit 2-0
            11x = Reserved
            101 = Reserved
            100 = Counter (CTR) mode
            011 = Output Feedback (OFB) mode
            010 = Cipher Feedback (CFB) mode
            001 = Cipher Block Chaining (CBC) mode
            000 = Electronic Codebook (ECB) mode
```

- Note 1: These bits are reset on system Resets or whenever the CRYMD bit is set.
 - 2: Writes to these bit fields are locked out whenever an operation is in progress (CRYGO bit is set).
 - 3: If the device enters Idle mode when CRYSIDL = 1, the module will stop its current operation. Entering into Idle mode while an OTP write operation is in process can result in irreversible corruption of the OTP.

REGISTER 23-2: CRYCONH: CRYPTOGRAPHIC CONTROL HIGH REGISTER

U-0	R/W-0 ⁽¹⁾	R/W-0 ⁽¹⁾	R/W-0 ⁽¹⁾				
_	CTRSIZE6 ^(2,3)	CTRSIZE5 ^(2,3)	CTRSIZE4 ^(2,3)	CTRSIZE3 ^(2,3)	CTRSIZE2 ^(2,3)	CTRSIZE1(2,3)	CTRSIZE0(2,3)
bit 15							bit 8

R/W-0 ⁽¹⁾	R/W-0 ⁽¹⁾	R/W-0 ⁽¹⁾	U-0	R/W-0 ⁽¹⁾	R/W-0 ⁽¹⁾	R/W-0 ⁽¹⁾	R/W-0 ⁽¹⁾
SKEYSEL	KEYMOD1 ⁽²⁾	KEYMOD0 ⁽²⁾	_	KEYSRC3 ⁽²⁾	KEYSRC2 ⁽²⁾	KEYSRC1 ⁽²⁾	KEYSRC0 ⁽²⁾
bit 7							bit 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 15 Unimplemented: Read as '0'

bit 14-8 CTRSIZE<6:0>: Counter Size Select bits^(1,2,3)

Counter is defined as CRYTXTB<n:0>, where n = CTRSIZEx. Counter increments after each operation and generates a rollover event when the counter rolls over from $(2^{n-1} - 1)$ to 0.

1111111 = 128 bits (CRYTXTB<127:0>) 1111110 = 127 bits (CRYTXTB<126:0>)

•

0000010 = 3 bits (CRYTXTB<2:0>)

0000001 = 2 bits (CRYTXTB<1:0>)

0000000 = 1 bit (CRYTXTB<0>); rollover event occurs when CRYTXTB<0> toggles from '1' to '0'

bit 7 **SKEYSEL:** Session Key Select bit⁽¹⁾

1 = Key generation/encryption/loading are performed with CRYKEY<255:128>

0 = Key generation/encryption/loading are performed with CRYKEY<127:0>

bit 6-5 **KEYMOD<1:0>:** AES/DES Encrypt/Decrypt Key Mode/Key Length Select bits^(1,2)

For DES Encrypt/Decrypt Operations (CPHRSEL = 0):

11 = 64-bit, 3-key 3DES

10 = Reserved

01 = 64-bit, standard 2-key 3DES

00 = 64-bit DES

For AES Encrypt/Decrypt Operations (CPHRSEL = 1):

11 = Reserved

10 = 256-bit AES

01 = 192-bit AES

00 = 128-bit AES

bit 4 Unimplemented: Read as '0'

bit 3-0 **KEYSRC<3:0>:** Cipher Key Source bits^(1,2)

Refer to Table 23-1 and Table 23-2 for KEYSRC<3:0> values.

- Note 1: These bits are reset on system Resets or whenever the CRYMD bit is set.
 - 2: Writes to these bit fields are locked out whenever an operation is in progress (CRYGO bit is set).
 - 3: Used only in CTR operations when CRYTXTB is being used as a counter; otherwise, these bits have no effect.

REGISTER 23-3: CRYSTAT: CRYPTOGRAPHIC STATUS REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

R/HSC-x ⁽¹⁾	R/HSC-0 ⁽¹⁾	R/C-0, HS ⁽²⁾	R/C-0, HS ⁽²⁾	U-0	R/HSC-0 ⁽¹⁾	R/HSC-x ⁽¹⁾	R/HSC-x ⁽¹⁾
CRYBSY ⁽⁴⁾	TXTABSY	CRYABRT ⁽⁵⁾	ROLLOVR	_	MODFAIL ⁽³⁾	KEYFAIL ^(3,4)	PGMFAIL ^(3,4)
bit 7							bit 0

Legend:

R = Readable bit W = Writable bit U = Unimplemented bit, read as '0'
HS = Hardware Settable bit C = Clearable bit HSC = Hardware Settable/Clearable bit

-n = Value at POR '1' = Bit is set '0' = Bit is cleared x = Reset State Conditional bit

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

bit 7 CRYBSY: Cryptographic OTP Array Busy Status bit (1,4)

1 = The cryptography module is performing a cryptographic operation or OTP operation

0 = The module is not currently performing any operation

bit 6 TXTABSY: CRYTXTA Busy Status bit (1)

1 = The CRYTXTA register is busy and may not be written to

0 = The CRYTXTA is free and may be written to

bit 5 CRYABRT: Cryptographic Operation Aborted Status bit (2,5)

 $\ensuremath{\mathtt{1}}$ = Last operation was aborted by clearing the CRYGO bit in software

0 = Last operation completed normally (CRYGO cleared in hardware)

bit 4 ROLLOVR: Counter Rollover Status bit (2)

1 = The CRYTXTB counter rolled over on the last CTR mode operation; once set, this bit must be cleared by software before the CRYGO bit can be set again

0 = No rollover event has occurred

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

bit 2 **MODFAIL:** Mode Configuration Fail Flag bit^(1,3)

1 = Currently selected operating and Cipher mode configuration is invalid; the CRYWR bit cannot be set until a valid mode is selected (automatically cleared by hardware with any valid configuration)

0 = Currently selected operating and Cipher mode configuration are valid

bit 1 **KEYFAIL:** Key Configuration Fail Status bit^(1,3,4)

See Table 23-1 and Table 23-2 for invalid key configurations.

1 = Currently selected key and mode configurations are invalid; the CRYWR bit cannot be set until a valid mode is selected (automatically cleared by hardware with any valid configuration)

0 = Currently selected configurations are valid

bit 0 **PGMFAIL:** Key Storage/Configuration Program Fail Flag bit (1,3,4)

1 = The page indicated by KEYPG<3:0> is reserved or locked; the CRYWR bit cannot be set and no programming operation can be started

0 = The page indicated by KEYPG<3:0> is available for programming

Note 1: These bits are reset on system Resets or whenever the CRYMD bit is set.

2: These bits are reset on system Resets when the CRYMD bit is set or when CRYGO is cleared.

3: These bits are functional even when the module is disabled (CRYON = 0); this allows mode configurations to be validated for compatibility before enabling the module.

4: These bits are automatically set during all OTP read operations, including the initial read at POR. Once the read is completed, the bit assumes the proper state that reflects the current configuration.

5: If this bit is set, a cryptographic operation cannot be performed.

REGISTER 23-4: CRYOTP: CRYPTOGRAPHIC OTP PAGE PROGRAM CONTROL REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_		_	_	_
bit 15							

R/HSC-x ⁽¹⁾	R/W-0 ⁽¹⁾	R/S/HC-1	R/W-0 ⁽¹⁾	R/W-0 ⁽¹⁾	R/W-0 ⁽¹⁾	R/W-0 ⁽¹⁾	R/S/HC-0 ⁽²⁾
PGMTST	OTPIE	CRYREAD ^(3,4)	KEYPG3	KEYPG2	KEYPG1	KEYPG0	CRYWR ^(3,4)
bit 7							bit 0

Legend:

R = Readable bit W = Writable bit U = Unimplemented bit, read as '0' S = Settable bit HSC = Hardware Settable/Clearable bit HC = Hardware Clearable bit -n = Value at POR '1' = Bit is set '0' = Bit is cleared x = Bit is unknown

bit 15-8 Unimplemented: Read as '0'

PGMTST: Key Storage/Configuration Program Test bit(1) bit 7

> This bit mirrors the state of the TSTPGM bit and is used to test the programming of the secure OTP array after programming.

- 1 = TSTPGM (CFGPAGE<30>) is programmed ('1')
- 0 = TSTPGM is not programmed ('0')
- bit 6 **OTPIE:** Key Storage/Configuration Program Interrupt Enable bit⁽¹⁾
 - 1 = Generates an interrupt when the current programming or read operation completes
 - 0 = Does not generate an interrupt when the current programming or read operation completes; software must poll the CRYWR, CRYREAD or CRYBSY bit to determine when the current programming operation is complete
- CRYREAD: Cryptographic Key Storage/Configuration Read bit (3,4) bit 5
 - 1 = This bit is set to start a read operation; read operation is in progress while this bit is set and CRYGO = 1
 - 0 = Read operation has completed
- bit 4-1 **KEYPG<3:0>:** Key Storage/Configuration Program Page Select bits⁽¹⁾

1111 =

= Reserved

1001 =

1000 **= OTP Page 8**

0111 = OTP Page 7

0110 = OTP Page 6

0101 = OTP Page 5

0100 = OTP Page 4

0011 = OTP Page 3

0010 = OTP Page 2

0001 = OTP Page 1

0000 = Configuration Page (CFGPAGE, OTP Page 0)

CRYWR: Cryptographic Key Storage/Configuration Program bit^(2,3,4) bit 0

- 1 = Programs the Key Storage/Configuration bits with the value found in CRYTXTC<63:0>
- 0 = Program operation has completed
- Note 1: These bits are reset on system Resets or whenever the CRYMD bit is set.
 - 2: These bits are reset on system Resets, when the CRYMD bit is set, or when CRYGO is cleared.
 - 3: Set this bit only when CRYON = 1 and CRYGO = 0. Do not set CRYREAD or CRYWR both at any given time.
 - Do not clear CRYON or these bits while they are set; always allow the hardware operation to complete and clear the bit automatically.

REGISTER 23-5: CFGPAGE: SECURE ARRAY CONFIGURATION BITS (OTP PAGE 0) REGISTER

r-x	R/PO-x	U-x	U-x	R/PO-x	R/PO-x	R/PO-x	R/PO-x
_	TSTPGM ⁽¹⁾	_	_	KEY4TYPE1	KEY4TYPE0	KEY3TYPE1	KEY3TYPE0
bit 31							bit 24

R/PO-x	R/PO-x	R/PO-x	R/PO-x	R/PO-x	R/PO-x	R/PO-x	R/PO-x
KEY2TYPE1	KEY2TYPE0	KEY1TYPE1	KEY1TYPE0	SKEYEN	LKYSRC7	LKYSRC6	LKYSRC5
bit 23							bit 16

R/PO-x	R/PO-x	R/PO-x	R/PO-x	R/PO-x	R/PO-x	R/PO-x	R/PO-x
LKYSRC4	LKYSRC3	LKYSRC2	LKYSRC1	LKYSRC0	SRCLCK	WRLOCK8	WRLOCK7
bit 15							bit 8

R/PO-x	R/PO-x	R/PO-x	R/PO-x	R/PO-x	R/PO-x	R/PO-x	R/PO-x
WRLOCK6	WRLOCK5	WRLOCK74	WRLOCK3	WRLOCK2	WRLOCK1	WRLOCK0	SWKYDIS
bit 7							bit 0

Legend: r = Reserved bit

R = Readable bit PO = Program Once 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 Reserved: Do not modify

bit 30 **TSTPGM:** Customer Program Test bit⁽¹⁾

1 = CFGPAGE has been programmed

0 = CFGPAGE has not been programmed

bit 29-28 **Unimplemented:** Read as '0'

bit 27-26 **KEY4TYPE<1:0>:** Key Type for OTP Pages 7 and 8 bits

00 = Keys in these pages are for DES/DES2 operations only

01 = Keys in these pages are for DES3 operations only

10 = Keys in these pages are for 128-bit AES operations only

11 = Keys in these pages are for 192-bit/256-bit AES operations only

bit 25-24 **KEY3TYPE<1:0>:** Key Type for OTP Pages 5 and 6 bits

00 = Keys in these pages are for DES/DES2 operations only

01 = Keys in these pages are for DES3 operations only

10 = Keys in these pages are for 128-bit AES operations only

11 = Keys in these pages are for 192-bit/256-bit AES operations only

bit 23-22 **KEY2TYPE<1:0>:** Key Type for OTP Pages 3 and 4 bits

00 = Keys in these pages are for DES/DES2 operations only

01 = Keys in these pages are for DES3 operations only

10 = Keys in these pages are for 128-bit AES operations only

11 = Keys in these pages are for 192-bit/256-bit AES operations only

bit 21-20 **KEY1TYPE<1:0>:** Key Type for OTP Pages 1 and 2 bits

00 = Keys in these pages are for DES/DES2 operations only

01 = Keys in these pages are for DES3 operations only

10 = Keys in these pages are for 128-bit AES operations only

11 = Keys in these pages are for 192-bit/256-bit AES operations only

Note 1: This bit's state is mirrored by the PGMTST bit (CRYOTP<7>).

REGISTER 23-5: CFGPAGE: SECURE ARRAY CONFIGURATION BITS (OTP PAGE 0) REGISTER (CONTINUED)

- bit 19 SKEYEN: Session Key Enable bit
 - 1 = Stored Key #1 may be used only as a Key Encryption Key
 - 0 = Stored Key #1 may be used for any operation
- bit 18-11 **LKYSRC<7:0>:** Locked Key Source Configuration bits

If SRCLCK = 1:

1xxxxxxx = Key Source is as if KEYSRC<3:0> = 1111
01xxxxxx = Key Source is as if KEYSRC<3:0> = 0111
001xxxxx = Key Source is as if KEYSRC<3:0> = 0110
0001xxxx = Key Source is as if KEYSRC<3:0> = 0101
00001xxx = Key Source is as if KEYSRC<3:0> = 0100
000001xx = Key Source is as if KEYSRC<3:0> = 0010
000001x = Key Source is as if KEYSRC<3:0> = 0010
0000001 = Key Source is as if KEYSRC<3:0> = 0001
00000000 = Key Source is as if KEYSRC<3:0> = 0000

If SRCLCK = 0:

These bits are ignored.

- bit 10 SRCLCK: Key Source Lock bit
 - 1 = The key source is determined by the KEYSRC<3:0> (CRYCONH<3:0>) bits (software key selection is disabled)
 - 0 = The key source is determined by the KEYSRC<3:0> (CRYCONH<3:0>) bits (locked key selection is disabled)
- bit 9-1 WRLOCK<8:0>: Write Lock Page Enable bits

For OTP Pages 0 (CFGPAGE) through 8:

- 1 = OTP page is permanently locked and may not be programmed
- 0 = OTP page is unlocked and may be programmed
- bit 0 **SWKYDIS:** Software Key Disable bit
 - 1 = Software key (CRYKEY register) is disabled; when KEYSRC<3:0> = 0000, the KEYFAIL status bit will be set and no encryption/decryption/session key operations can be started until the KEYSRC<3:0> bits are changed to a value other than '0000'
 - 0 = Software key (CRYKEY register) can be used as a key source when KEYSRC<3:0> = 0000
- Note 1: This bit's state is mirrored by the PGMTST bit (CRYOTP<7>).

TABLE 23-1: DES/3DES KEY SOURCE SELECTION

Mode of	VEVMOD 44.05	VEVCDC 42.05	Session K	ey Source	OTP Array
Operation	KEYMOD<1:0>	KEYSRC<3:0>	SKEYEN = 0	SKEYEN = 1	Address
		0000(1)	CRYKE	Y<63:0>	_
		0001	DES Key #1	Key Config Error ⁽²⁾	<63:0>
		0010	DES Key #2		<127:64>
		0011	DES F	Key #3	<191:128>
64-Bit DES		0100	DES F	Key #4	<255:192>
04-BILDES	00	0101	DES F	Key #5	<319:256>
		0110	DES F	Key #6	<383:320>
		0111	DES F	Key #7	<447:384>
		1111		ved ⁽²⁾	_
		All Others	Key Conf	ig Error ⁽²⁾	_
		0000(1)	CRYKEY<63:0> (1st/3rd) CRYKEY<127:64> (2nd)		_
		0001	DES Key #1 (1st/3rd) DES Key #2 (2nd)	Key Config Error ⁽²⁾	<63:0> <127:64>
64-Bit 2-Key 3DES	01	0010	DES Key #3 (1st/3rd) DES Key #4 (2nd)		<191:128> <255:192>
(Standard 2-Key E-D-E/D-E-D)		0011	DES Key #5 (1st/3rd) DES Key #6 (2nd)		<319:256> <383:320>
		0100	DES Key # DES Key	‡7 (1st/3rd) ± #8 (2nd)	<447:384> <511:448>
		1111	Reser	ved ⁽²⁾	_
		All Others	Key Conf	ig Error ⁽²⁾	_
(Reserved)	10	xxxx	Key Conf	ig Error ⁽²⁾	_
		0000(1)	CRYKEY<127:64	> (1st Iteration) 4> (2nd Iteration) 28> (3rd Iteration)	1
64-Bit 3-Key	11	0001	DES Key #1 (1st) DES Key #2 (2nd) DES Key #3 (3rd)	Key Config Error ⁽²⁾	<63:0> <127:64> <191:128>
3DES		0010	DES Key	/ #4 (1st) / #5 (2nd) / #6 (3rd)	<255:192> <319:256> <383:320>
		1111	Reserved ⁽²⁾		_
		All Others	Key Conf	ig Error ⁽²⁾	

Note 1: This configuration is considered a Key Configuration Error (KEYFAIL bit is set) if SWKYDIS is also set.

^{2:} The KEYFAIL bit (CRYSTAT<1>) is set when these configurations are selected and remains set until a valid configuration is selected.

TABLE 23-2: AES KEY MODE/SOURCE SELECTION

Mode of	KEYMOD<1:0>	KEYSRC<3:0>	Key S	Source	OTP Address
Operation	KEYMOD<1:0>	KE13KU<3:0>	SKEYEN = 0	SKEYEN = 1	OTP Address
		0000(1)	CRYKE	Y<127:0>	_
		0001	AES Key #1	Key Config Error ⁽²⁾	<127:0>
		0010	AES I	Key #2	<255:128>
128-Bit AES	00	0011	AES I	Key #3	<383:256>
		0100	AES I	Key #4	<511:384>
		1111	Reserved ⁽²⁾		_
		All Others	Key Config Error ⁽²⁾		_
		0000(1)	CRYKEY<191:0>		_
		0001	AES Key #1	Key Config Error ⁽²⁾	<191:0>
192-Bit AES	01	0010	AES Key #2		<383:192>
		1111	Rese	rved ⁽²⁾	_
		All Others	Key Cont	fig Error ⁽²⁾	_
		0000(1)	CRYKE'	Y<255:0>	_
		0001	AES Key #1	Key Config Error ⁽²⁾	<255:0>
256-Bit AES	10	0010	AES I	Key #2	<511:256>
		1111	Reserved ⁽²⁾		_
		All Others	Key Cont	fig Error ⁽²⁾	
(Reserved)	11	XXXX	Key Cont	fig Error ⁽²⁾	

Note 1: This configuration is considered a Key Configuration Error (KEYFAIL bit is set) if SWKYDIS is also set.

^{2:} The KEYFAIL bit (CRYSTAT<1>) is set when these configurations are selected and remains set until a valid configuration is selected.

NOTES:		

24.0 32-BIT PROGRAMMABLE CYCLIC REDUNDANCY CHECK (CRC) GENERATOR

Note:

This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the "dsPIC33/PIC24 Family Reference Manual", "32-Bit Programmable Cyclic Redundancy Check (CRC)" (DS30009729). The information in this data sheet supersedes the information in the FRM.

The 32-bit programmable CRC generator provides a hardware implemented method of quickly generating checksums for various networking and security applications. It offers the following features:

- User-programmable CRC polynomial equation, up to 32 bits
- Programmable shift direction (little or big-endian)
- · Independent data and polynomial lengths
- · Configurable interrupt output
- Data FIFO

Figure 24-1 displays a simplified block diagram of the CRC generator. A simple version of the CRC shift engine is displayed in Figure 24-2.

FIGURE 24-1: CRC MODULE BLOCK DIAGRAM

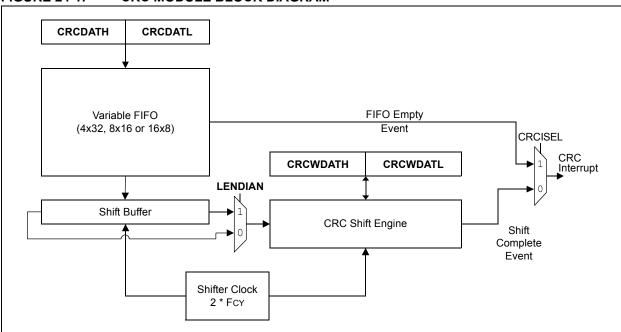
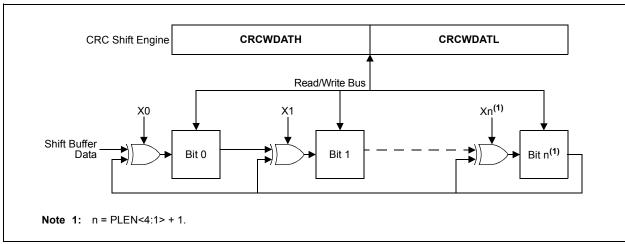


FIGURE 24-2: CRC SHIFT ENGINE DETAIL



24.1 User Interface

24.1.1 POLYNOMIAL INTERFACE

The CRC module can be programmed for CRC polynomials of up to the 32nd order, using up to 32 bits.

Polynomial length, which reflects the highest exponent in the equation, is selected by the PLEN<4:0> bits (CRCCON2<4:0>).

The CRCXORL and CRCXORH registers control which exponent terms are included in the equation. Setting a particular bit includes that exponent term in the equation. Functionally, this includes an XOR operation on the corresponding bit in the CRC engine. Clearing the bit disables the XOR.

For example, consider two CRC polynomials, one is a 16-bit and the other is a 32-bit equation.

EQUATION 24-1: 16-BIT, 32-BIT CRC POLYNOMIALS

X16 + X12 + X5 + 1

and

To program these polynomials into the CRC generator, set the register bits, as shown in Table 24-1.

Note that the appropriate positions are set to '1' to indicate that they are used in the equation (for example, X26 and X23). The '0' bit required by the equation is always XORed; thus, X0 is a don't care. For a polynomial of length 32, it is assumed that the 32nd bit will be used. Therefore, the X<31:1> bits do not have the 32nd bit.

24.1.2 DATA INTERFACE

The module incorporates a FIFO that works with a variable data width. Input data width can be configured to any value between 1 and 32 bits using the DWIDTH<4:0> bits (CRCCON2<12:8>). When the data width is greater than 15, the FIFO is 4 words deep. When the DWIDTHx bits are between 15 and 8, the FIFO is 8 words deep. When the DWIDTHx bits are less than 8, the FIFO is 16 words deep.

The data for which the CRC is to be calculated must first be written into the FIFO. Even if the data width is less than 8, the smallest data element that can be written into the FIFO is 1 byte. For example, if the DWIDTHx bits are 5, then the size of the data is DWIDTH<4:0> + 1 or 6. The data is written as a whole byte; the two unused upper bits are ignored by the module.

Once data is written into the MSb of the CRCDAT registers (that is, the MSb as defined by the data width), the value of the VWORD<4:0> bits (CRCCON1<12:8>) increments by one. For example, if the DWIDTHx bits are 24, the VWORDx bits will increment when bit 7 of CRCDATH is written. Therefore, CRCDATL must always be written to before CRCDATH.

The CRC engine starts shifting data when the CRCGO bit is set and the value of the VWORDx bits is greater than zero.

Each word is copied out of the FIFO into a buffer register, which decrements the VWORDx bits. The data is then shifted out of the buffer. The CRC engine continues shifting at a rate of two bits per instruction cycle, until the VWORDx bits reach zero. This means that for a given data width, it takes half that number of instructions for each word to complete the calculation. For example, it takes 16 cycles to calculate the CRC for a single word of 32-bit data.

When the VWORDx bits reach the maximum value for the configured value of the DWIDTHx bits (4, 8 or 16), the CRCFUL bit becomes set. When the VWORDx bits reach zero, the CRCMPT bit becomes set. The FIFO is emptied and the VWORD<4:0> bits are set to '00000' whenever CRCEN is '0'.

At least one instruction cycle must pass after a write to CRCWDAT before a read of the VWORDx bits is done.

TABLE 24-1: CRC SETUP EXAMPLES FOR 16 AND 32-BIT POLYNOMIALS

CBC Control Bito	Bit Values					
CRC Control Bits	16-Bit Polynomial	32-Bit Polynomial				
PLEN<4:0>	01111	11111				
X<31:16>	0000 0000 0000 0001	0000 0100 1100 0001				
X<15:0>	0001 0000 0010 00x	0001 1101 1011 011x				

24.1.3 DATA SHIFT DIRECTION

The LENDIAN bit (CRCCON1<3>) is used to control the shift direction. By default, the CRC will shift data through the engine, MSb first. Setting LENDIAN (= 1) causes the CRC to shift data, LSb first. This setting allows better integration with various communication schemes and removes the overhead of reversing the bit order in software. Note that this only changes the direction the data is shifted into the engine. The result of the CRC calculation will still be a normal CRC result, not a reverse CRC result.

24.1.4 INTERRUPT OPERATION

The module generates an interrupt that is configurable by the user for either of two conditions.

If CRCISEL is '0', an interrupt is generated when the VWORD<4:0> bits make a transition from a value of '1' to '0'. If CRCISEL is '1', an interrupt will be generated after the CRC operation finishes and the module sets the CRCGO bit to '0'. Manually setting CRCGO to '0' will not generate an interrupt. Note that when an interrupt occurs, the CRC calculation would not yet be complete. The module will still need (PLEN + 1)/2 clock cycles, after the interrupt is generated, until the CRC calculation is finished.

24.1.5 TYPICAL OPERATION

To use the module for a typical CRC calculation:

- 1. Set the CRCEN bit to enable the module.
- Configure the module for desired operation:
 - a) Program the desired polynomial using the CRCXORL and CRCXORH registers, and the PLEN<4:0> bits.
 - b) Configure the data width and shift direction using the DWIDTHx and LENDIAN bits.
 - c) Select the desired Interrupt mode using the CRCISEL bit.
- Preload the FIFO by writing to the CRCDATL and CRCDATH registers until the CRCFUL bit is set or no data is left.

- Clear old results by writing 00h to CRCWDATL and CRCWDATH. The CRCWDAT registers can also be left unchanged to resume a previously halted calculation.
- 5. Set the CRCGO bit to start calculation.
- Write remaining data into the FIFO as space becomes available.
- When the calculation completes, CRCGO is automatically cleared. An interrupt will be generated if CRCISEL = 1.
- 8. Read CRCWDATL and CRCWDATH for the result of the calculation.

There are eight registers used to control programmable CRC operation:

- CRCCON1
- CRCCON2
- CRCXORL
- CRCXORH
- CRCDATL
- CRCDATH
- CRCWDATL
- CRCWDATH

The CRCCON1 and CRCCON2 registers (Register 24-1 and Register 24-2) control the operation of the module and configure the various settings.

The CRCXORL/H registers (Register 24-3 and Register 24-4) select the polynomial terms to be used in the CRC equation. The CRCDAT and CRCWDAT registers are each register pairs that serve as buffers for the double-word input data and CRC processed output, respectively.

REGISTER 24-1: CRCCON1: CRC CONTROL 1 REGISTER

R/W-0	U-0	R/W-0	R-0, HSC				
CRCEN	_	CSIDL	VWORD4	VWORD3	VWORD2	VWORD1	VWORD0
bit 15							bit 8

R-0, HSC	R-1, HSC	R/W-0	R/W-0, HC	R/W-0	U-0	U-0	U-0
CRCFUL	CRCMPT	CRCISEL	CRCGO	LENDIAN	_	_	_
bit 7							bit 0

Legend:HC = Hardware Clearable bitHSC = Hardware Settable/Clearable bitR = Readable bitW = Writable bitU = Unimplemented bit, read as '0'-n = Value at POR'1' = Bit is set'0' = Bit is clearedx = Bit is unknown

bit 15 CRCEN: CRC Enable bit

bit 14

1 = Enables module

0 = Disables module; all state machines, pointers and the CRCWDAT/CRCDATH registers reset; other SFRs are NOT reset

Unimplemented: Read as '0'

bit 13 CSIDL: CRC Stop in Idle Mode bit

1 = Discontinues module operation when device enters Idle mode

0 = Continues module operation in Idle mode

bit 12-8 VWORD<4:0>: Pointer Value bits

Indicates the number of valid words in the FIFO. Has a maximum value of 8 when PLEN<4:0> \geq 7 or 16

when PLEN<4:0> \leq 7.

bit 7 CRCFUL: CRC FIFO Full bit

1 = FIFO is full 0 = FIFO is not full

bit 6 CRCMPT: CRC FIFO Empty bit

1 = FIFO is empty
0 = FIFO is not empty

bit 5 CRCISEL: CRC Interrupt Selection bit

1 = Interrupt on FIFO is empty; the final word of data is still shifting through the CRC

0 = Interrupt on shift is complete and results are ready

bit 4 CRCGO: Start CRC bit

1 = Starts CRC serial shifter

0 = CRC serial shifter is turned off

bit 3 LENDIAN: Data Shift Direction Select bit

1 = Data word is shifted into the FIFO, starting with the LSb (little-endian)

0 = Data word is shifted into the FIFO, starting with the MSb (big-endian)

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

REGISTER 24-2: CRCCON2: CRC CONTROL 2 REGISTER

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	_	DWIDTH4	DWIDTH3	DWIDTH2	DWIDTH1	DWIDTH0
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	_	PLEN4	PLEN3	PLEN2	PLEN1	PLEN0
bit 7							bit 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 15-13 **Unimplemented:** Read as '0'

bit 12-8 **DWIDTH<4:0>:** Data Word Width Configuration bits

Configures the width of the data word (Data Word Width -1).

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

bit 4-0 **PLEN<4:0>:** Polynomial Length Configuration bits

Configures the length of the polynomial (Polynomial Length – 1).

REGISTER 24-3: CRCXORL: CRC XOR POLYNOMIAL REGISTER, LOW BYTE

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	
X<15:8>								
bit 15							bit 8	

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0
			X<7:1>				_
bit 7							bit 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 15-1 X<15:1>: XOR of Polynomial Term xⁿ Enable bits

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

REGISTER 24-4: CRCXORH: CRC XOR POLYNOMIAL REGISTER, HIGH BYTE

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	
X<31:24>								
bit 15							bit 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		
X<23:16>									
bit 7							bit 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 15-0 X<31:16>: XOR of Polynomial Term xⁿ Enable bits

25.0 12-BIT A/D CONVERTER WITH THRESHOLD DETECT

Note:

This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information on the 12-Bit A/D Converter, refer to the "dsPIC33/PIC24 Family Reference Manual", "12-Bit A/D Converter with Threshold Detect" (DS39739).

The 12-bit A/D Converter has the following key features:

- Successive Approximation Register (SAR) Conversion
- · Conversion Speeds of up to 200 ksps
- Up to 19 Analog Input Channels (internal and external)
- Selectable 10-Bit or 12-Bit (default) Conversion Resolution
- · Multiple Internal Reference Input Channels
- · External Voltage Reference Input Pins
- Unipolar Differential Sample-and-Hold (S/H) Amplifier
- Automated Threshold Scan and Compare Operation to Pre-Evaluate Conversion Results
- · Selectable Conversion Trigger Source
- Fixed Length (one word per channel), Configurable Conversion Result Buffer
- Four Options for Results Alignment
- · Configurable Interrupt Generation
- Enhanced DMA Operations with Indirect Address Generation
- · Operation During CPU Sleep and Idle modes

The 12-bit A/D Converter module is an enhanced version of the 10-bit module offered in earlier PIC24 devices. It is a Successive Approximation Register (SAR) Converter, enhanced with 12-bit resolution, a wide range of automatic sampling options, tighter integration with other analog modules and a configurable results buffer.

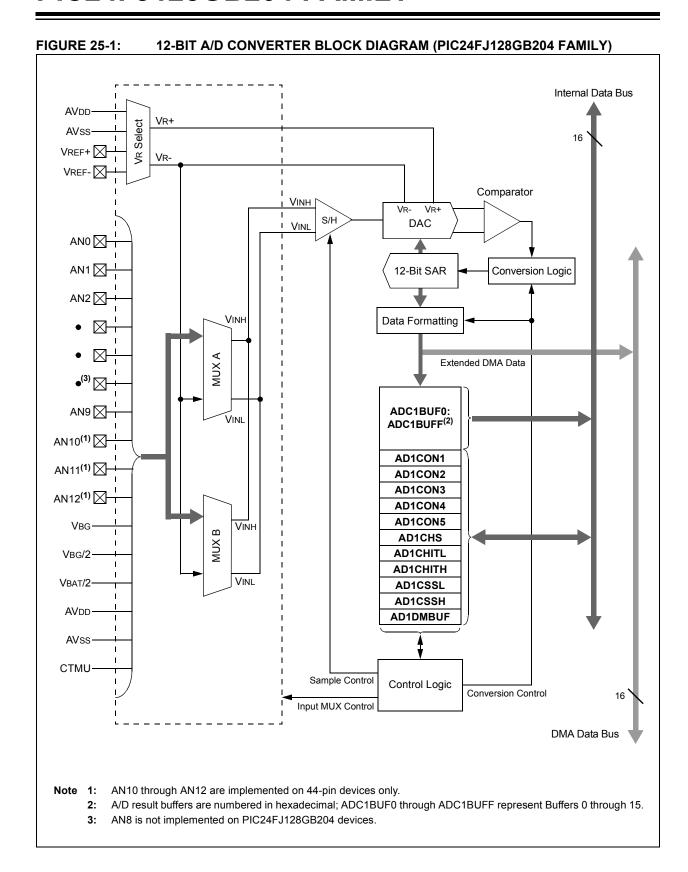
It also includes a unique Threshold Detect feature that allows the module itself to make simple decisions based on the conversion results, and enhanced operation with the DMA Controller through Peripheral Indirect Addressing (PIA).

A simplified block diagram for the module is shown in Figure 25-1.

25.1 Basic Operation

To perform a standard A/D conversion:

- 1. Configure the module:
 - a) Configure port pins as analog inputs by setting the appropriate bits in the ANSx registers (see Section 11.2 "Configuring Analog Port Pins (ANSx)" for more information).
 - Select the voltage reference source to match the expected range on analog inputs (AD1CON2<15:13>).
 - Select the positive and negative multiplexer inputs for each channel (AD1CHS<15:0>).
 - Select the analog conversion clock to match the desired data rate with the processor clock (AD1CON3<7:0>).
 - e) Select the appropriate sample/conversion sequence (the AD1CON1<7:4> and AD1CON3<12:8> bits).
 - f) For Channel A scanning operations, select the positive channels to be included (AD1CSSH and AD1CSSL registers).
 - g) Select how the conversion results are presented in the buffer (AD1CON1<9:8> and the AD1CON5 register).
 - h) Select the interrupt rate (AD1CON2<6:2>).
 - i) Turn on A/D module (AD1CON1<15>).
- 2. Configure the A/D interrupt (if required):
 - a) Clear the AD1IF bit (IFS0<13>).
 - b) Enable the AD1IE interrupt (IEC0<13>).
 - c) Select the A/D interrupt priority (IPC3<6:4>).
- If the module is configured for manual sampling, set the SAMP bit (AD1CON1<1>) to begin sampling.



25.2 Extended DMA Operations

In addition to the standard features available on all 12-bit A/D Converters, PIC24FJ128GB204 family devices implement a limited extension of DMA functionality. This extension adds features that work with the device's DMA Controller to expand the A/D module's data storage abilities beyond the module's built-in buffer.

The Extended DMA functionality is controlled by the DMAEN bit (AD1CON1<11>); setting this bit enables the functionality. The DMABM bit (AD1CON1<12>) configures how the DMA feature operates.

25.2.1 EXTENDED BUFFER MODE

Extended Buffer mode (DMABM = 1) is useful for storing the results of channels. It can also be used to store the conversion results on any A/D channel in any implemented address in data RAM.

In Extended Buffer mode, all data from the A/D Buffer register, and channels above 26, are mapped into data RAM. Conversion data is written to a destination specified by the DMA Controller, specifically by the DMADSTn register. This allows users to read the conversion results of channels above 26, which do not have their own memory-mapped A/D buffer locations, from data memory.

When using Extended Buffer mode, always set the BUFREGEN bit to disable FIFO operation. In addition, disable the Split Buffer mode by clearing the BUFM bit.

25.2.2 PIA MODE

When DMABM = 0, the A/D module is configured to function with the DMA Controller for Peripheral Indirect Addressing (PIA) mode operations. In this mode, the A/D module generates an 11-bit Indirect Address (IA). This is ORed with the destination address in the DMA Controller to define where the A/D conversion data will be stored.

In PIA mode, the buffer space is created as a series of contiguous smaller buffers, one per analog channel. The size of the channel buffer determines how many analog channels can be accommodated. The size of the buffer is selected by the DMABLx bits (AD1CON4<2:0>). The size options range from a single word per buffer to 128 words. Each channel is allocated a buffer of this size, regardless of whether or not the channel will actually have conversion data.

The IA is created by combining the base address within a channel buffer with three to five bits (depending on the buffer size) to identify the channel. The base address ranges from zero to seven bits wide, depending on the buffer size. The address is right-padded with a '0' in order to maintain address alignment in the Data Space. The concatenated channel and base address bits are then left-padded with zeros, as necessary, to complete the 11-bit IA.

The IA is configured to auto-increment during write operations by using the SMPIx bits (AD1CON2<6:2>).

As with PIA operations for any DMA-enabled module, the base destination address in the DMADSTn register must be masked properly to accommodate the IA. Table 25-1 shows how complete addresses are formed. Note that the address masking varies for each buffer size option. Because of masking requirements, some address ranges may not be available for certain buffer sizes. Users should verify that the DMA base address is compatible with the buffer size selected.

Figure 25-2 shows how the parts of the address define the buffer locations in data memory. In this case, the module "allocates" 256 bytes of data RAM (1000h to 1100h) for 32 buffers of four words each. However, this is not a hard allocation and nothing prevents these locations from being used for other purposes. For example, in the current case, if Analog Channels 1, 3 and 8 are being sampled and converted, conversion data will only be written to the channel buffers, starting at 1008h, 1018h and 1040h. The holes in the PIA buffer space can be used for any other purpose. It is the user's responsibility to keep track of buffer locations and preventing data overwrites.

25.3 A/D Operation with VBAT

One of the A/D channels is connected to the VBAT pin to monitor the VBAT voltage. This allows monitoring the VBAT pin voltage (battery voltage) with no external connection. The voltage measured, using the A/D VBAT monitor, is VBAT/2. The voltage can be calculated by reading A/D = ((VBAT/2)/VDD) * 1024 for 10-bit A/D and ((VBAT/2)/VDD) * 4096 for 12 bit A/D.

When using the VBAT A/D monitor:

- Connect the A/D channel to ground to discharge the sample capacitor.
- Because of the high-impedance of VBAT, select higher sampling time to get an accurate reading.

Since the VBAT pin is connected to the A/D during sampling, to prolong the VBAT battery life, the recommendation is to only select the VBAT channel when needed.

25.4 Registers

The 12-bit A/D Converter is controlled through a total of 11 registers:

- AD1CON1 through AD1CON5 (Register 25-1 through Register 25-5)
- AD1CHS (Register 25-6)

- AD1CHITL (Register 25-8)
- AD1CSSH and AD1CSSL (Register 25-9 and Register 25-10)
- AD1CTMENL (Register 25-11)
- AD1DMBUF (not shown) The 16-bit conversion buffer for Extended Buffer mode

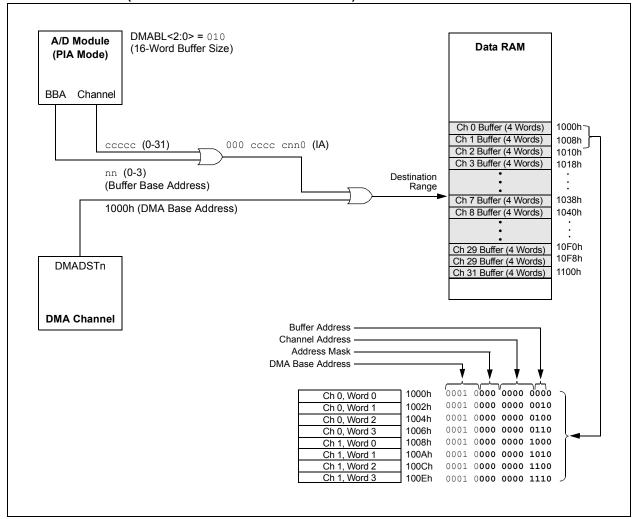
TABLE 25-1: INDIRECT ADDRESS GENERATION IN PIA MODE

DMABL<2:0>	Buffer Size per Channel (words)	Generated Offset Address (lower 11 bits)	Available Input Channels	Allowable DMADSTn Addresses
000	1	000 00cc ccc0	32	xxxx xxxx xx00 0000
001	2	000 0ccc ccn0	32	xxxx xxxx x000 0000
010	4	000 cccc cnn0	32	xxxx xxxx 0000 0000
011	8	00c cccc nnn0	32	xxxx xxx0 0000 0000
100	16	Occ cccn nnnO	32	xxxx xx00 0000 0000
101	32	ccc ccnn nnn0	32	xxxx x000 0000 0000
110	64	ccc cnnn nnn0	16	xxxx x000 0000 0000
111	128	ccc nnnn nnn0	8	xxxx x000 0000 0000

Legend: ccc = Channel number (three to five bits), n = Base buffer address (zero to seven bits),

x = User-definable range of DMADSTn for base address, 0 = Masked bits of DMADSTn for IA.

FIGURE 25-2: EXAMPLE OF BUFFER ADDRESS GENERATION IN PIA MODE (4-WORD BUFFERS PER CHANNEL)



REGISTER 25-1: AD1CON1: ADC1 CONTROL REGISTER 1

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADON	_	ADSIDL	DMABM ⁽¹⁾	DMAEN	MODE12	FORM1	FORM0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0, HSC	R/C-0, HSC
SSRC3	SSRC2	SSRC1	SSRC0	_	ASAM	SAMP	DONE
bit 7							bit 0

Legend:	C = Clearable bit	U = Unimplemented bit,	U = Unimplemented bit, read as '0'		
R = Readable bit	W = Writable bit	HSC = Hardware Settab	HSC = Hardware Settable/Clearable bit		
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown		

bit 15 ADON: ADC1 Operating Mode bit

1 = A/D Converter module is operating

0 = A/D Converter is off

bit 14 Unimplemented: Read as '0'

bit 13 ADSIDL: ADC1 Stop in Idle Mode bit

1 = Discontinues module operation when device enters Idle mode

0 = Continues module operation in Idle mode

bit 12 **DMABM:** Extended DMA Buffer Mode Select bit⁽¹⁾

1 = Extended Buffer mode: Buffer address is defined by the DMADSTn register

0 = PIA mode: Buffer addresses are defined by the DMA Controller and AD1CON4<2:0>

bit 11 DMAEN: Extended DMA/Buffer Enable bit

1 = Extended DMA and buffer features are enabled

0 = Extended features are disabled

bit 10 MODE12: ADC1 12-Bit Operation Mode bit

1 = 12-bit A/D operation

0 = 10-bit A/D operation

bit 9-8 **FORM<1:0>:** Data Output Format bits (see the following formats)

11 = Fractional result, signed, left justified

10 = Absolute fractional result, unsigned, left justified

01 = Decimal result, signed, right justified

00 = Absolute decimal result, unsigned, right justified

bit 7-4 SSRC<3:0>: Sample Clock Source Select bits

1xxx = Unimplemented, do not use

0111 = Internal counter ends sampling and starts conversion (auto-convert); do not use in Auto-Scan mode

0110 = Unimplemented

0101 **= TMR1**

0100 = CTMU

0011 **= TMR5**

0011 = TMR3

0010 - INTO

0001 **= INTO**

0000 = The SAMP bit must be cleared by software to start conversion

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

bit 2 ASAM: ADC1 Sample Auto-Start bit

1 = Sampling begins immediately after last conversion; SAMP bit is auto-set

0 = Sampling begins when SAMP bit is manually set

Note 1: This bit is only available when Extended DMA/Buffer features are available (DMAEN = 1).

REGISTER 25-1: AD1CON1: ADC1 CONTROL REGISTER 1 (CONTINUED)

bit 1 SAMP: ADC1 Sample Enable bit

1 = A/D Sample-and-Hold amplifiers are sampling 0 = A/D Sample-and-Hold amplifiers are holding

bit 0 **DONE:** ADC1 Conversion Status bit

1 = A/D conversion cycle has completed

0 = A/D conversion cycle has not started or is in progress

Note 1: This bit is only available when Extended DMA/Buffer features are available (DMAEN = 1).

REGISTER 25-2: AD1CON2: ADC1 CONTROL REGISTER 2

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0
PVCFG1	PVCFG0	NVCFG0	OFFCAL	BUFREGEN	CSCNA	_	_
bit 15							bit 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
BUFS ⁽¹⁾	SMPI4	SMPI3	SMPI2	SMPI1	SMPI0	BUFM ⁽¹⁾	ALTS
bit 7							bit 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 15-14 **PVCFG<1:0>:** Converter Positive Voltage Reference Configuration bits

1x = Unimplemented, do not use

01 = External VREF+

00 = AVDD

bit 13 **NVCFG0:** Converter Negative Voltage Reference Configuration bit

1 = External VREF-

0 = AVss

bit 12 OFFCAL: Offset Calibration Mode Select bit

1 = Inverting and non-inverting inputs of channel Sample-and-Hold are connected to AVss

0 = Inverting and non-inverting inputs of channel Sample-and-Hold are connected to normal inputs

bit 11 BUFREGEN: ADC1 Buffer Register Enable bit

1 = Conversion result is loaded into the buffer location determined by the converted channel

0 = A/D result buffer is treated as a FIFO

bit 10 CSCNA: Scan Input Selections for CH0+ During Sample A bit

1 = Scans inputs

0 = Does not scan inputs

bit 9-8 **Unimplemented:** Read as '0'

bit 7 **BUFS**: Buffer Fill Status bit⁽¹⁾

1 = A/D is currently filling ADC1BUF8-ADC1BUFF, user should access data in ADC1BUF0-ADC1BUF7

0 = A/D is currently filling ADC1BUF0-ADC1BUF7, user should access data ADC1BUF8-ADC1BUFF

Note 1: These bits are only applicable when the buffer is used in FIFO mode (BUFREGEN = 0). In addition, BUFS is only used when BUFM = 1.

REGISTER 25-2: AD1CON2: ADC1 CONTROL REGISTER 2 (CONTINUED)

bit 6-2 **SMPI<4:0>:** Interrupt Sample/DMA Increment Rate Select bits

When DMAEN = 1:

- 11111 = Increments the DMA address after completion of the 32nd sample/conversion operation 11110 = Increments the DMA address after completion of the 31st sample/conversion operation
- •
- •
- 00001 = Increments the DMA address after completion of the 2nd sample/conversion operation
- 00000 = Increments the DMA address after completion of each sample/conversion operation

When DMAEN = 0:

- 11111 = Interrupts at the completion of the conversion for each 32nd sample
- 11110 = Interrupts at the completion of the conversion for each 31st sample
- •
- •
- 00001 = Interrupts at the completion of the conversion for every other sample
- 00000 = Interrupts at the completion of the conversion for each sample
- bit 1 **BUFM:** Buffer Fill Mode Select bit⁽¹⁾
 - 1 = Starts buffer filling at ADC1BUF0 on the first interrupt and ADC1BUF8 on the next interrupt
 - 0 = Always starts filling buffer at ADC1BUF0
- bit 0 ALTS: Alternate Input Sample Mode Select bit
 - 1 = Uses channel input selects for Sample A on the first sample and Sample B on the next sample
 - 0 = Always uses channel input selects for Sample A
- **Note 1:** These bits are only applicable when the buffer is used in FIFO mode (BUFREGEN = 0). In addition, BUFS is only used when BUFM = 1.

REGISTER 25-3: AD1CON3: ADC1 CONTROL REGISTER 3

R/W-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADRC	EXTSAM	PUMPEN	SAMC4	SAMC3	SAMC2	SAMC1	SAMC0
bit 15							bit 8

| R/W-0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| ADCS7 | ADCS6 | ADCS5 | ADCS4 | ADCS3 | ADCS2 | ADCS1 | ADCS0 |
| bit 7 | | | | | | | bit 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 15 ADRC: ADC1 Conversion Clock Source bit

1 = RC clock

0 = Clock derived from system clock

bit 14 **EXTSAM:** Extended Sampling Time bit

1 = A/D is still sampling after SAMP = 0

0 = A/D is finished sampling

bit 13 **PUMPEN:** Charge Pump Enable bit

1 = Charge pump for switches is enabled

0 = Charge pump for switches is disabled

bit 12-8 **SAMC<4:0>:** Auto-Sample Time Select bits

11111 = **31** TAD

•

•

•

00001 = 1 TAD

00000 **= 0 TAD**

bit 7-0 ADCS<7:0>: ADC1 Conversion Clock Select bits

11111111 = 256 • TCY = TAD

•

00000001 = 2 • TCY = TAD

00000000 = Tcy = Tab

REGISTER 25-4: AD1CON4: ADC1 CONTROL REGISTER 4

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_		_	_	_			_
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
_	_	_	_	_		DMABL<2:0> ⁽¹)
bit 7							bit 0

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

bit 15-3 **Unimplemented:** Read as '0'

bit 2-0 DMABL<2:0>: DMA Buffer Size Select bits⁽¹⁾

111 = Allocates 128 words of buffer to each analog input

110 = Allocates 64 words of buffer to each analog input

101 = Allocates 32 words of buffer to each analog input

100 = Allocates 16 words of buffer to each analog input

011 = Allocates 8 words of buffer to each analog input

010 = Allocates 4 words of buffer to each analog input

001 = Allocates 2 words of buffer to each analog input

000 = Allocates 1 word of buffer to each analog input

Note 1: The DMABL<2:0> bits are only used when AD1CON1<11> = 1 and AD1CON<12> = 0; otherwise, their value is ignored.

REGISTER 25-5: AD1CON5: ADC1 CONTROL REGISTER 5

R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
ASEN	LPEN	CTMREQ	BGREQ	_	_	ASINT1	ASINT0
bit 15							bit 8

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	_	_	WM1	WM0	CM1	CM0
bit 7							bit 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 15 ASEN: Auto-Scan Enable bit

1 = Auto-scan is enabled

0 = Auto-scan is disabled

bit 14 LPEN: Low-Power Enable bit

1 = Low power is enabled after scan0 = Full power is enabled after scan

bit 13 CTMREQ: CTMU Request bit

1 = CTMU is enabled when the A/D is enabled and active

0 = CTMU is not enabled by the A/D

bit 12 **BGREQ:** Band Gap Request bit

1 = Band gap is enabled when the A/D is enabled and active

0 = Band gap is not enabled by the A/D

bit 11-10 Unimplemented: Read as '0'

bit 9-8 ASINT<1:0>: Auto-Scan (Threshold Detect) Interrupt Mode bits

11 = Interrupt after Threshold Detect sequence completed and valid compare has occurred

10 = Interrupt after valid compare has occurred

01 = Interrupt after Threshold Detect sequence completed

00 = No interrupt

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

bit 3-2 WM<1:0>: Write Mode bits

11 = Reserved

10 = Auto-compare only (conversion results are not saved, but interrupts are generated when a valid match occurs, as defined by the CMx and ASINTx bits)

01 = Convert and save (conversion results are saved to locations as determined by the register bits when a match occurs, as defined by the CMx bits)

00 = Legacy operation (conversion data is saved to a location determined by the buffer register bits)

bit 1-0 CM<1:0>: Compare Mode bits

11 = Outside Window mode (valid match occurs if the conversion result is outside of the window defined by the corresponding buffer pair)

10 = Inside Window mode (valid match occurs if the conversion result is inside the window defined by the corresponding buffer pair)

01 = Greater Than mode (valid match occurs if the result is greater than the value in the corresponding buffer register)

00 = Less Than mode (valid match occurs if the result is less than the value in the corresponding buffer register)

REGISTER 25-6: AD1CHS: ADC1 SAMPLE SELECT REGISTER

| R/W-0 |
|--------|--------|--------|--------|--------|--------|--------|--------|
| CH0NB2 | CH0NB1 | CH0NB0 | CH0SB4 | CH0SB3 | CH0SB2 | CH0SB1 | CH0SB0 |
| bit 15 | | | | | | | bit 8 |

| R/W-0 |
|--------|--------|--------|--------|--------|--------|--------|--------|
| CH0NA2 | CH0NA1 | CH0NA0 | CH0SA4 | CH0SA3 | CH0SA2 | CH0SA1 | CH0SA0 |
| bit 7 | | | | | | | bit 0 |

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

```
bit 15-13
              CH0NB<2:0>: Sample B Channel 0 Negative Input Select bits
              1xx = Unimplemented
              011 = Unimplemented
              010 = AN1
              001 = Unimplemented
              000 = VREF-/AVSS
              CH0SB<4:0>: Sample B Channel 0 Positive Input Select bits
bit 12-8
              11111 = VBAT/2<sup>(1)</sup>
              11110 = AVDD<sup>(1)</sup>
              11101 = AVss<sup>(1)</sup>
              11100 = Band Gap Voltage (VBG) reference<sup>(1)</sup>
              11011 = VBG/2^{(1)}
              01110 = CTMU
              01101 = CTMU temperature sensor input (does not require AD1CTMENL<12> to be set)
              01100 = AN12<sup>(2)</sup>
              01011 = AN11<sup>(2)</sup>
              01010 = AN10^{(2)}
              01001 = AN9
              01000 = Unimplemented
              00111 = AN7
              00110 = AN6
              00101 = AN5
              00100 = AN4
              00011 = AN3
              00010 = AN2
              00001 = AN1
              00000 = AN0
bit 7-5
              CH0NA<2:0>: Sample A Channel 0 Negative Input Select bits
              Same definitions as for CHONB<2:0>.
bit 4-0
              CH0SA<4:0>: Sample A Channel 0 Positive Input Select bits
              Same definitions as for CHOSB<4:0>.
```

Note 1: These input channels do not have corresponding memory-mapped result buffers.

2: These channels are unimplemented in 28-pin devices.

REGISTER 25-7: ANCFG: A/D BAND GAP REFERENCE CONFIGURATION REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
_	_	_	_	_	_	VBG2EN	VBGEN
bit 7							bit 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 15-2 Unimplemented: Read as '0'

bit 1 VBG2EN: A/D Input VBG/2 Enable bit

1 = Band Gap Voltage, divided by two reference (VBG/2), is enabled 0 = Band Gap Voltage, divided by two reference (VBG/2), is disabled

bit 0 VBGEN: A/D Input VBG Enable bit

1 = Band Gap Voltage (VBG) reference is enabled 0 = Band Gap Voltage (VBG) reference is disabled

REGISTER 25-8: AD1CHITL: ADC1 SCAN COMPARE HIT REGISTER (LOW WORD)

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0
_	_	_		CHH<1	12:9> ⁽¹⁾		_
bit 15							bit 8

| R/W-0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| | | | CHH | <7:0> | | | |
| bit 7 | | | | | | | bit 0 |

Legend:

bit 8

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 15-13 Unimplemented: Read as '0'

bit 12-9 CHH<12:9>: ADC1 Compare Hit bits⁽¹⁾

If CM<1:0> = 11:

1 = A/D Result Buffer n has been written with data or a match has occurred

0 = A/D Result Buffer n has not been written with data

For All Other Values of CM<1:0>:

1 = A match has occurred on A/D Result Channel n 0 = No match has occurred on A/D Result Channel n

Unimplemented: Read as '0'

bit 7-0 CHH<8:0>: ADC1 Compare Hit bits

If CM<1:0> = 11:

1 = A/D Result Buffer n has been written with data or a match has occurred

0 = A/D Result Buffer n has not been written with data

For All Other Values of CM<1:0>:

1 = A match has occurred on A/D Result Channel n

0 = No match has occurred on A/D Result Channel n

Note 1: The CHH<12:10> bits are unimplemented in 28-pin devices, read as '0'.

REGISTER 25-9: AD1CSSH: ADC1 INPUT SCAN SELECT REGISTER (HIGH WORD)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
		CSS<31:27>			_	_	_
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 7							bit 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 15-11 CSS<31:27>: ADC1 Input Scan Selection bits

1 = Includes corresponding channel for input scan

0 = Skips channel for input scan

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

REGISTER 25-10: AD1CSSL: ADC1 INPUT SCAN SELECT REGISTER (LOW WORD)

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0
_	CSS<14:9> ⁽¹⁾						_
bit 15							bit 8

| R/W-0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| | | | CSS- | <7:0> | | | |
| bit 7 | | | | | | | bit 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 15 **Unimplemented:** Read as '0'

bit 14-9 CSS<14:9>: ADC1 Input Scan Selection bits⁽¹⁾

1 = Includes corresponding channel for input scan

0 = Skips channel for input scan

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

bit 7-0 CSS<7:0>: ADC1 Input Scan Selection bits

1 = Includes corresponding channel for input scan

0 = Skips channel for input scan

Note 1: The CSS<12:10> bits are unimplemented in 28-pin devices, read as '0'.

REGISTER 25-11: AD1CTMENL: ADC1 CTMU ENABLE REGISTER (LOW WORD)(1)

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0
_	_	_		CTMEN<	<12:9> ⁽²⁾		_
bit 15							bit 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
CTMEN<7:0>							
bit 7							bit 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 15-13 **Unimplemented:** Read as '0'

bit 12-9 CTMEN<12:9>: CTMU Enable During Conversion bits⁽²⁾

1 = CTMU is enabled and connected to the selected channel during conversion

0 = CTMU is not connected to this channel

bit 8 Unimplemented: Read as '0'

bit 7-0 CTMEN<7:0>: CTMU Enable During Conversion bits

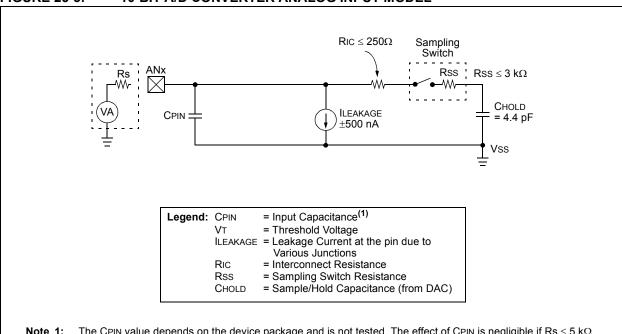
1 = CTMU is enabled and connected to the selected channel during conversion

0 = CTMU is not connected to this channel

Note 1: The actual number of channels available depends on which channels are implemented on a specific device. For more information, refer to Table 1-1 and Table 1-2 in Section 1.0 "Device Overview". Unimplemented channels are read as '0'.

2: The CTMEN<12:10> bits are unimplemented in 28-pin devices, read as '0'.

FIGURE 25-3: 10-BIT A/D CONVERTER ANALOG INPUT MODEL



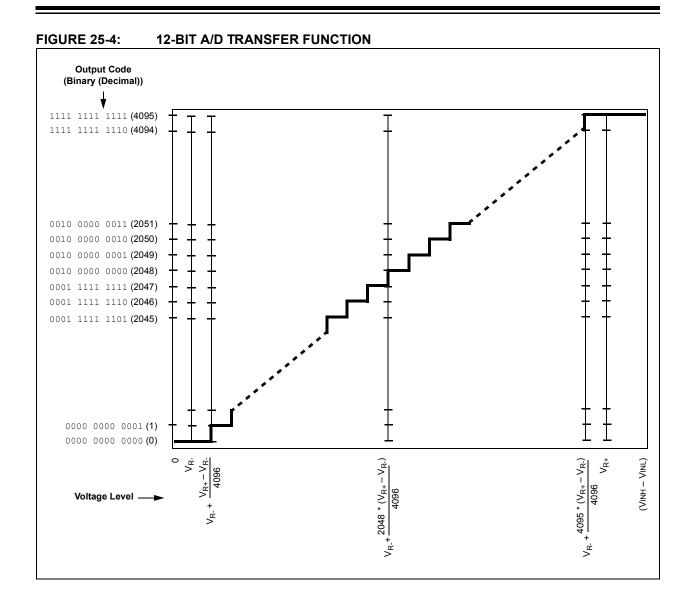
Note 1: The CPIN value depends on the device package and is not tested. The effect of CPIN is negligible if Rs \leq 5 k Ω .

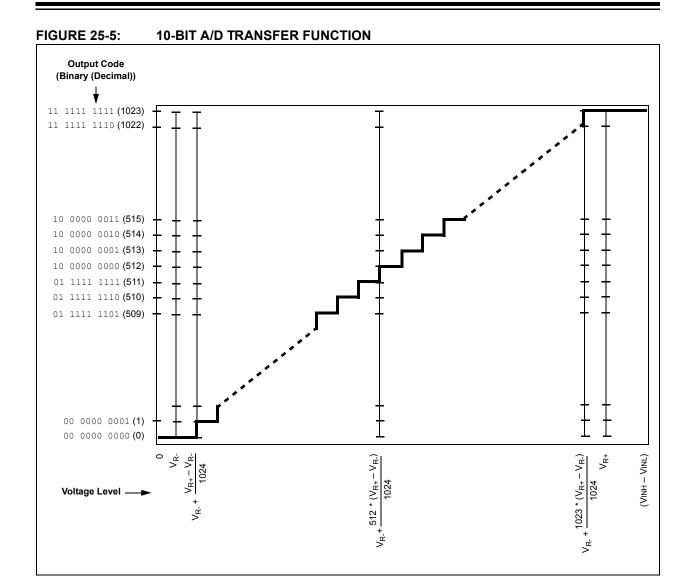
EQUATION 25-1: A/D CONVERSION CLOCK PERIOD

$$TAD = TCY(ADCS + 1)$$

$$ADCS = \frac{TAD}{TCY} - 1$$

Note: Based on Tcy = 2/Fosc; Doze mode and PLL are disabled.





TRIPLE COMPARATOR 26.0 **MODULE**

Note: This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the "dsPIC33/PIC24 Family Reference Manual", "Scalable Comparator Module"

(DS39734). The information in this data sheet supersedes the information in the FRM

The triple comparator module provides three dual input comparators. The inputs to the comparator can be configured to use any one of five external analog inputs (CxINA, CxINB, CxINC, CxIND and VREF+) and a

voltage reference input from one of the internal band gap references or the comparator voltage reference generator (VBG, VBG/2 and CVREF).

The comparator outputs may be directly connected to the CxOUT pins. When the respective COE equals '1', the I/O pad logic makes the unsynchronized output of the comparator available on the pin.

A simplified block diagram of the module in shown in Figure 26-1. Diagrams of the possible individual comparator configurations are shown in Figure 26-2, Figure 26-3 and Figure 26-4.

Each comparator has its own control register, CMxCON (Register 26-1), for enabling and configuring its operation. The output and event status of all three comparators is provided in the CMSTAT register (Register 26-2).

FIGURE 26-1: TRIPLE COMPARATOR MODULE BLOCK DIAGRAM

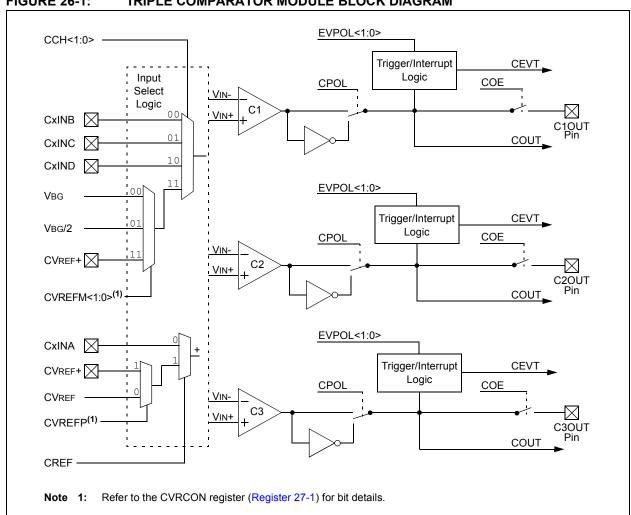


FIGURE 26-2: INDIVIDUAL COMPARATOR CONFIGURATIONS WHEN CREF = 0

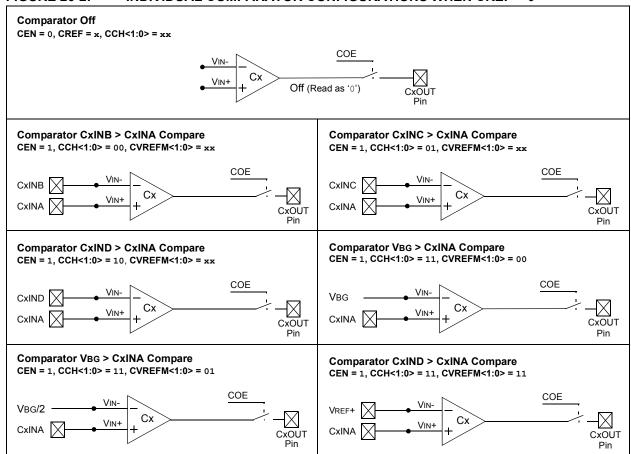


FIGURE 26-3: INDIVIDUAL COMPARATOR CONFIGURATIONS WHEN CREF = 1 AND CVREFP = 0

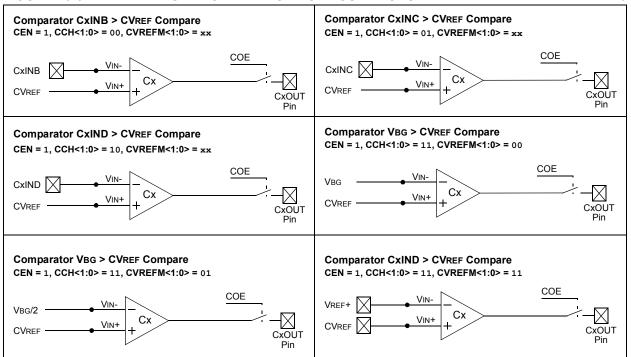
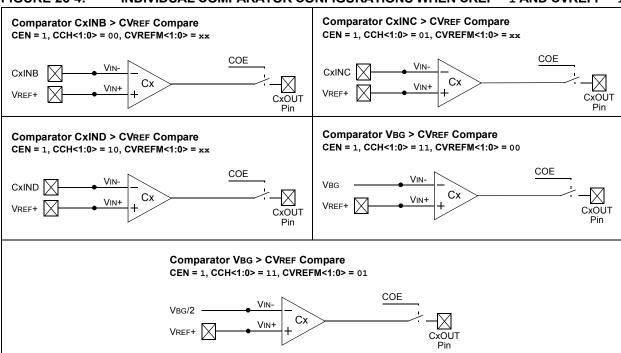


FIGURE 26-4: INDIVIDUAL COMPARATOR CONFIGURATIONS WHEN CREF = 1 AND CVREFP = 1



REGISTER 26-1: CMxCON: COMPARATOR x CONTROL REGISTERS (COMPARATORS 1 THROUGH 3)

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	R/W-0, HS	R-0, HSC
CON	COE	CPOL	_	_	_	CEVT	COUT
bit 15							bit 8

R/W-0	R/W-0	U-0	R/W-0	U-0	U-0	R/W-0	R/W-0
EVPOL1 ⁽¹⁾	EVPOL0 ⁽¹⁾	_	CREF	_	_	CCH1	CCH0
bit 7							bit 0

Legend:	HS = Hardware Settable bit	HSC = Hardware Settable/Clearable bit	
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 15 CON: Comparator Enable bit

1 = Comparator is enabled

0 = Comparator is disabled

bit 14 **COE:** Comparator Output Enable bit

1 = Comparator output is present on the CxOUT pin

0 = Comparator output is internal only

bit 13 CPOL: Comparator Output Polarity Select bit

1 = Comparator output is inverted

0 = Comparator output is not inverted

bit 12-10 Unimplemented: Read as '0'

bit 9 **CEVT:** Comparator Event bit

1 = Comparator event that is defined by the EVPOL<1:0> bits has occurred; subsequent triggers and interrupts are disabled until the bit is cleared

0 = Comparator event has not occurred

bit 8 **COUT:** Comparator Output bit

When CPOL = 0:

1 = VIN+ > VIN-

0 = VIN+ < VIN-

When CPOL = 1:

1 = VIN+ < VIN-

0 = VIN+ > VIN-

bit 7-6 **EVPOL<1:0>:** Trigger/Event/Interrupt Polarity Select bits⁽¹⁾

11 = Trigger/event/interrupt is generated on any change of the comparator output (while CEVT = 0)

10 = Trigger/event/interrupt is generated on the high-to-low transition of the comparator output

01 = Trigger/event/interrupt is generated on the low-to-high transition of the comparator output

00 = Trigger/event/interrupt generation is disabled

bit 5 **Unimplemented:** Read as '0'

bit 4 CREF: Comparator Reference Select bit (non-inverting input)

 $\ensuremath{\mathtt{1}}$ = Non-inverting input connects to the internal CVREF voltage

0 = Non-inverting input connects to the CxINA pin

bit 3-2 **Unimplemented:** Read as '0'

Note 1: If the EVPOL<1:0> bits are set to a value other than '00', the first interrupt generated will occur on any transition of COUT. Subsequent interrupts will occur based on the EVPOLx bits setting.

REGISTER 26-1: CMxCON: COMPARATOR x CONTROL REGISTERS (COMPARATORS 1 THROUGH 3) (CONTINUED)

bit 1-0 **CCH<1:0>:** Comparator Channel Select bits

- 11 = Inverting input of the comparator connects to the internal selectable reference voltage specified by the CVREFM<1:0> bits in the CVRCON register
- 10 = Inverting input of the comparator connects to the CxIND pin
- 01 = Inverting input of the comparator connects to the CxINC pin
- 00 = Inverting input of the comparator connects to the CxINB pin

Note 1: If the EVPOL<1:0> bits are set to a value other than '00', the first interrupt generated will occur on any transition of COUT. Subsequent interrupts will occur based on the EVPOLx bits setting.

REGISTER 26-2: CMSTAT: COMPARATOR MODULE STATUS REGISTER

R/W-0	U-0	U-0	U-0	U-0	R-0, HSC	R-0, HSC	R-0, HSC
CMIDL	_	_	_	_	C3EVT	C2EVT	C1EVT
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R-0, HSC	R-0, HSC	R-0, HSC
_	_	_	_	_	C3OUT	C2OUT	C1OUT
bit 7							bit 0

Legend:	HSC = Hardware Setta	HSC = Hardware Settable/Clearable bit					
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 15 CMIDL: Comparator Stop in Idle Mode bit

1 = Discontinues operation of all comparators when device enters Idle mode

0 = Continues operation of all enabled comparators in Idle mode

bit 14-11 **Unimplemented:** Read as '0'

bit 10 **C3EVT:** Comparator 3 Event Status bit (read-only)

Shows the current event status of Comparator 3 (CM3CON<9>).

bit 9 **C2EVT:** Comparator 2 Event Status bit (read-only)

Shows the current event status of Comparator 2 (CM2CON<9>).

bit 8 **C1EVT:** Comparator 1 Event Status bit (read-only)

Shows the current event status of Comparator 1 (CM1CON<9>).

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

bit 2 **C3OUT:** Comparator 3 Output Status bit (read-only)

Shows the current output of Comparator 3 (CM3CON<8>).

bit 1 **C2OUT:** Comparator 2 Output Status bit (read-only)

Shows the current output of Comparator 2 (CM2CON<8>).

bit 0 C10UT: Comparator 1 Output Status bit (read-only)

Shows the current output of Comparator 1 (CM1CON<8>).

PIC24FJ128GB204 FAMILY	
NOTES:	

27.0 COMPARATOR VOLTAGE REFERENCE

Note:

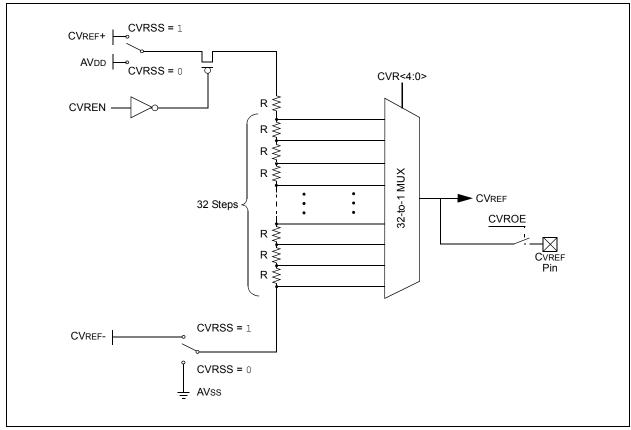
This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the "dsPIC33/PIC24 Family Reference Manual", "Comparator Voltage Reference Module" (DS39709). The information in this data sheet supersedes the information in the FRM.

27.1 Configuring the Comparator Voltage Reference

The comparator voltage reference module is controlled through the CVRCON register (Register 27-1). The comparator voltage reference provides a range of output voltages with 32 distinct levels. The comparator reference supply voltage can come from either VDD and Vss, or the external CVREF+ and CVREF- pins. The voltage source is selected by the CVRSS bit (CVRCON<5>).

The settling time of the comparator voltage reference must be considered when changing the CVREF output.

FIGURE 27-1: COMPARATOR VOLTAGE REFERENCE BLOCK DIAGRAM



REGISTER 27-1: CVRCON: COMPARATOR VOLTAGE REFERENCE CONTROL REGISTER

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
_	_	_	_	_	CVREFP	CVREFM1	CVREFM0
bit 15							bit 8

| R/W-0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| CVREN | CVROE | CVRSS | CVR4 | CVR3 | CVR2 | CVR1 | CVR0 |
| bit 7 | | | | | | | bit 0 |

Legend:

bit 4-0

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 15-11 **Unimplemented:** Read as '0'

bit 10 **CVREFP:** Voltage Reference Select bit (valid only when CREF is '1')

1 = VREF+ is used as a reference voltage to the comparators

0 = The CVR<4:0> bits (4-bit DAC) within this module provide the reference voltage to the comparators

bit 9-8 **CVREFM<1:0>:** Band Gap Reference Source Select bits (valid only when CCH<1:0> = 11)

00 = Band gap voltage is provided as an input to the comparators

01 = Band gap voltage, divided by two, is provided as an input to the comparators

10 = Reserved

11 = VREF+ pin is provided as an input to the comparators

bit 7 **CVREN:** Comparator Voltage Reference Enable bit

1 = CVREF circuit is powered on0 = CVREF circuit is powered down

bit 6 **CVROE:** Comparator VREF Output Enable bit

1 = CVREF voltage level is output on the CVREF pin

0 = CVREF voltage level is disconnected from the CVREF pin

bit 5 CVRSS: Comparator VREF Source Selection bit

1 = Comparator reference source, CVRSRC = VREF+ - VREF-0 = Comparator reference source, CVRSRC = AVDD - AVSS

CVR<4:0>: Comparator VREF Value Selection bits

CVREF = (CVR<4:0>/32) • (CVRSRC)

28.0 CHARGE TIME MEASUREMENT UNIT (CTMU)

Note:

This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information on the Charge Time Measurement Unit, refer to the "dsPIC33/PIC24 Family Reference Manual", "Charge Time Measurement Unit (CTMU) with Threshold Detect" (DS39743). The information in this data sheet supersedes the information in the FRM.

The Charge Time Measurement Unit (CTMU) is a flexible analog module that provides charge measurement, accurate differential time measurement between pulse sources and asynchronous pulse generation. Its key features include:

- · Thirteen external edge input trigger sources
- · Polarity control for each edge source
- · Control of edge sequence
- Control of response to edge levels or edge transitions
- · Time measurement resolution of one nanosecond
- Accurate current source suitable for capacitive measurement

Together with other on-chip analog modules, the CTMU can be used to precisely measure time, measure capacitance, measure relative changes in capacitance or generate output pulses that are independent of the system clock. The CTMU module is ideal for interfacing with capacitive-based touch sensors.

The CTMU is controlled through three registers: CTMUCON1, CTMUCON2 and CTMUICON. CTMUCON1 enables the module and controls the mode of operation of the CTMU, as well as controlling edge sequencing. CTMUCON2 controls edge source selection and edge source polarity selection. The CTMUICON register selects the current range of the current source and trims the current.

28.1 Measuring Capacitance

The CTMU module measures capacitance by generating an output pulse, with a width equal to the time between edge events, on two separate input channels. The pulse edge events to both input channels can be selected from four sources: two internal peripheral modules (OC1 and Timer1) and up to 13 external pins (CTED1 through CTED13). This pulse is used with the module's precision current source to calculate capacitance according to the relationship:

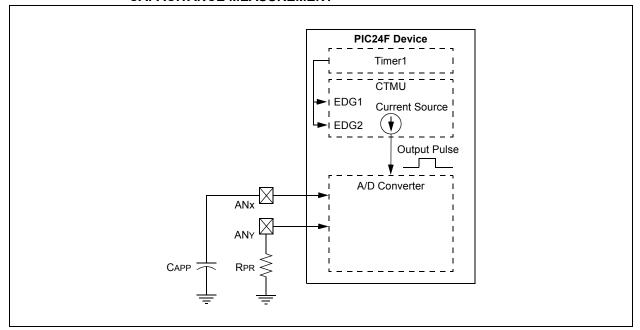
EQUATION 28-1:

$$I = C \bullet \frac{dV}{dT}$$

For capacitance measurements, the A/D Converter samples an external Capacitor (CAPP) on one of its input channels after the CTMU output's pulse. A Precision Resistor (RPR) provides current source calibration on a second A/D channel. After the pulse ends, the converter determines the voltage on the capacitor. The actual calculation of capacitance is performed in software by the application.

Figure 28-1 illustrates the external connections used for capacitance measurements and how the CTMU and A/D modules are related in this application. This example also shows the edge events coming from Timer1, but other configurations using external edge sources are possible. A detailed discussion on measuring capacitance and time with the CTMU module is provided in the "dsPIC33/PIC24 Family Reference Manual", "Charge Time Measurement Unit (CTMU) with Threshold Detect" (DS39743).

FIGURE 28-1: TYPICAL CONNECTIONS AND INTERNAL CONFIGURATION FOR CAPACITANCE MEASUREMENT



28.2 Measuring Time

Time measurements on the pulse width can be similarly performed using the A/D module's Internal Capacitor (CAD) and a precision resistor for current calibration. Figure 28-2 displays the external connections used for time measurements, and how the CTMU and A/D modules are related in this application. This example also shows both edge events coming from the external CTED pins, but other configurations using internal edge sources are possible.

28.3 Pulse Generation and Delay

The CTMU module can also generate an output pulse with edges that are not synchronous with the device's system clock. More specifically, it can generate a pulse with a programmable delay from an edge event input to the module.

When the module is configured for pulse generation delay by setting the TGEN bit (CTMUCON1<12>), the internal current source is connected to the B input of Comparator 2. A Capacitor (CDELAY) is connected to the Comparator 2 pin, C2INB, and the Comparator voltage Reference, CVREF, is connected to C2INA. CVREF is then configured for a specific trip point. The module begins to charge CDELAY when an edge event is detected. When CDELAY charges above the CVREF trip point, a pulse is output on CTPLS. The length of the pulse delay is determined by the value of CDELAY and the CVREF trip point.

Figure 28-3 illustrates the external connections for pulse generation, as well as the relationship of the different analog modules required. While CTED1 is shown as the input pulse source, other options are available. A detailed discussion on pulse generation with the CTMU module is provided in the "dsPIC33/PIC24 Family Reference Manual".

FIGURE 28-2: TYPICAL CONNECTIONS AND INTERNAL CONFIGURATION FOR TIME MEASUREMENT

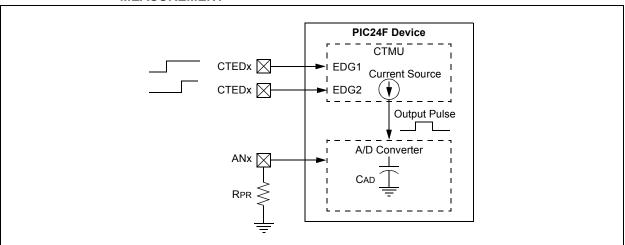
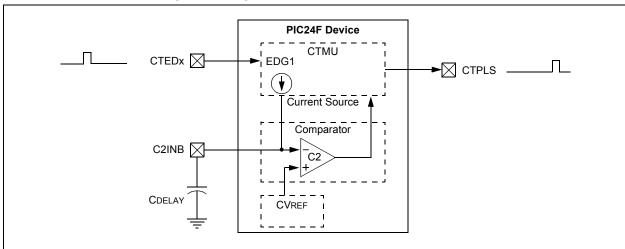


FIGURE 28-3: TYPICAL CONNECTIONS AND INTERNAL CONFIGURATION FOR PULSE DELAY GENERATION



REGISTER 28-1: CTMUCON1: CTMU CONTROL REGISTER 1

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CTMUEN	_	CTMUSIDL	TGEN	EDGEN	EDGSEQEN	IDISSEN	CTTRIG
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	_
bit 7							bit 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 15 CTMUEN: CTMU Enable bit

1 = Module is enabled

0 = Module is disabled

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

bit 13 CTMUSIDL: CTMU Stop in Idle Mode bit

1 = Discontinues module operation when device enters Idle mode

0 = Continues module operation in Idle mode

bit 12 **TGEN:** Time Generation Enable bit

1 = Enables edge delay generation

0 = Disables edge delay generation

bit 11 EDGEN: Edge Enable bit

1 = Edges are not blocked

0 = Edges are blocked

bit 10 EDGSEQEN: Edge Sequence Enable bit

1 = Edge 1 event must occur before Edge 2 event can occur

0 = No edge sequence is needed

bit 9 IDISSEN: Analog Current Source Control bit

1 = Analog current source output is grounded

0 = Analog current source output is not grounded

bit 8 CTTRIG: CTMU Trigger Control bit

1 = Trigger output is enabled

0 = Trigger output is disabled

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

REGISTER 28-2: CTMUCON2: CTMU CONTROL REGISTER 2

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
EDG1MOD	EDG1POL	EDG1SEL3	EDG1SEL2	EDG1SEL1	EDG1SEL0	EDG2STAT	EDG1STAT
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0
EDG2MOD	EDG2POL	EDG2SEL3	EDG2SEL2	EDG2SEL1	EDG2SEL0		_
bit 7							bit 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 15 EDG1MOD: Edge 1 Edge-Sensitive Select bit

1 = Input is edge-sensitive

0 = Input is level-sensitive

bit 14 **EDG1POL:** Edge 1 Polarity Select bit

1 = Edge 1 is programmed for a positive edge response 0 = Edge 1 is programmed for a negative edge response

bit 13-10 EDG1SEL<3:0>: Edge 1 Source Select bits

1111 = Edge 1 source is the Comparator 3 output

1110 = Edge 1 source is the Comparator 2 output

1101 = Edge 1 source is the Comparator 1 output

1100 = Edge 1 source is IC3

1011 = Edge 1 source is IC2

1010 = Edge 1 source is IC1

1001 = Edge 1 source is CTED8

1000 = Edge 1 source is CTED7⁽¹⁾

0111 = Edge 1 source is CTED6

0110 = Edge 1 source is CTED5

0101 = Edge 1 source is CTED4

0100 = Edge 1 source is CTED3

0011 = Edge 1 source is CTED1

0010 = Edge 1 source is CTED2

0001 = Edge 1 source is OC1

0000 = Edge 1 source is Timer1

bit 9 **EDG2STAT:** Edge 2 Status bit

Indicates the status of Edge 2 and can be written to control the current source.

1 = Edge 2 has occurred

0 = Edge 2 has not occurred

bit 8 EDG1STAT: Edge 1 Status bit

Indicates the status of Edge 1 and can be written to control the current source.

1 = Edge 1 has occurred

0 = Edge 1 has not occurred

bit 7 **EDG2MOD:** Edge 2 Edge-Sensitive Select bit

1 = Input is edge-sensitive

0 = Input is level-sensitive

bit 6 EDG2POL: Edge 2 Polarity Select bit

1 = Edge 2 is programmed for a positive edge

0 = Edge 2 is programmed for a negative edge

Note 1: Edge source, CTED7, is not available in 28-pin packages.

REGISTER 28-2: CTMUCON2: CTMU CONTROL REGISTER 2 (CONTINUED)

```
bit 5-2
             EDG2SEL<3:0>: Edge 2 Source Select bits
             1111 = Edge 2 source is the Comparator 3 output
             1110 = Edge 2 source is the Comparator 2 output
             1101 = Edge 2 source is the Comparator 1 output
             1100 = Unimplemented; do not use
             1011 = Edge 2 source is IC3
             1010 = Edge 2 source is IC2
             1001 = Edge 2 source is IC1
             1000 = Edge 2 source is CTED13
             0111 = Edge 2 source is CTED12
             0110 = Edge 2 source is CTED11
             0101 = Edge 2 source is CTED10
             0100 = Edge 2 source is CTED9
             0011 = Edge 2 source is CTED1
             0010 = Edge 2 source is CTED2
             0001 = Edge 2 source is OC1
             0000 = Edge 2 source is Timer1
bit 1-0
             Unimplemented: Read as '0'
```

Note 1: Edge source, CTED7, is not available in 28-pin packages.

REGISTER 28-3: CTMUICON: CTMU CURRENT CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ITRIM5	ITRIM4	ITRIM3	ITRIM2	ITRIM1	ITRIM0	IRNG1	IRNG0
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
_	_	_	_	_	_	_	
bit 7							bit 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 15-10 ITRIM<5:0>: Current Source Trim bits

011111 = Maximum positive change from nominal current

011110

•

.

000001 = Minimum positive change from nominal current

000000 = Nominal current output specified by IRNG<1:0>

111111 = Minimum negative change from nominal current

•

•

•

100010

100001 = Maximum negative change from nominal current

bit 9-8 IRNG<1:0>: Current Source Range Select bits

11 = 100 × Base Current

10 = 10 × Base Current

01 = Base current level (0.55 μ A nominal)

00 = 1000 × Base Current

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

PICZ4FJ IZOGDZU4 FAIVIIL I									
			ZOGDZO4 I AIVIIL I	ZOGDZO4 I AIVIIL I	ZOGDZO4 I AIVIIL I				

29.0 HIGH/LOW-VOLTAGE DETECT (HLVD)

Note: This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information on the High/Low-Voltage Detect, refer to the

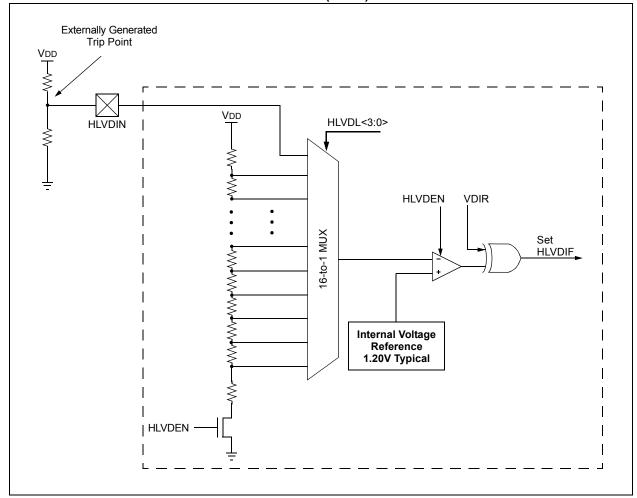
"dsPIC33/PIC24 Family Reference Manual", "High-Level Integration with Programmable High/Low-Voltage Detect (HLVD)" (DS39725).

The High/Low-Voltage Detect (HLVD) module is a programmable circuit that allows the user to specify both the device voltage trip point and the direction of change.

An interrupt flag is set if the device experiences an excursion past the trip point in the direction of change. If the interrupt is enabled, the program execution will branch to the interrupt vector address and the software can then respond to the interrupt.

The HLVD Control register (see Register 29-1) completely controls the operation of the HLVD module. This allows the circuitry to be "turned off" by the user under software control, which minimizes the current consumption for the device.

FIGURE 29-1: HIGH/LOW-VOLTAGE DETECT (HLVD) MODULE BLOCK DIAGRAM



REGISTER 29-1: HLVDCON: HIGH/LOW-VOLTAGE DETECT CONTROL REGISTER

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
HLVDEN	_	LSIDL	_	_	_	_	_
bit 15							bit 8

R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
VDIR	BGVST	IRVST	_	HLVDL3	HLVDL2	HLVDL1	HLVDL0
bit 7							bit 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 15 **HLVDEN:** High/Low-Voltage Detect Power Enable bit

> 1 = HLVD is enabled 0 = HLVD is disabled

bit 14 Unimplemented: Read as '0'

bit 13 LSIDL: HLVD Stop in Idle Mode bit

1 = Discontinues module operation when device enters Idle mode

0 = Continues module operation in Idle mode

bit 12-8 Unimplemented: Read as '0'

bit 7 VDIR: Voltage Change Direction Select bit

1 = Event occurs when voltage equals or exceeds the trip point (HLVDL<3:0>)

0 = Event occurs when voltage equals or falls below the trip point (HLVDL<3:0>)

bit 6 **BGVST:** Band Gap Voltage Stable Flag bit

1 = Indicates that the band gap voltage is stable

0 = Indicates that the band gap voltage is unstable

bit 5 IRVST: Internal Reference Voltage Stable Flag bit

> 1 = Internal reference voltage is stable; the High-Voltage Detect logic generates the interrupt flag at the specified voltage range

0 = Internal reference voltage is unstable; the High-Voltage Detect logic will not generate the interrupt flag at the specified voltage range and the HLVD interrupt should not be enabled

Unimplemented: Read as '0' bit 4

bit 3-0 HLVDL<3:0>: High/Low-Voltage Detection Limit bits

1111 = External analog input is used (input comes from the HLVDIN pin)

1110 = Trip Point 1⁽¹⁾

1101 = Trip Point 2⁽¹⁾

1100 = Trip Point 3⁽¹⁾

0100 = Trip Point 11⁽¹⁾

00xx = Unused

Note 1: For the actual trip point, see Section 33.0 "Electrical Characteristics".

30.0 SPECIAL FEATURES

Note:

This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the following sections of the "dsPIC33/PIC24 Family Reference Manual". The information in this data sheet supersedes the information in the FRMs.

- "Watchdog Timer (WDT)" (DS39697)
- "High-Level Device Integration" (DS39719)
- "Programming and Diagnostics" (DS39716)

PIC24FJ128GB204 family devices include several features intended to maximize application flexibility and reliability, and minimize cost through elimination of external components. These are:

- · Flexible Configuration
- Watchdog Timer (WDT)
- · Code Protection
- · JTAG Boundary Scan Interface
- In-Circuit Serial Programming™ (ICSP™)
- In-Circuit Emulation (ICE)

30.1 Configuration Bits

The Configuration bits can be programmed (read as '0'), or left unprogrammed (read as '1'), to select various device configurations. These bits are mapped starting at program memory location, F80000h. A detailed explanation of the various bit functions is provided in Register 30-1 through Register 30-6.

Note that address, F80000h, is beyond the user program memory space. In fact, it belongs to the configuration memory space (800000h-FFFFFFh), which can only be accessed using Table Reads and Table Writes.

30.1.1 CONSIDERATIONS FOR CONFIGURING PIC24FJ128GB204 FAMILY DEVICES

In PIC24FJ128GB204 family devices, the Configuration bytes are implemented as volatile memory. This means that configuration data must be programmed each time the device is powered up. Configuration data is stored in the four words at the top of the on-chip program memory space, known as the Flash Configuration Words. Their specific locations are shown in Table 30-1. These are packed representations of the actual device Configuration bits, whose actual locations are distributed among several locations in configuration space. The configuration data is automatically loaded from the Flash Configuration Words to the proper Configuration registers during device Resets.

Note: Configuration data is reloaded on all types of device Resets.

When creating applications for these devices, users should always specifically allocate the location of the Flash Configuration Word for configuration data. This is to make certain that program code is not stored in this address when the code is compiled.

The upper byte of all Flash Configuration Words in program memory should always be: '0000 0000'. This makes them appear to be NOP instructions in the remote event that their locations are ever executed by accident. Since Configuration bits are not implemented in the corresponding locations, writing '0's to these locations has no effect on device operation.

Note: Performing a page erase operation on the last page of program memory clears the Flash Configuration Words, enabling code protection as a result. Therefore, users should avoid performing page erase operations on the last page of program memory.

TABLE 30-1: FLASH CONFIGURATION WORD LOCATIONS FOR THE PIC24FJ128GB204 FAMILY

Device	Configuration Word Addresses						
	1	2	3	4			
PIC24FJ64GB2XX	ABFEh	ABFCh	ABFAh	ABF8h			
PIC24FJ128GB2XX	157FEh	157FCh	157FAh	157F8h			

REGISTER 30-1: CW1: FLASH CONFIGURATION WORD 1

U-1	U-1	U-1	U-1	U-1	U-1	U-1	U-1
_	_	_	_	_	_	_	_
bit 23							bit 16

r-x	R/PO-1						
_	JTAGEN	GCP	GWRP	DEBUG	LPCFG	ICS1	ICS0
bit 15							bit 8

R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1
FWDTEN1	FWDTEN0	WINDIS	FWPSA	WDTPS3	WDTPS2	WDTPS1	WDTPS0
bit 7							bit 0

Legend:	r = Reserved bit	PO = Program Once bit	t
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 23-16 **Unimplemented:** Read as '1'

bit 15 **Reserved:** The value is unknown; program as '0'

bit 14 JTAGEN: JTAG Port Enable bit

1 = JTAG port is enabled0 = JTAG port is disabled

bit 13 GCP: General Segment Program Memory Code Protection bit

1 = Code protection is disabled

0 = Code protection is enabled for the entire program memory space

bit 12 **GWRP:** General Segment Code Flash Write Protection bit

1 = Writes to program memory are allowed

0 = Writes to program memory are not allowed

bit 11 **DEBUG:** Background Debugger Enable bit

1 = Device resets into Operational mode

0 = Device resets into Debug mode

bit 10 LPCFG: Low-Voltage/Retention Regulator Configuration bit

1 = Low-voltage/retention regulator is always disabled

0 = Low-power, low-voltage/retention regulator is enabled and controlled in firmware by the RETEN bit

bit 9-8 ICS<1:0>: Emulator Pin Placement Select bits

11 = Emulator functions are shared with PGEC1/PGED1

10 = Emulator functions are shared with PGEC2/PGED2

01 = Emulator functions are shared with PGEC3/PGED3

00 = Reserved; do not use

bit 7-6 **FWDTEN<1:0>:** Watchdog Timer Configuration bits

11 = WDT is always enabled; SWDTEN bit has no effect

10 = WDT is enabled and controlled in firmware by the SWDTEN bit

01 = WDT is enabled only in Run mode and disabled in Sleep modes; SWDTEN bit is disabled

00 = WDT is disabled; SWDTEN bit is disabled

bit 5 WINDIS: Windowed Watchdog Timer Disable bit

1 = Standard Watchdog Timer is enabled

0 = Windowed Watchdog Timer is enabled (FWDTEN<1:0> must not be '00')

REGISTER 30-1: CW1: FLASH CONFIGURATION WORD 1 (CONTINUED)

bit 4 **FWPSA:** WDT Prescaler Ratio Select bit

1 = Prescaler ratio of 1:1280 = Prescaler ratio of 1:32

bit 3-0 **WDTPS<3:0>:** Watchdog Timer Postscaler Select bits

1111 = 1:32,768 1110 = 1:16,384 1101 = 1:8,192 1100 = 1:4,096 1011 = 1:2,048 1010 = 1:1,024 1001 = 1:512 1000 = 1:256 0111 = 1:128 0110 = 1:64

0011 = 1:8 0010 = 1:4 0001 = 1:2 0000 = 1:1

0101 = 1:32 0100 = 1:16

REGISTER 30-2: CW2: FLASH CONFIGURATION WORD 2

U-1	U-1	U-1	U-1	U-1	U-1	U-1	U-1
_	_	_	_	_	_	_	_
bit 23							bit 16

R/PO-1	r-0	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1
IESO	_	WDTCMX	ALTCMPI	ALTRB6 ⁽²⁾	FNOSC2	FNOSC1	FNOSC0
bit 15							bit 8

R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	r-1	R/PO-1	R/PO-1
FCKSM1	FCKSM0	OSCIOFCN	WDTCLK1	WDTCLK0	_	POSCMD1	POSCMD0
bit 7							bit 0

Legend:	r = Reserved bit	PO = Program Once bit		
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 23-16 Unimplemented: Read as '1'

bit 15 **IESO:** Internal External Switchover bit

1 = IESO mode (Two-Speed Start-up) is enabled 0 = IESO mode (Two-Speed Start-up) is disabled

bit 14 Reserved: Read as '0'

bit 13 WDTCMX: WDT Clock Multiplex Control bit

1 = WDT clock source is determined by the WDTCLK<1:0> Configuration bits

0 = WDT always uses LPRC as its clock source

bit 12 ALTCMPI: Alternate Comparator Input bit

1 = C1INC is on RB13, C2INC is on RB9 and C3INC is on RA0

0 = C1INC, C2INC and C3INC are on RB9

bit 11 ALTRB6: Alternate RB6 Pin Function Enable bit⁽²⁾

1 = Appends the RP6/ASCL1/PMD6 functions of RB6 to RA1 pin functions

0 = Keeps the RP6/ASCL1/PMD6 functions to RB6

bit 10-8 FNOSC<2:0>: Initial Oscillator Select bits

111 = Fast RC Oscillator with Postscaler (FRCDIV)

110 = Reserved

101 = Low-Power RC Oscillator (LPRC)

100 = Secondary Oscillator (SOSC)

011 = Primary Oscillator with PLL module (XTPLL, HSPLL, ECPLL)

010 = Primary Oscillator (XT, HS, EC)

001 = Fast RC Oscillator with Postscaler and PLL module (FRCPLL)

000 = Fast RC Oscillator (FRC)

bit 7-6 FCKSM<1:0>: Clock Switching and Fail-Safe Clock Monitor Configuration bits

1x = Clock switching and Fail-Safe Clock Monitor are disabled

01 = Clock switching is enabled, Fail-Safe Clock Monitor is disabled

00 = Clock switching is enabled, Fail-Safe Clock Monitor is enabled

Note 1: The 31 kHz FRC source is used when a Windowed WDT mode is selected and the LPRC is not being used as the system clock. The LPRC is used when the device is in Sleep mode and in all other cases.

2: When VBUS functionality is used, this Configuration bit must be programmed to '1'.

REGISTER 30-2: CW2: FLASH CONFIGURATION WORD 2 (CONTINUED)

bit 5 OSCIOFCN: OSCO Pin Configuration bit

If POSCMD < 1:0 > = 11 or 00:

1 = OSCO/CLKO/RA3 functions as CLKO (Fosc/2) 0 = OSCO/CLKO/RA3 functions as port I/O (RA3)

If POSCMD<1:0> = 10 or 01:

OSCIOFCN has no effect on OSCO/CLKO/RA3.

bit 4-3 WDTCLK<1:0>: WDT Clock Source Select bits

When WDTCMX = 1:

11 = LPRC

10 = Either the 31 kHz FRC source or LPRC, depending on device configuration (1)

01 = SOSC input

00 = System clock when active, LPRC while in Sleep mode

When WDTCMX = 0:

LPRC is always the WDT clock source.

bit 2 Reserved: Configure as '1'

bit 1-0 **POSCMD<1:0>:** Primary Oscillator Configuration bits

11 = Primary Oscillator mode is disabled 10 = HS Oscillator mode is selected 01 = XT Oscillator mode is selected 00 = EC Oscillator mode is selected

Note 1: The 31 kHz FRC source is used when a Windowed WDT mode is selected and the LPRC is not being used as the system clock. The LPRC is used when the device is in Sleep mode and in all other cases.

2: When VBUS functionality is used, this Configuration bit must be programmed to '1'.

REGISTER 30-3: CW3: FLASH CONFIGURATION WORD 3

U-1	U-1	U-1	U-1	U-1	U-1	U-1	U-1
_	_	_	_	_	_	_	_
bit 23							bit 16

R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1
WPEND	WPCFG	WPDIS	BOREN	PLLSS ⁽⁴⁾	WDTWIN1	WDTWIN0	SOSCSEL
bit 15							bit 8

r-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1
_	WPFP6 ⁽³⁾	WPFP5	WPFP4	WPFP3	WPFP2	WPFP1	WPFP0
bit 7							bit 0

Legend:	r = Reserved bit	PO = Program Once bit	
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 23-16 Unimplemented: Read as '1'
- bit 15 **WPEND:** Segment Write Protection End Page Select bit
 - 1 = Protected program memory segment upper boundary is at the last page of program memory; the lower boundary is the code page specified by WPFP<6:0>
 - 0 = Protected program memory segment lower boundary is at the bottom of the program memory (000000h); upper boundary is the code page specified by WPFP<6:0>
- bit 14 WPCFG: Configuration Word Code Page Write Protection Select bit
 - 1 = Last page (at the top of program memory) and Flash Configuration Words are not write-protected (1)
 - 0 = Last page and Flash Configuration Words are write-protected provided WPDIS = 0
- bit 13 WPDIS: Segment Write Protection Disable bit
 - 1 = Segmented program memory write protection is disabled
 - 0 = Segmented program memory write protection is enabled; protected segment is defined by the WPEND, WPCFG and WPFPx Configuration bits
- bit 12 **BOREN:** Brown-out Reset Enable bit
 - 1 = BOR is enabled (all modes except Deep Sleep)
 - 0 = BOR is disabled
- bit 11 PLLSS: PLL Secondary Selection Configuration bit (4)
 - 1 = PLL is fed by the Primary Oscillator
 - 0 = PLL is fed by the on-chip Fast RC (FRC) Oscillator
- bit 10-9 WDTWIN<1:0>: Watchdog Timer Window Width Select bits
 - 11 **= 25%**
 - 10 = 37.5%
 - 01 = 50%
 - 00 = 75%
- bit 8 SOSCSEL: SOSC Selection bit
 - 1 = SOSC circuit is selected
 - 0 = Digital (SCLKI) mode⁽²⁾
- **Note 1:** Regardless of WPCFG status, if WPEND = 1 or if the WPFP<6:0> bits correspond to the Configuration Word page, the Configuration Word page is protected.
 - 2: Ensure that the SCLKI pin is made a digital input while using this configuration (see Table 11-1).
 - 3: For the 64K devices (PIC24FJ64GB2XX), maintain WPFP6 as '0'.
 - 4: This Configuration bit only takes effect when PLL is not being used.

REGISTER 30-3: CW3: FLASH CONFIGURATION WORD 3 (CONTINUED)

bit 7 **Reserved:** Always maintain as '1'

bit 6-0 **WPFP<6:0>:** Write-Protected Code Segment Boundary Page bits⁽³⁾

Designates the 512 instruction words page boundary of the protected Code Segment.

If WPEND = 1:

Specifies the lower page boundary of the protected Code Segment; the last page being the last implemented page in the device.

If WPEND = 0:

Specifies the upper page boundary of the protected Code Segment; Page 0 being the lower boundary.

- **Note 1:** Regardless of WPCFG status, if WPEND = 1 or if the WPFP<6:0> bits correspond to the Configuration Word page, the Configuration Word page is protected.
 - 2: Ensure that the SCLKI pin is made a digital input while using this configuration (see Table 11-1).
 - 3: For the 64K devices (PIC24FJ64GB2XX), maintain WPFP6 as '0'.
 - 4: This Configuration bit only takes effect when PLL is not being used.

REGISTER 30-4: CW4: FLASH CONFIGURATION WORD 4

U-1	U-1	U-1	U-1	U-1	U-1	U-1	U-1
_	_	_	_	_	_	_	_
bit 23 bit 16							

R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	r-1	R/PO-1
IOL1WAY	I2C1SEL	PLLDIV3	PLLDIV2	PLLDIV1	PLLDIV0	_	DSSWEN
bit 15							bit 8

R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1
DSWDTEN	DSBOREN	DSWDTOSC	DSWDTPS4	DSWDTPS3	DSWDTPS2	DSWDTPS1	DSWDTPS0
bit 7							bit 0

Legend:	r = Reserved bit	PO = Program Once bit	
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 23-16 Unimplemented: Read as '1'

bit 15 **IOL1WAY:** IOLOCK One-Way Set Enable bit

1 = The IOLOCK bit (OSCCON<6>) can be set once, provided the unlock sequence has been completed; once set, the Peripheral Pin Select registers cannot be written to a second time

0 = The IOLOCK bit can be set and cleared as needed, provided the unlock sequence has been completed

bit 14 I2C1SEL: Alternate I2C1 Location Select bit

1 = I2C1 uses the SCL1 and SDA1 pins

0 = I2C1 uses the ASCL1 and ASDA1 pins

bit 13-10 PLLDIV<3:0>: USB 96 MHz PLL Prescaler Select bits

1111 = PLL is disabled

1110 = 8x PLL is selected

1101 = 6x PLL is selected

1100 = 4x PLL is selected

1011

.... = Reserved, do not use

1000

0111 = Oscillator input divided by 12 (48 MHz input)

0110 = Oscillator input divided by 8 (32 MHz input)

0101 = Oscillator input divided by 6 (24 MHz input)

0100 = Oscillator input divided by 5 (20 MHz input)

0011 = Oscillator input divided by 4 (16 MHz input)

0010 = Oscillator input divided by 3 (12 MHz input)

0001 = Oscillator input divided by 2 (8 MHz input)

0000 = Oscillator input used directly (4 MHz input)

bit 9 **Reserved:** Always maintain as '1'

bit 8 DSSWEN: Deep Sleep Software Control Select bit

1 = Deep Sleep operation is enabled and controlled by the DSEN bit

0 = Deep Sleep operation is disabled

bit 7 DSWDTEN: Deep Sleep Watchdog Timer Enable bit

1 = Deep Sleep WDT is enabled

0 = Deep Sleep WDT is disabled

REGISTER 30-4: CW4: FLASH CONFIGURATION WORD 4 (CONTINUED)

```
bit 6
             DSBOREN: Deep Sleep Brown-out Reset Enable bit
             1 = BOR is enabled in Deep Sleep mode
             0 = BOR is disabled in Deep Sleep mode (remains active in other Sleep modes)
bit 5
             DSWDTOSC: Deep Sleep Watchdog Timer Clock Select bit
             1 = Clock source is LPRC
             0 = Clock source is SOSC
bit 4-0
             DSWDTPS<4:0>: Deep Sleep Watchdog Timer Postscaler Select bits
             11111 = 1:68,719,476,736 (25.7 days)
             11110 = 1:34,359,738,368(12.8 days)
             11101 = 1:17,179,869,184 (6.4 days)
             11100 = 1:8,589,934592 (77.0 hours)
             11011 = 1:4,294,967,296 (38.5 hours)
             11010 = 1:2,147,483,648 (19.2 hours)
             11001 = 1:1,073,741,824 (9.6 hours)
             11000 = 1:536,870,912 (4.8 hours)
             10111 = 1:268,435,456 (2.4 hours)
             10110 = 1:134,217,728 (72.2 minutes)
             10101 = 1:67,108,864 (36.1 minutes)
             10100 = 1:33,554,432 (18.0 minutes)
             10011 = 1:16,777,216 (9.0 minutes)
             10010 = 1:8,388,608 (4.5 minutes)
             10001 = 1:4,194,304 (135.3s)
             10000 = 1:2,097,152 (67.7s)
             01111 = 1:1,048,576 (33.825s)
             01110 = 1:524,288 (16.912s)
             01101 = 1:262,114 (8.456s)
             01100 = 1:131,072 (4.228s)
             01011 = 1:65,536 (2.114s)
             01010 = 1:32,768 (1.057s)
             01001 = 1:16,384 (528.5 ms)
             01000 = 1:8,192 (264.3 ms)
             00111 = 1:4,096 (132.1 ms)
             00110 = 1:2,048 (66.1 ms)
             00101 = 1:1,024 (33 ms)
             00100 = 1:512 (16.5 ms)
             00011 = 1:256 (8.3 ms)
             00010 = 1:128 (4.1 ms)
             00001 = 1:64 (2.1 ms)
             00000 = 1:32 (1 ms)
```

REGISTER 30-5: DEVID: DEVICE ID REGISTER

U-1	U-1	U-1	U-1	U-1	U-1	U-1	U-1
_	_	_	_	_	_	_	_
bit 23							bit 16

R	R	R	R	R	R	R	R
FAMID7	FAMID6	FAMID5	FAMID4	FAMID3	FAMID2	FAMID1	FAMID0
bit 15							bit 8

R	R	R	R	R	R	R	R
DEV7	DEV6	DEV5	DEV4	DEV3	DEV2	DEV1	DEV0
bit 7							bit 0

Legend: R = Readable bit U = Unimplemented bit, read as '1'

bit 23-16 Unimplemented: Read as '1'

bit 15-8 **FAMID<7:0>:** Device Family Identifier bits

0100 1100 = PIC24FJ128GB204 family

bit 7-0 **DEV<7:0>:** Individual Device Identifier bits

0101 1000 = PIC24FJ64GB202 0101 1010 = PIC24FJ128GB202 0101 1001 = PIC24FJ64GB204 0101 1011 = PIC24FJ128GB204

REGISTER 30-6: DEVREV: DEVICE REVISION REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0			
_	_	_	_	_	_	1	_			
bit 23	bit 23 bit 16									

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	
_	_	_	_	_	_	_	_	
bit 15 bit								

U-0	U-0	U-0	U-0	R	R	R	R
_	_	_	_	REV3	REV2	REV1	REV0
bit 7							bit 0

Legend: R = Readable bit U = Unimplemented bit, read as '1'

bit 23-4 **Unimplemented:** Read as '1'

bit 3-0 **REV<3:0>:** Device Revision Identifier bits

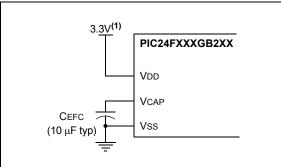
30.2 On-Chip Voltage Regulator

All PIC24FJ128GB204 family devices power their core digital logic at a nominal 1.8V. This may create an issue for designs that are required to operate at a higher typical voltage, such as 3.3V. To simplify system design, all devices in the PIC24FJ128GB204 family incorporate an on-chip regulator that allows the device to run its core logic from VDD.

This regulator is always enabled. It provides a constant voltage (1.8V nominal) to the digital core logic, from a VDD of about 2.1V, all the way up to the device's VDDMAX. It does not have the capability to boost VDD levels. In order to prevent "brown-out" conditions when the voltage drops too low for the regulator, the Brown-out Reset occurs. Then, the regulator output follows VDD with a typical voltage drop of 300 mV.

A low-ESR capacitor (such as ceramic) must be connected to the VCAP pin (Figure 30-1). This helps to maintain the stability of the regulator. The recommended value for the filter capacitor (CEFC) is provided in Section 33.1 "DC Characteristics".

FIGURE 30-1: CONNECTIONS FOR THE ON-CHIP REGULATOR



Note 1: This is a typical operating voltage. Refer to Section 33.0 "Electrical Characteristics" for the full operating ranges of VDD.

30.2.1 ON-CHIP REGULATOR AND POR

The voltage regulator requires a small amount of time to transition from a disabled or standby state into normal operating mode. During this time, designated as TVREG, code execution is disabled. TVREG is applied every time the device resumes operation after any power-down, including Sleep mode. TVREG is determined by the status of the VREGS bit (RCON<8>). Refer to Section 33.0 "Electrical Characteristics" for more information on TVREG.

Note: For more information, see Section 33.0 "Electrical Characteristics". The information in this data sheet supersedes the information in the "dsPIC33/PIC24 Family Reference Manual".

30.2.2 VOLTAGE REGULATOR STANDBY MODE

The on-chip regulator always consumes a small incremental amount of current over IDD/IPD, including when the device is in Sleep mode, even though the core digital logic does not require power. To provide additional savings in applications where power resources are critical, the regulator can be made to enter Standby mode on its own, whenever the device goes into Sleep mode. This feature is controlled by the VREGS bit (RCON<8>). Clearing the VREGS bit enables the Standby mode. When waking up from Standby mode, the regulator needs to wait for TVREG to expire before wake-up.

30.2.3 LOW-VOLTAGE/RETENTION REGULATOR

When a power-saving mode, such as Sleep is used, PIC24FJ128GB204 family devices may use a separate low-power, low-voltage/retention regulator to power critical circuits. This regulator, which operates at 1.2V nominal, maintains power to data RAM and the RTCC while all other core digital logic is powered down. It operates only in Sleep and VBAT modes.

The low-voltage/retention regulator is described in more detail in Section 10.1.3 "Low-Voltage/Retention Regulator".

30.3 Watchdog Timer (WDT)

For PIC24FJ128GB204 family devices, the WDT is driven by the LPRC Oscillator. When the WDT is enabled, the clock source is also enabled.

The nominal WDT clock source from LPRC is 31 kHz. This feeds a prescaler that can be configured for either 5-bit (divide-by-32) or 7-bit (divide-by-128) operation. The prescaler is set by the FWPSA Configuration bit. With a 31 kHz input, the prescaler yields a nominal WDT Time-out (TWDT) period of 1 ms in 5-bit mode or 4 ms in 7-bit mode.

A variable postscaler divides down the WDT prescaler output and allows for a wide range of time-out periods. The postscaler is controlled by the WDTPS<3:0> Configuration bits (CW1<3:0>), which allows the selection of a total of 16 settings, from 1:1 to 1:32,768. Using the prescaler and postscaler time-out periods, ranges from 1 ms to 131 seconds can be achieved.

The WDT, prescaler and postscaler are reset:

- · On any device Reset
- On the completion of a clock switch, whether invoked by software (i.e., setting the OSWEN bit after changing the NOSCx bits) or by hardware (i.e., Fail-Safe Clock Monitor)
- When a PWRSAV instruction is executed (i.e., Sleep or Idle mode is entered)
- When the device exits Sleep or Idle mode to resume normal operation
- By a CLRWDT instruction during normal execution

If the WDT is enabled, it will continue to run during Sleep or Idle modes. When the WDT time-out occurs, the device will wake the device and code execution will continue from where the PWRSAV instruction was executed. The corresponding SLEEP or IDLE (RCON<3:2>) bits will need to be cleared in software after the device wakes up.

The WDT Flag bit, WDTO (RCON<4>), is not automatically cleared following a WDT time-out. To detect subsequent WDT events, the flag must be cleared in software.

Note: The CLRWDT and PWRSAV instructions clear the prescaler and postscaler counts when executed.

30.3.1 WINDOWED OPERATION

The Watchdog Timer has an optional Fixed Window mode of operation. In this Windowed mode, CLRWDT instructions can only reset the WDT during the window width, 25%, 37.5%, 50% or 75% of the programmed WDT period, controlled by the WDTWIN<1:0> Configuration bits (CW3<10:9>). A CLRWDT instruction executed before that window causes a WDT Reset, similar to a WDT time-out.

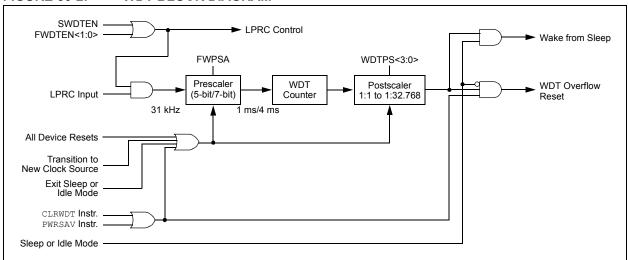
Windowed WDT mode is enabled by programming the WINDIS Configuration bit (CW1<5>) to '0'.

30.3.2 CONTROL REGISTER

The WDT is enabled or disabled by the FWDTEN<1:0> Configuration bits. When the Configuration bits, FWDTEN<1:0> = 11, the WDT is always enabled.

The WDT can be optionally controlled in software when the Configuration bits, FWDTEN<1:0> = 10. When FWDTEN<1:0> = 00, the Watchdog Timer is always disabled. The WDT is enabled in software by setting the SWDTEN control bit (RCON<5>). The SWDTEN control bit is cleared on any device Reset. The software WDT option allows the user to enable the WDT for critical code segments and disable the WDT during non-critical segments for maximum power savings.





30.4 Program Verification and Code Protection

PIC24FJ128GB204 family devices provide two complimentary methods to protect application code from overwrites and erasures. These also help to protect the device from inadvertent configuration changes during run time.

30.4.1 GENERAL SEGMENT PROTECTION

For all devices in the PIC24FJ128GB204 family, the on-chip program memory space is treated as a single block, known as the General Segment (GS). Code protection for this block is controlled by one Configuration bit, GCP. This bit inhibits external reads and writes to the program memory space. It has no direct effect in normal execution mode.

Write protection is controlled by the GWRP bit in Configuration Word 1. When GWRP is programmed to '0', internal write and erase operations to program memory are blocked.

30.4.2 CODE SEGMENT PROTECTION

In addition to global General Segment protection, a separate subrange of the program memory space can be individually protected against writes and erases. This area can be used for many purposes where a separate block of write and erase-protected code is needed, such as bootloader applications. Unlike common boot block implementations, the specially protected segment in the PIC24FJ128GB204 family devices can be located by the user anywhere in the program space and configured in a wide range of sizes.

Code Segment (CS) protection provides an added level of protection to a designated area of program memory by disabling the NVM safety interlock whenever a write or erase address falls within a specified range. It does not override General Segment protection controlled by the GCP or GWRP bits. For example, if GCP and GWRP are enabled, enabling segmented code protection for the bottom half of program memory does not undo General Segment protection for the top half.

The size and type of protection for the segmented code range are configured by the WPFPx, WPEND, WPCFG and WPDIS bits in Configuration Word 3. Code Segment protection is enabled by programming the WPDIS bit (= 0). The WPFPx bits specify the size of the segment to be protected, by specifying the 512-word code page that is the start or end of the protected segment. The specified region is inclusive, therefore, this page will also be protected.

The WPEND bit determines if the protected segment uses the top or bottom of the program space as a boundary. Programming WPEND (= 0) sets the bottom of program memory (000000h) as the lower boundary of the protected segment. Leaving WPEND unprogrammed (= 1) protects the specified page through the last page of implemented program memory, including the Configuration Word locations.

A separate bit, WPCFG, is used to protect the last page of program space, including the Flash Configuration Words. Programming WPCFG (= 0) protects the last page in addition to the pages selected by the WPEND and WPFP<6:0> bits setting. This is useful in circumstances where write protection is needed for both the Code Segment in the bottom of the memory and the Flash Configuration Words.

The various options for segment code protection are shown in Table 30-2.

TABLE 30-2: CODE SEGMENT PROTECTION CONFIGURATION OPTIONS

Segmen	t Configura	tion Bits	Write/Errose Protection of Code Segment
WPDIS	WPEND	WPCFG	Write/Erase Protection of Code Segment
1	Х	Х	No additional protection is enabled; all program memory protection is configured by GCP and GWRP.
0	1	Х	Addresses from the first address of the code page are defined by WPFP<6:0> through the end of implemented program memory (inclusive); erase/write-protected, including Flash Configuration Words.
0	0	1	Address, 000000h through the last address of the code page, is defined by WPFP<6:0> (inclusive); erase/write-protected.
0	0	0	Address, 000000h through the last address of the code page, is defined by WPFP<6:0> (inclusive); erase/write-protected and the last page, including the Flash Configuration Words, are erase/write-protected.

30.4.3 CONFIGURATION REGISTER PROTECTION

The Configuration registers are protected against inadvertent or unwanted changes or reads in two ways. The primary protection method is the same as that of the RP registers – shadow registers contain a complimentary value which is constantly compared with the actual value.

To safeguard against unpredictable events, Configuration bit changes, resulting from individual cell-level disruptions (such as ESD events), will cause a parity error and trigger a device Reset.

The data for the Configuration registers is derived from the Flash Configuration Words in program memory. When the GCP bit is set, the source data for device configuration is also protected as a consequence. Even if General Segment protection is not enabled, the device configuration can be protected by using the appropriate Code Segment protection setting.

30.5 JTAG Interface

PIC24FJ128GB204 family devices implement a JTAG interface, which supports boundary scan device testing and programming.

30.6 In-Circuit Serial Programming

PIC24FJ128GB204 family microcontrollers can be serially programmed while in the end application circuit. This is simply done with two lines for clock (PGECx) and data (PGEDx), and three other lines for power (VDD), ground (VSS) and MCLR. This allows customers to manufacture boards with unprogrammed devices and then program the microcontroller just before shipping the product. This also allows the most recent firmware or a custom firmware to be programmed.

30.7 In-Circuit Debugger

When MPLAB® ICD 3 is selected as a debugger, the in-circuit debugging functionality is enabled. This function allows simple debugging functions when used with MPLAB X IDE. Debugging functionality is controlled through the PGECx (Emulation/Debug Clock) and PGEDx (Emulation/Debug Data) pins.

To use the in-circuit debugger function of the device, the design must implement ICSP connections to MCLR, VDD, Vss and the PGECx/PGEDx pin pair, designated by the ICSx Configuration bits. In addition, when the feature is enabled, some of the resources are not available for general use. These resources include the first 80 bytes of data RAM and two I/O pins.

31.0 DEVELOPMENT SUPPORT

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

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

31.1 MPLAB X Integrated Development Environment Software

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

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

Feature-Rich Editor:

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

User-Friendly, Customizable Interface:

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

Project-Based Workspaces:

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

File History and Bug Tracking:

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

31.2 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

31.3 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

31.4 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

31.5 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

31.6 MPLAB X SIM Software Simulator

The MPLAB X SIM Software Simulator allows code development in a PC-hosted environment by simulating the PIC MCUs and dsPIC DSCs on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a comprehensive stimulus controller. Registers can be logged to files for further run-time analysis. The trace buffer and logic analyzer display extend the power of the simulator to record and track program execution, actions on I/O, most peripherals and internal registers.

The MPLAB X SIM Software Simulator fully supports symbolic debugging using the MPLAB XC Compilers, and the MPASM and MPLAB Assemblers. The software simulator offers the flexibility to develop and debug code outside of the hardware laboratory environment, making it an excellent, economical software development tool.

31.7 MPLAB REAL ICE In-Circuit Emulator System

The MPLAB REAL ICE In-Circuit Emulator System is Microchip's next generation high-speed emulator for Microchip Flash DSC and MCU devices. It debugs and programs all 8, 16 and 32-bit MCU, and DSC devices with the easy-to-use, powerful graphical user interface of the MPLAB X IDE.

The emulator is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with either a connector compatible with in-circuit debugger systems (RJ-11) or with the new high-speed, noise tolerant, Low-Voltage Differential Signal (LVDS) interconnection (CAT5).

The emulator is field upgradable through future firmware downloads in MPLAB X IDE. MPLAB REAL ICE offers significant advantages over competitive emulators including full-speed emulation, run-time variable watches, trace analysis, complex breakpoints, logic probes, a ruggedized probe interface and long (up to three meters) interconnection cables.

31.8 MPLAB ICD 3 In-Circuit Debugger System

The MPLAB ICD 3 In-Circuit Debugger System is Microchip's most cost-effective, high-speed hardware debugger/programmer for Microchip Flash DSC and MCU devices. It debugs and programs PIC Flash microcontrollers and dsPIC DSCs with the powerful, yet easy-to-use graphical user interface of the MPLAB X IDE.

The MPLAB ICD 3 In-Circuit Debugger probe is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with a connector compatible with the MPLAB ICD 2 or MPLAB REAL ICE systems (RJ-11). MPLAB ICD 3 supports all MPLAB ICD 2 headers.

31.9 PICkit 3 In-Circuit Debugger/ Programmer

The MPLAB PICkit 3 allows debugging and programming of PIC and dsPIC Flash microcontrollers at a most affordable price point using the powerful graphical user interface of the MPLAB X IDE. The MPLAB PICkit 3 is connected to the design engineer's PC using a full-speed USB interface and can be connected to the target via a Microchip debug (RJ-11) connector (compatible with MPLAB ICD 3 and MPLAB REAL ICE). The connector uses two device I/O pins and the Reset line to implement in-circuit debugging and In-Circuit Serial Programming™ (ICSP™).

31.10 MPLAB PM3 Device Programmer

The MPLAB PM3 Device Programmer is a universal, CE compliant device programmer with programmable voltage verification at VDDMIN and VDDMAX for maximum reliability. It features a large LCD display (128 x 64) for menus and error messages, and a modular, detachable socket assembly to support various package types. The ICSP cable assembly is included as a standard item. In Stand-Alone mode, the MPLAB PM3 Device Programmer can read, verify and program PIC devices without a PC connection. It can also set code protection in this mode. The MPLAB PM3 connects to the host PC via an RS-232 or USB cable. The MPLAB PM3 has high-speed communications and optimized algorithms for quick programming of large memory devices, and incorporates an MMC card for file storage and data applications.

31.11 Demonstration/Development Boards, Evaluation Kits, and Starter Kits

A wide variety of demonstration, development and evaluation boards for various PIC MCUs and dsPIC DSCs allows quick application development on fully functional systems. Most boards include prototyping areas for adding custom circuitry and provide application firmware and source code for examination and modification.

The boards support a variety of features, including LEDs, temperature sensors, switches, speakers, RS-232 interfaces, LCD displays, potentiometers and additional EEPROM memory.

The demonstration and development boards can be used in teaching environments, for prototyping custom circuits and for learning about various microcontroller applications.

In addition to the PICDEM™ and dsPICDEM™ demonstration/development board series of circuits, Microchip has a line of evaluation kits and demonstration software for analog filter design, KEELoq® security ICs, CAN, IrDA®, PowerSmart battery management, SEEVAL® evaluation system, Sigma-Delta ADC, flow rate sensing, plus many more.

Also available are starter kits that contain everything needed to experience the specified device. This usually includes a single application and debug capability, all on one board.

Check the Microchip web page (www.microchip.com) for the complete list of demonstration, development and evaluation kits.

31.12 Third-Party Development Tools

Microchip also offers a great collection of tools from third-party vendors. These tools are carefully selected to offer good value and unique functionality.

- Device Programmers and Gang Programmers from companies, such as SoftLog and CCS
- Software Tools from companies, such as Gimpel and Trace Systems
- Protocol Analyzers from companies, such as Saleae and Total Phase
- Demonstration Boards from companies, such as MikroElektronika, Digilent[®] and Olimex
- Embedded Ethernet Solutions from companies, such as EZ Web Lynx, WIZnet and IPLogika[®]

32.0 INSTRUCTION SET SUMMARY

Note:

This chapter is a brief summary of the PIC24F Instruction Set Architecture (ISA) and is not intended to be a comprehensive reference source.

The PIC24F instruction set adds many enhancements to the previous PIC® MCU instruction sets, while maintaining an easy migration from previous PIC MCU instruction sets. Most instructions are a single program memory word. Only three instructions require two program memory locations.

Each single-word instruction is a 24-bit word divided into an 8-bit opcode, which specifies the instruction type and one or more operands, which further specify the operation of the instruction. The instruction set is highly orthogonal and is grouped into four basic categories:

- · Word or byte-oriented operations
- · Bit-oriented operations
- · Literal operations
- · Control operations

Table 32-1 shows the general symbols used in describing the instructions. The PIC24F instruction set summary in Table 32-2 lists all the instructions, along with the status flags affected by each instruction.

Most word or byte-oriented W register instructions (including barrel shift instructions) have three operands:

- The first source operand, which is typically a register, 'Wb', without any address modifier
- The second source operand, which is typically a register, 'Ws', with or without an address modifier
- The destination of the result, which is typically a register, 'Wd', with or without an address modifier

However, word or byte-oriented file register instructions have two operands:

- · The file register specified by the value, 'f'
- The destination, which could either be the file register, 'f', or the W0 register, which is denoted as 'WREG'

Most bit-oriented instructions (including simple rotate/shift instructions) have two operands:

- The W register (with or without an address modifier) or file register (specified by the value of 'Ws' or 'f')
- The bit in the W register or file register (specified by a literal value or indirectly by the contents of register, 'Wb')

The literal instructions that involve data movement may use some of the following operands:

- A literal value to be loaded into a W register or file register (specified by the value of 'k')
- The W register or file register where the literal value is to be loaded (specified by 'Wb' or 'f')

However, literal instructions that involve arithmetic or logical operations use some of the following operands:

- The first source operand, which is a register, 'Wb', without any address modifier
- The second source operand, which is a literal value
- The destination of the result (only if not the same as the first source operand), which is typically a register, 'Wd', with or without an address modifier

The control instructions may use some of the following operands:

- · A program memory address
- The mode of the Table Read and Table Write instructions

All instructions are a single word, except for certain double-word instructions, which were made double-word instructions so that all the required information is available in these 48 bits. In the second word, the 8 MSbs are '0's. If this second word is executed as an instruction (by itself), it will execute as a NOP.

Most single-word instructions are executed in a single instruction cycle, unless a conditional test is true or the Program Counter is changed as a result of the instruction. In these cases, the execution takes two instruction cycles, with the additional instruction cycle(s) executed as a NOP. Notable exceptions are the BRA (unconditional/computed branch), indirect CALL/GOTO, all Table Reads and Table Writes, and RETURN/RETFIE instructions, which are single-word instructions but take two or three cycles.

Certain instructions that involve skipping over the subsequent instruction require either two or three cycles if the skip is performed, depending on whether the instruction being skipped is a single-word or two-word instruction. Moreover, double-word moves require two cycles. The double-word instructions execute in two instruction cycles.

TABLE 32-1: SYMBOLS USED IN OPCODE DESCRIPTIONS

#tout	
#text	Means literal defined by "text"
(text)	Means "content of text"
<text]< td=""><td>Means "the location addressed by text"</td></text]<>	Means "the location addressed by text"
{ }	Optional field or operation
<n:m></n:m>	Register bit field
.b	Byte mode selection
.d	Double-Word mode selection
.S	Shadow register select
.w	Word mode selection (default)
bit4	4-bit Bit Selection field (used in word addressed instructions) ∈ {015}
C, DC, N, OV, Z	MCU Status bits: Carry, Digit Carry, Negative, Overflow, Sticky Zero
Expr	Absolute address, label or expression (resolved by the linker)
f	File register address ∈ {0000h1FFFh}
lit1	1-bit unsigned literal ∈ {0,1}
lit4	4-bit unsigned literal ∈ {015}
lit5	5-bit unsigned literal ∈ {031}
lit8	8-bit unsigned literal ∈ {0255}
lit10	10-bit unsigned literal ∈ {0255} for Byte mode, {0:1023} for Word mode
lit14	14-bit unsigned literal ∈ {016383}
lit16	16-bit unsigned literal ∈ {065535}
lit23	23-bit unsigned literal ∈ {08388607}; LSB must be '0'
None	Field does not require an entry, may be blank
PC	Program Counter
Slit10	10-bit signed literal ∈ {-512511}
Slit16	16-bit signed literal ∈ {-3276832767}
Slit6	6-bit signed literal ∈ {-1616}
Wb	Base W register ∈ {W0W15}
Wd	$Destination\ W\ register\ \in \{\ Wd,\ [Wd],\ [Wd++],\ [Wd],\ [++Wd],\ [Wd]\ \}$
Wdo	Destination W register ∈ { Wnd, [Wnd], [Wnd++], [Wnd], [++Wnd], [Wnd], [Wnd+Wb] }
Wm,Wn	Dividend, Divisor Working register pair (direct addressing)
Wn	One of 16 Working registers ∈ {W0W15}
Wnd	One of 16 Destination Working registers ∈ {W0W15}
Wns	One of 16 Source Working registers ∈ {W0W15}
WREG	W0 (Working register used in file register instructions)
Ws	Source W register ∈ { Ws, [Ws], [Ws++], [Ws], [++Ws], [Ws] }
Wso	Source W register ∈ { Wns, [Wns], [Wns++], [Wns], [++Wns], [Wns], [Wns+Wb] }

TABLE 32-2: INSTRUCTION SET OVERVIEW

Assembly Mnemonic		Assembly Syntax	Description	# of Words	# of Cycles	Status Flags Affected
ADD	ADD	f	f = f + WREG	1	1	C, DC, N, OV, Z
	ADD	f,WREG	WREG = f + WREG	1	1	C, DC, N, OV, Z
	ADD	#lit10,Wn	Wd = lit10 + Wd	1	1	C, DC, N, OV, Z
	ADD	Wb, Ws, Wd	Wd = Wb + Ws	1	1	C, DC, N, OV, Z
	ADD	Wb,#lit5,Wd	Wd = Wb + lit5	1	1	C, DC, N, OV, Z
ADDC	ADDC	f	f = f + WREG + (C)	1	1	C, DC, N, OV, Z
	ADDC	f,WREG	WREG = f + WREG + (C)	1	1	C, DC, N, OV, Z
	ADDC	#lit10,Wn	Wd = lit10 + Wd + (C)	1	1	C, DC, N, OV, Z
	ADDC	Wb, Ws, Wd	Wd = Wb + Ws + (C)	1	1	C, DC, N, OV, Z
	ADDC	Wb,#lit5,Wd	Wd = Wb + lit5 + (C)	1	1	C, DC, N, OV, Z
AND	AND	f	f = f .AND. WREG	1	1	N, Z
	AND	f,WREG	WREG = f .AND. WREG	1	1	N, Z
	AND	#lit10,Wn	Wd = lit10 .AND. Wd	1	1	N, Z
	AND	Wb, Ws, Wd	Wd = Wb .AND. Ws	1	1	N, Z
	AND	Wb,#lit5,Wd	Wd = Wb .AND. lit5	1	1	N, Z
ASR	ASR	f	f = Arithmetic Right Shift f	1	1	C, N, OV, Z
	ASR	f,WREG	WREG = Arithmetic Right Shift f	1	1	C, N, OV, Z
	ASR	Ws,Wd	Wd = Arithmetic Right Shift Ws	1	1	C, N, OV, Z
	ASR	Wb, Wns, Wnd	Wnd = Arithmetic Right Shift Wb by Wns	1	1	N, Z
	ASR	Wb,#lit5,Wnd	Wnd = Arithmetic Right Shift Wb by lit5	1	1	N, Z
BCLR	BCLR	f,#bit4	Bit Clear f	1	1	None
	BCLR	Ws,#bit4	Bit Clear Ws	1	1	None
BRA	BRA	C,Expr	Branch if Carry	1	1 (2)	None
	BRA	GE, Expr	Branch if Greater than or Equal	1	1 (2)	None
	BRA	GEU, Expr	Branch if Unsigned Greater than or Equal	1	1 (2)	None
	BRA	GT, Expr	Branch if Greater than	1	1 (2)	None
	BRA	GTU, Expr	Branch if Unsigned Greater than	1	1 (2)	None
	BRA	LE, Expr	Branch if Less than or Equal	1	1 (2)	None
	BRA	LEU, Expr	Branch if Unsigned Less than or Equal	1	1 (2)	None
	BRA	LT, Expr	Branch if Less than	1	1 (2)	None
	BRA	LTU, Expr	Branch if Unsigned Less than	1	1 (2)	None
	BRA	N, Expr	Branch if Negative	1	1 (2)	None
	BRA	NC, Expr	Branch if Not Carry	1	1 (2)	None
	BRA	NN, Expr	Branch if Not Negative	1	1 (2)	None
	BRA	NOV, Expr	Branch if Not Overflow	1	1 (2)	None
	BRA	NZ,Expr	Branch if Not Zero	1	1 (2)	None
	BRA	OV, Expr	Branch if Overflow	1	1 (2)	None
	BRA	Expr	Branch Unconditionally	1	2	None
	BRA	Z,Expr	Branch if Zero	1	1 (2)	None
	BRA	Wn	Computed Branch	1	2	None
BSET	BSET	f,#bit4	Bit Set f	1	1	None
2021	BSET	Ws,#bit4	Bit Set Ws	1	1	None
BSW	BSW.C	Ws, Wb	Write C bit to Ws <wb></wb>	1	1	None
	BSW.Z	Ws,Wb	Write Z bit to Ws <wb></wb>	1	1	None
BTG	BTG	f,#bit4	Bit Toggle f	1	1	None
210	BTG	Ws,#bit4	Bit Toggle Ws	1	1	None
BTSC	BTSC	f,#bit4	Bit Test f, Skip if Clear	1	1 (2 or 3)	None
	BTSC	Ws,#bit4	Bit Test Ws, Skip if Clear	1	1 (2 or 3)	None

TABLE 32-2: INSTRUCTION SET OVERVIEW (CONTINUED)

Assembly Mnemonic		Assembly Syntax	Description	# of Words	# of Cycles	Status Flags Affected
BTSS	BTSS	f,#bit4	Bit Test f, Skip if Set	1	1 (2 or 3)	None
	BTSS	Ws,#bit4	Bit Test Ws, Skip if Set	1	1 (2 or 3)	None
BTST	BTST	f,#bit4	Bit Test f	1	1	Z
	BTST.C	Ws,#bit4	Bit Test Ws to C	1	1	С
	BTST.Z	Ws,#bit4	Bit Test Ws to Z	1	1	Z
	BTST.C	Ws, Wb	Bit Test Ws <wb> to C</wb>	1	1	С
	BTST.Z	Ws, Wb	Bit Test Ws <wb> to Z</wb>	1	1	Z
BTSTS	BTSTS	f,#bit4	Bit Test then Set f	1	1	Z
	BTSTS.C	Ws,#bit4	Bit Test Ws to C, then Set	1	1	С
	BTSTS.Z	Ws,#bit4	Bit Test Ws to Z, then Set	1	1	Z
CALL	CALL	lit23	Call Subroutine	2	2	None
	CALL	Wn	Call Indirect Subroutine	1	2	None
CLR	CLR	f	f = 0x0000	1	1	None
	CLR	WREG	WREG = 0x0000	1	1	None
	CLR	Ws	Ws = 0x0000	1	1	None
CLRWDT	CLRWDT		Clear Watchdog Timer	1	1	WDTO, Sleep
COM	COM	f	$f = \bar{f}$	1	1	N, Z
	COM	f,WREG	WREG = Ī	1	1	N, Z
	COM	Ws, Wd	Wd = Ws	1	1	N, Z
CP	CP	f	Compare f with WREG	1	1	C, DC, N, OV, Z
01	CP	Wb,#lit5	Compare Wb with lit5	1	1	C, DC, N, OV, Z
+	CP	Wb, Ws	Compare Wb with Ws (Wb – Ws)	1	1	C, DC, N, OV, Z
CP0	CP0	f	Compare f with 0x0000	1	1	C, DC, N, OV, Z
010	CP0	Ws	Compare Ws with 0x0000	1	1	C, DC, N, OV, Z
CPB	CPB	f	Compare f with WREG, with Borrow	1	1	C, DC, N, OV, Z
CID	CPB	Wb,#lit5	Compare Wb with lit5, with Borrow	1	1	C, DC, N, OV, Z
	СРВ	Wb, Ws	Compare Wb with Ws, with Borrow (Wb – Ws – \overline{C})	1	1	C, DC, N, OV, Z
CPSEQ	CPSEQ	Wb,Wn	Compare Wb with Wn, Skip if =	1	1 (2 or 3)	None
CPSGT	CPSGT	Wb,Wn	Compare Wb with Wn, Skip if >	1	1 (2 or 3)	None
CPSLT	CPSLT	Wb,Wn	Compare Wb with Wn, Skip if <	1	1 (2 or 3)	None
CPSNE	CPSNE	Wb,Wn	Compare Wb with Wn, Skip if ≠	1	1 (2 or 3)	None
DAW	DAW.B	Wn	Wn = Decimal Adjust Wn	1	1	С
DEC	DEC	f	f = f -1	1	1	C, DC, N, OV, Z
	DEC	f,WREG	WREG = f –1	1	1	C, DC, N, OV, Z
	DEC	Ws, Wd	Wd = Ws - 1	1	1	C, DC, N, OV, Z
DEC2	DEC2	f	f = f - 2	1	1	C, DC, N, OV, Z
	DEC2	f,WREG	WREG = f – 2	1	1	C, DC, N, OV, Z
	DEC2	Ws, Wd	Wd = Ws - 2	1	1	C, DC, N, OV, Z
DISI	DISI	#lit14	Disable Interrupts for k Instruction Cycles	1	1	None
DIV	DIV.SW	Wm, Wn	Signed 16/16-bit Integer Divide	1	18	N, Z, C, OV
	DIV.SD	Wm, Wn	Signed 32/16-bit Integer Divide	1	18	N, Z, C, OV
	DIV.UW	Wm, Wn	Unsigned 16/16-bit Integer Divide	1	18	N, Z, C, OV
	DIV.UD	Wm, Wn	Unsigned 32/16-bit Integer Divide	1	18	N, Z, C, OV
EXCH	EXCH	Wns, Wnd	Swap Wns with Wnd	1	1	None
FF1L	FF1L	Ws, Wnd	Find First One from Left (MSb) Side	1	1	С
FF1R	FF1R	Ws, Wnd	Find First One from Right (LSb) Side	1	1	С

TABLE 32-2: INSTRUCTION SET OVERVIEW (CONTINUED)

Assembly Mnemonic		Assembly Syntax	Description	# of Words	# of Cycles	Status Flags Affected
GOTO	GOTO	Expr	Go to Address	2	2	None
	GOTO	Wn	Go to Indirect	1	2	None
INC	INC	f	f = f + 1	1	1	C, DC, N, OV, Z
	INC	f,WREG	WREG = f + 1	1	1	C, DC, N, OV, Z
	INC	Ws,Wd	Wd = Ws + 1	1	1	C, DC, N, OV, Z
INC2	INC2	f	f = f + 2	1	1	C, DC, N, OV, Z
	INC2	f,WREG	WREG = f + 2	1	1	C, DC, N, OV, Z
	INC2	Ws,Wd	Wd = Ws + 2	1	1	C, DC, N, OV, Z
IOR	IOR	f	f = f .IOR. WREG	1	1	N, Z
	IOR	f,WREG	WREG = f.IOR. WREG	1	1	N, Z
	IOR	#lit10,Wn	Wd = lit10 .IOR. Wd	1	1	N, Z
	IOR	Wb, Ws, Wd	Wd = Wb .IOR. Ws	1	1	N, Z
	IOR	Wb,#lit5,Wd	Wd = Wb .IOR. lit5	1	1	N, Z
LNK	LNK	#lit14	Link Frame Pointer	1	1	None
LSR	LSR	f	f = Logical Right Shift f	1	1	C, N, OV, Z
	LSR	f,WREG	WREG = Logical Right Shift f	1	1	C, N, OV, Z
	LSR	Ws, Wd	Wd = Logical Right Shift Ws	1	1	C, N, OV, Z
	LSR	Wb, Wns, Wnd	Wnd = Logical Right Shift Wb by Wns	1	1	N, Z
	LSR	Wb, #lit5, Wnd	Wnd = Logical Right Shift Wb by lit5	1	1	N, Z
MOV	MOV	f,Wn	Move f to Wn	1	1	None
	MOV	[Wns+Slit10],Wnd	Move [Wns+Slit10] to Wnd	1	1	None
	MOV	f	Move f to f	1	1	N, Z
	MOV	f,WREG	Move f to WREG	1	1	N, Z
-	MOV	#lit16,Wn	Move 16-bit Literal to Wn	1	1	None
	MOV.b	#lit8,Wn	Move 8-bit Literal to Wn	1	1	None
	MOV	Wn,f	Move Wn to f	1	1	None
	MOV	Wns, [Wns+Slit10]	Move Wns to [Wns+Slit10]	1	1	
	MOV	Wso, Wdo	Move Ws to Wd	1	1	None
	MOV	WREG, f	Move WREG to f	1	1	N, Z
	MOV.D	Wns, Wd	Move Double from W(ns):W(ns+1) to Wd	1	2	None
	MOV.D	Ws, Wnd	Move Double from Ws to W(nd+1):W(nd)	1	2	None
MUL	MUL.SS	Wb, Ws, Wnd	{Wnd+1, Wnd} = Signed(Wb) * Signed(Ws)	1	1	None
1102	MUL.SU	Wb, Ws, Wnd	{Wnd+1, Wnd} = Signed(Wb) * Unsigned(Ws)	1	1	None
	MUL.US	Wb, Ws, Wnd	{Wnd+1, Wnd} = Unsigned(Wb) * Signed(Ws)	1	1	None
	MUL.UU	Wb, Ws, Wnd	{Wnd+1, Wnd} = Unsigned(Wb) * Unsigned(Ws)	1	1	None
	MUL.SU	Wb,#lit5,Wnd	{Wnd+1, Wnd} = Signed(Wb) * Unsigned(lit5)	1	1	None
	MUL.UU	Wb,#lit5,Wnd	{Wnd+1, Wnd} = Unsigned(Wb) * Unsigned(lit5)	1	1	None
	MUL	f	W3:W2 = f * WREG	1	1	None
NEG	NEG	f	$f = \overline{f} + 1$	1	1	C, DC, N, OV, Z
NEG			WREG = f + 1			
	NEG	f,WREG		1	1	C, DC, N, OV, Z
	NEG	Ws, Wd	$Wd = \overline{Ws} + 1$	1	1	C, DC, N, OV, Z
NOP	NOP		No Operation	1	1	None
	NOPR		No Operation	1	1	None
POP	POP	f	Pop f from Top-of-Stack (TOS)	1	1	None
	POP	Wdo	Pop from Top-of-Stack (TOS) to Wdo	1	1	None
	POP.D	Wnd	Pop from Top-of-Stack (TOS) to W(nd):W(nd+1)	1	2	None
	POP.S		Pop Shadow Registers	1	1	All
PUSH	PUSH	f	Push f to Top-of-Stack (TOS)	1	1	None
	PUSH	Wso	Push Wso to Top-of-Stack (TOS)	1	1	None
	PUSH.D	Wns	Push W(ns):W(ns+1) to Top-of-Stack (TOS)	1	2	None
	PUSH.S		Push Shadow Registers	1	1	None

TABLE 32-2: INSTRUCTION SET OVERVIEW (CONTINUED)

Assembly Mnemonic		Assembly Syntax	Description	# of Words	# of Cycles	Status Flags Affected
PWRSAV	PWRSAV	#lit1	Go into Sleep or Idle mode	1	1	WDTO, Sleep
RCALL	RCALL	Expr	Relative Call	1	2	None
	RCALL	Wn	Computed Call	1	2	None
REPEAT	REPEAT	#lit14	Repeat Next Instruction lit14 + 1 times	1	1	None
	REPEAT	Wn	Repeat Next Instruction (Wn) + 1 times	1	1	None
RESET	RESET		Software Device Reset	1	1	None
RETFIE	RETFIE		Return from Interrupt	1	3 (2)	None
RETLW	RETLW	#lit10,Wn	Return with Literal in Wn	1	3 (2)	None
RETURN	RETURN		Return from Subroutine	1	3 (2)	None
RLC	RLC	f	f = Rotate Left through Carry f	1	1	C, N, Z
	RLC	f,WREG	WREG = Rotate Left through Carry f	1	1	C, N, Z
	RLC	Ws,Wd	Wd = Rotate Left through Carry Ws	1	1	C, N, Z
RLNC	RLNC	f	f = Rotate Left (No Carry) f	1	1	N, Z
	RLNC	f,WREG	WREG = Rotate Left (No Carry) f	1	1	N, Z
	RLNC	Ws,Wd	Wd = Rotate Left (No Carry) Ws	1	1	N, Z
RRC	RRC	f	f = Rotate Right through Carry f	1	1	C, N, Z
	RRC	f,WREG	WREG = Rotate Right through Carry f	1	1	C, N, Z
	RRC	Ws,Wd	Wd = Rotate Right through Carry Ws	1	1	C, N, Z
RRNC	RRNC	f	f = Rotate Right (No Carry) f	1	1	N, Z
	RRNC	f,WREG	WREG = Rotate Right (No Carry) f	1	1	N, Z
	RRNC	Ws,Wd	Wd = Rotate Right (No Carry) Ws	1	1	N, Z
SE	SE	Ws, Wnd	Wnd = Sign-Extended Ws	1	1	C, N, Z
SETM	SETM	f	f = FFFFh	1	1	None
	SETM	WREG	WREG = FFFFh	1	1	None
	SETM	Ws	Ws = FFFFh	1	1	None
SL	SL	f	f = Left Shift f	1	1	C, N, OV, Z
	SL	f,WREG	WREG = Left Shift f	1	1	C, N, OV, Z
	SL	Ws,Wd	Wd = Left Shift Ws	1	1	C, N, OV, Z
	SL	Wb, Wns, Wnd	Wnd = Left Shift Wb by Wns	1	1	N, Z
	SL	Wb,#lit5,Wnd	Wnd = Left Shift Wb by lit5	1	1	N, Z
SUB	SUB	f	f = f – WREG	1	1	C, DC, N, OV, Z
	SUB	f,WREG	WREG = f – WREG	1	1	C, DC, N, OV, Z
	SUB	#lit10,Wn	Wn = Wn – lit10	1	1	C, DC, N, OV, Z
	SUB	Wb, Ws, Wd	Wd = Wb – Ws	1	1	C, DC, N, OV, Z
	SUB	Wb,#lit5,Wd	Wd = Wb – lit5	1	1	C, DC, N, OV, Z
SUBB	SUBB	f	$f = f - WREG - (\overline{C})$	1	1	C, DC, N, OV, Z
COLL	SUBB	f,WREG	WREG = $f - WREG - (\overline{C})$	1	1	C, DC, N, OV, Z
		<u> </u>	$Wn = Wn - lit10 - (\overline{C})$			
	SUBB	#lit10,Wn	<u>`</u>	1	1	C, DC, N, OV, Z
	SUBB	Wb, Ws, Wd	Wd = Wb - Ws - (C)	1	1	C, DC, N, OV, Z
	SUBB	Wb,#lit5,Wd	Wd = Wb - lit5 - (C)	1	1	C, DC, N, OV, Z
SUBR	SUBR	f	f = WREG – f	1	1	C, DC, N, OV, Z
	SUBR	f,WREG	WREG = WREG – f	1	1	C, DC, N, OV, Z
	SUBR	Wb, Ws, Wd	Wd = Ws – Wb	1	1	C, DC, N, OV, Z
	SUBR	Wb,#lit5,Wd	Wd = lit5 – Wb	1	1	C, DC, N, OV, Z
SUBBR	SUBBR	f	$f = WREG - f - (\overline{C})$	1	1	C, DC, N, OV, Z
	SUBBR	f,WREG	WREG = WREG – $f - (\overline{C})$	1	1	C, DC, N, OV, Z
	SUBBR	Wb, Ws, Wd	$Wd = Ws - Wb - (\overline{C})$	1	1	C, DC, N, OV, Z
	SUBBR	Wb,#lit5,Wd	$Wd = lit5 - Wb - (\overline{C})$	1	1	C, DC, N, OV, Z
SWAP	SWAP.b	Wn	Wn = Nibble Swap Wn	1	1	None
	SWAP	Wn	Wn = Byte Swap Wn	1	1	None

TABLE 32-2: INSTRUCTION SET OVERVIEW (CONTINUED)

Assembly Mnemonic		Assembly Syntax	Description	# of Words	# of Cycles	Status Flags Affected
TBLRDH	TBLRDH	Ws,Wd	Read Prog<23:16> to Wd<7:0>	1	2	None
TBLRDL	TBLRDL	Ws, Wd	Read Prog<15:0> to Wd	1	2	None
TBLWTH	TBLWTH	Ws, Wd	Write Ws<7:0> to Prog<23:16>	1	2	None
TBLWTL	TBLWTL	Ws, Wd	Write Ws to Prog<15:0>	1	2	None
ULNK	ULNK		Unlink Frame Pointer	1	1	None
XOR	XOR	f	f = f .XOR. WREG	1	1	N, Z
	XOR	f,WREG	WREG = f .XOR. WREG	1	1	N, Z
	XOR	#lit10,Wn	Wd = lit10 .XOR. Wd	1	1	N, Z
	XOR	Wb, Ws, Wd	Wd = Wb .XOR. Ws	1	1	N, Z
	XOR	Wb,#lit5,Wd	Wd = Wb .XOR. lit5	1	1	N, Z
ZE	ZE	Ws, Wnd	Wnd = Zero-Extend Ws	1	1	C, Z, N

PIC24FJ128GB204 FAMILY								
NOTES:								

33.0 ELECTRICAL CHARACTERISTICS

This section provides an overview of the PIC24FJ128GB204 family electrical characteristics. Additional information will be provided in future revisions of this document as it becomes available.

Absolute maximum ratings for the PIC24FJ128GB204 family are listed below. Exposure to these maximum rating conditions for extended periods may affect device reliability. Functional operation of the device at these, or any other conditions above the parameters indicated in the operation listings of this specification, is not implied.

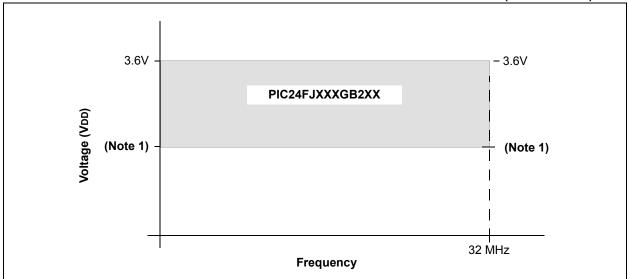
Absolute Maximum Ratings(†)

Ambient temperature under bias40°C to +100°C
Storage temperature65°C to +150°C
/oltage on VDD with respect to Vss0.3V to +4.0V
/oltage on any general purpose digital or analog pin (not 5.5V tolerant) with respect to Vss0.3V to (VDD + 0.3V)
/oltage on any general purpose digital or analog pin (5.5V tolerant, including MCLR) with respect to Vss:
When VDD = 0V:0.3V to +4.0V
When VDD \geq 2.0V:0.3V to +6.0V
/oltage on AVDD with respect to Vss(VDD – 0.3V) to (lesser of: 4.0V or (VDD + 0.3V))
/oltage on AVss with respect to Vss0.3V to +0.3V
/oltage on VBAT with respect to Vss0.3V to +4.0V
/oltage on VUSB3V3 with respect to Vss(VCAP – 0.3V) to +4.0V
/oltage on VBUS with respect to VSS0.3V to +6.0V
/oltage on D+ or D- with respect to Vss:
0Ω source impedance (Note 1)0.5V to (VUSB3V3 + 0.5V)
Source Impedance \geq 28 Ω , VUSB3V3 \geq 3.0V1.0V to +4.6V
Maximum current out of Vss pin
Maximum current into VDD pin (Note 2)250 mA
Maximum output current sunk by any I/O pin25 mA
Maximum output current sourced by any I/O pin
Maximum current sunk by all ports
Maximum current sourced by all ports (Note 2)

- Note 1: The original USB 2.0 Specifications indicated that USB devices should withstand 24-hour short circuits of D+ or D- to VBUS voltages. This requirement was later removed in an Engineering Change Notice (ECN) supplement to the USB Specifications, which supersedes the original specifications. PIC24FJ128GB204 family devices will typically be able to survive this short-circuit test, but it is recommended to adhere to the absolute maximum specified here to avoid damaging the device.
 - 2: Maximum allowable current is a function of device maximum power dissipation (see Table 33-1).
- † **NOTICE:** Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

33.1 DC Characteristics

FIGURE 33-1: PIC24FJ128GB204 FAMILY VOLTAGE-FREQUENCY GRAPH (INDUSTRIAL)



Note 1: Lower operating boundary is 2.0V or VBOR (when BOR is enabled), whichever is lower. For best analog performance, operation above 2.2V is suggested, but not required.

TABLE 33-1: THERMAL OPERATING CONDITIONS

Rating	Symbol	Min	Тур	Max	Unit
PIC24FJ128GB204:					
Operating Junction Temperature Range	TJ	-40	_	+125	°C
Operating Ambient Temperature Range	TA	-40	_	+85	°C
Power Dissipation: Internal Chip Power Dissipation: PINT = VDD x (IDD – Σ IOH) I/O Pin Power Dissipation: PI/O = Σ ({VDD – VOH} x IOH) + Σ (VOL x IOL)	PD				
Maximum Allowed Power Dissipation	Ромах	(ΓJ — TA)/θJ	IA	W

TABLE 33-2: THERMAL PACKAGING CHARACTERISTICS

Characteristic	Symbol	Тур	Max	Unit	Notes
Package Thermal Resistance, 7.50 mm 28-Pin SOIC	θЈА	49	_	°C/W	(Note 1)
Package Thermal Resistance, 6x6x0.9 mm 28-Pin QFN-S	θЈА	33.7	_	°C/W	(Note 1)
Package Thermal Resistance, 8x8 mm 44-Pin QFN	θЈА	28	_	°C/W	(Note 1)
Package Thermal Resistance, 10x10x1 mm 44-Pin TQFP	θЈА	39.3	_	°C/W	(Note 1)
Package Thermal Resistance, 5.30 mm 28-Pin SSOP	θЈА		_	°C/W	(Note 1)
Package Thermal Resistance, 300 mil 28-Pin SPDIP	θЈА	_	_	°C/W	(Note 1)

Note 1: Junction to ambient thermal resistance; Theta-JA (θJA) numbers are achieved by package simulations.

TABLE 33-3: DC CHARACTERISTICS: TEMPERATURE AND VOLTAGE SPECIFICATIONS

DC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature $ -40^{\circ}C \leq TA \leq +85^{\circ}C \text{ for Industrial} \\ -40^{\circ}C \leq TA \leq +125^{\circ}C \text{ for Extended} $						
Param No.	Symbol	Characteristic	Min	Тур	Max	Units	Conditions		
Operati	ing Volta	je			•				
DC10	VDD	Supply Voltage	2.0	_	3.6	V	BOR disabled		
			VBOR	_	3.6	V	BOR enabled		
DC12	VDR	RAM Data Retention Voltage ⁽¹⁾	Greater of: VPORREL or VBOR	_	_	V	VBOR used only if BOR is enabled (BOREN = 1)		
DC16	VPOR	VDD Start Voltage to Ensure Internal Power-on Reset Signal	Vss	_	_	V	(Note 2)		
DC16A	VPORREL	VDD Power-on Reset Release Voltage	1.80	1.88	1.95	V	(Note 3)		
DC17A	SRVDD	Recommended VDD Rise Rate to Ensure Internal Power-on Reset Signal	0.05	_		V/ms	0-3.3V in 66 ms, 0-2.5V in 50 ms (Note 2)		
DC17B	VBOR	Brown-out Reset Voltage on VDD Transition, High-to-Low	2.0	2.1	2.2	V	(Note 3)		

Note 1: This is the limit to which VDD may be lowered and the RAM contents will always be retained.

^{2:} If the VPOR or SRVDD parameters are not met, or the application experiences slow power-down VDD ramp rates, it is recommended to enable and use BOR.

^{3:} On a rising VDD power-up sequence, application firmware execution begins at the higher of the VPORREL or VBOR level (when BOREN = 1).

TABLE 33-4: DC CHARACTERISTICS: OPERATING CURRENT (IDD)

DC CHARAC	CTERISTICS		Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature $ -40^{\circ}C \leq TA \leq +85^{\circ}C \text{ for Industrial} \\ -40^{\circ}C \leq TA \leq +125^{\circ}C \text{ for Extended} $									
Parameter No.	Typical ⁽¹⁾	Max	Units	Operating Temperature	Conditions							
Operating C	Operating Current (IDD) ⁽²⁾											
DC19	0.20	0.28	mA	-40°C to +125°C	2.0V	0.5 MIPS,						
DC20A	0.21	0.28	mA	-40°C to +125°C	3.3V	Fosc = 1 MHz						
DC20	0.38	0.52	mA	-40°C to +125°C	2.0V	1 MIPS,						
	0.39	0.52	mA	-40°C to +125°C	3.3V	Fosc = 2 MHz						
DC23	1.5	2.0	mA	-40°C to +125°C	2.0V	4 MIPS,						
	1.5	2.0	mA	-40°C to +125°C	3.3V	Fosc = 8 MHz						
DC24	5.6	7.6	mA	-40°C to +125°C	2.0V	16 MIPS,						
	5.7	7.6	mA	-40°C to +125°C	3.3V	Fosc = 32 MHz						
DC31	23	78	μА	-40°C to +85°C	2.0V	LPRC (15.5 KIPS),						
	_	98	μΑ	+125°C	2.0V	Fosc = 31 kHz						
	25 80		μА	-40°C to +85°C	3.3V							
	_	100	μΑ	+125°C	3.3V							

Note 1: Data in the "Typical" column is at 3.3V, +25°C unless otherwise stated. Typical parameters are for design guidance only and are not tested.

^{2:} The test conditions for all IDD measurements are as follows: OSC1 driven with external square wave from rail-to-rail. All I/O pins are configured as outputs and driven to Vss. MCLR = VDD; WDT and FSCM are disabled. CPU, program memory and data memory are operational. No peripheral modules are operating; however, every peripheral is being clocked (PMDx bits are all zeroed).

TABLE 33-5: DC CHARACTERISTICS: IDLE CURRENT (IIDLE)

DC CHARAC	CTERISTICS		Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{TA} \le +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \le \text{TA} \le +125^{\circ}\text{C}$ for Extended					
Parameter No.	Typical ⁽¹⁾	Max	Units	Operating Temperature	VDD	Conditions		
Idle Current	(IIDLE) ⁽²⁾							
DC40	116	150	μА	-40°C to +85°C	2.0V	1 MIPS,		
	_	170	μА	+125°C	2.0V	Fosc = 2 MHz		
	123	160	μΑ	-40°C to +85°C	3.3V			
	_	180	μΑ	+125°C	3.3V			
DC43	0.39	0.5	mA	-40°C to +85°C	2.0V	4 MIPS,		
	_	0.52	mA	+125°C	2.0V	Fosc = 8 MHz		
	0.41	0.54	mA	-40°C to +85°C	3.3V			
	_	0.56	mA	+125°C	3.3V			
DC47	1.5	1.9	mA	-40°C to +85°C	2.0V	16 MIPS,		
	_	2	mA	+125°C	2.0V	Fosc = 32 MHz		
	1.6	2.0	mA	-40°C to +85°C	3.3V			
	_	2.1	mA	+125°C	3.3V			
DC50	0.54	0.61	mA	-40°C to +85°C	2.0V	4 MIPS (FRC),		
	0.54	0.64	mA	-40°C to +85°C	3.3V	Fosc = 8 MHz		
DC51	17	78	μΑ	-40°C to +85°C	2.0V	LPRC (15.5 KIPS),		
		128	μΑ	+125°C	2.0V	Fosc = 31 kHz		
	18	80	μΑ	-40°C to +85°C	3.3V			
	_	130	μΑ	+125°C	3.3V			

Note 1: Data in the "Typical" column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

^{2:} Base IIDLE current is measured with the core off, the clock on and all modules turned off. Peripheral Module Disable SFR registers are zeroed. All I/O pins are configured as inputs and pulled to Vss.

TABLE 33-6: DC CHARACTERISTICS: POWER-DOWN CURRENT (IPD)

DC CHARA	CTERISTIC	s		emperature -40°	$^{\circ}C \leq TA \leq +$	to 3.6V (unless otherwise stated) -85°C for Industrial -125°C for Extended			
Parameter No. Typical ⁽¹⁾ Max		Units	Operating Temperature	VDD	Conditions				
Power-Dov	vn Current ((IPD) ^(5,6)							
DC60	2.9	17	μΑ	-40°C					
	4.3	17	μΑ	+25°C					
	8.3	27.5	μΑ	+60°C	2.0V				
	20	27.5	μΑ	+85°C					
	_	79	μΑ	+125°C		- Sleep ⁽²⁾			
	2.9	18	μΑ	-40°C		- Sieep			
	4.3	18	μΑ	+25°C					
	8.4	28	μΑ	+60°C	3.3V				
	20.5	28	μΑ	+85°C					
	_	80	μΑ	+125°C					
DC61	0.07	_	μΑ	-40°C					
	0.38	_	μΑ	+25°C	0.01/				
	2.6	_	μΑ	+60°C	2.0V				
	9.0	_	μΑ	+125°C		Lava Valta and Olama (3)			
	0.09	_	μΑ	-40°C		Low-Voltage Sleep ⁽³⁾			
	0.42	_	μΑ	+25°C	0.01/				
	2.75	_	μΑ	+60°C	3.3V				
	9.0	_	μΑ	+125°C					
DC70	0.1	700	nA	-40°C					
	18	700	nA	+25°C					
	230	1700	nA	+60°C	2.0V				
	1.8	3.0	μΑ	+85°C					
	_	24	μА	+125°C		Doon Sloop			
	5	900	nA	-40°C		Deep Sleep			
	75	900	nA	+25°C					
	540	3450	nA	+60°C	3.3V				
	1.5	6.0	μА	+85°C					
	_	48	μΑ	+125°C					
DC74	0.4	2.0	μΑ	-40°C to +125°C	0V	RTCC with VBAT mode (LPRC/SOSC)(4)			

Note 1: Data in the Typical column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

- 2: The retention low-voltage regulator is disabled; RETEN (RCON<12>) = 0, \overline{LPCFG} (CW1<10>) = 1.
- 3: The retention low-voltage regulator is enabled; RETEN (RCON<12>) = 1, \overline{LPCFG} (CW1<10>) = 0.
- **4:** The VBAT pin is connected to the battery and RTCC is running with VDD = 0.
- **5:** Base IPD is measured with all peripherals and clocks shut down. All I/Os are configured as inputs and pulled to Vss. WDT, etc., are all switched off.
- **6:** These currents are measured on the device containing the most memory in this family.

TABLE 33-7: DC CHARACTERISTICS: △ CURRENT (BOR, WDT, DSBOR, DSWDT)⁽⁴⁾

DC CHARAC	TERISTICS		Standard O Operating to	emperature -40	$^{\circ}$ C \leq TA \leq +8	o 3.6V (unless otherwise stated) 85°C for Industrial 125°C for Extended
Parameter No.	Typical ⁽¹⁾	Max	Units	Operating Temperature	VDD	Conditions
Incremental (Current Brow	n-out Rese	t (∆BOR) ⁽²⁾			
DC25	3.1	5.0	μΑ	-40°C to +125°C	2.0V	∆BOR ⁽²⁾
	4.3	6.0	μΑ	-40°C to +125°C	3.3V	
Incremental (Current Watc	hdog Timer	(∆WDT) ⁽²⁾			
DC71	0.8	1.5	μΑ	-40°C to +125°C	2.0V	∧WDT ⁽²⁾
	0.8	1.5	μΑ	-40°C to +125°C	3.3V	
Incremental (Current High	/Low-Voltag	e Detect (∆⊦	HLVD) ⁽²⁾		
DC75	4.2	15	μΑ	-40°C to +125°C	2.0V	ΛΗLVD ⁽²⁾
	4.2	15	μΑ	-40°C to +125°C	3.3V	ZHLVD.
Incremental (Current Real-	-Time Clock	and Calend	lar (∆RTCC) ⁽²⁾		
DC77	0.3	1.0	μΑ	-40°C to +125°C	2.0V	△RTCC (with SOSC) ⁽²⁾
	0.35	1.0	μΑ	-40°C to +125°C	3.3V	ARTEC (WILL SOSC)
DC77A	0.3	1.0	μΑ	-40°C to +125°C	2.0V	△RTCC (with LPRC) ⁽²⁾
	0.35	1.0	μΑ	-40°C to +125°C	3.3V	AKIGG (WILLIERG):
Incremental C	Current Deep	Sleep BOR	(ADSBOR)	(2)		
DC81	0.11	0.40	μΑ	-40°C to +125°C	2.0V	∆Deep Sleep BOR ⁽²⁾
	0.12	0.40	μΑ	-40°C to +125°C	3.3V	Apeeb Sleep BOK.
Incremental (Current Deep	Sleep Wate	hdog Time	r Reset (∆DSWDT	') ⁽²⁾	
DC80	0.24	0.40	μΑ	-40°C to +125°C	2.0V	∆Deep Sleep WDT ⁽²⁾
	0.24	0.40	μΑ	-40°C to +125°C	3.3V	Topech Siech MD1.
VBAT A/D Moi	nitor ⁽³⁾					
DC91	1.5		μΑ	-40°C to +125°C	3.3V	VBAT = 2V
	4	_	μΑ	-40°C to +125°C	3.3V	VBAT = 3.3V

Note 1: Data in the Typical column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

^{2:} Incremental current while the module is enabled and running.

^{3:} The A/D channel is connected to the VBAT pin internally; this is the current during A/D VBAT operation.

^{4:} The Δ current is the additional current consumed when the module is enabled. This current should be added to the base IPD current.

TABLE 33-8: DC CHARACTERISTICS: I/O PIN INPUT SPECIFICATIONS

DC CHARACTERISTICS Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{TA} \le +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \le \text{TA} \le +125^{\circ}\text{C}$ for Extended

	1	1	-40 C S IA S + 125 C IOI Exterided				
Param No.	Symbol	Characteristic	Min	Typ ⁽¹⁾	Max	Units	Conditions
	VIL	Input Low Voltage ⁽³⁾					
DI10		I/O Pins with ST Buffer	Vss		0.2 VDD	V	
DI11		I/O Pins with TTL Buffer	Vss	_	0.15 VDD	V	
DI15		MCLR	Vss		0.2 VDD	V	
DI16		OSCI (XT mode)	Vss	_	0.2 VDD	V	
DI17		OSCI (HS mode)	Vss	_	0.2 VDD	V	
DI18		I/O Pins with I ² C™ Buffer	Vss	_	0.3 VDD	V	
DI19		I/O Pins with SMBus Buffer	Vss	_	0.8	V	SMBus enabled
	VIH	Input High Voltage ⁽³⁾					
DI20		I/O Pins with ST Buffer: with Analog Functions Digital Only	0.8 Vdd 0.8 Vdd	_ _	VDD 5.5	V V	
DI21		I/O Pins with TTL Buffer: with Analog Functions Digital Only	0.25 VDD + 0.8 0.25 VDD + 0.8	_ _	VDD 5.5	V V	
DI25		MCLR	0.8 Vdd		VDD	V	
DI26		OSCI (XT mode)	0.7 VDD	_	VDD	V	
DI27		OSCI (HS mode)	0.7 VDD	_	VDD	V	
DI28		I/O Pins with I ² C Buffer: with Analog Functions Digital Only	0.7 Vdd 0.7 Vdd	_ _	VDD 5.5	V V	
DI29		I/O Pins with SMBus Buffer: with Analog Functions	2.1	_	VDD	٧	$2.5V \le VPIN \le VDD$
		Digital Only	2.1	_	5.5	V	
DI30	ICNPU	CNxx Pull-up Current	150	340	550	μΑ	VDD = 3.3V, VPIN = VSS
DI30A	ICNPD	CNxx Pull-Down Current	150	310	550	μΑ	VDD = 3.3V, VPIN = VDD
DI50	lıL	Input Leakage Current ⁽²⁾ I/O Ports	_	_	±1	μΑ	Vss ≤ VPIN ≤ VDD, pin at high-impedance
DI51		Analog Input Pins	_	_	±1	μΑ	$Vss \leq VPIN \leq VDD, \\ pin at high-impedance$
DI55		MCLR	_	_	±1	μΑ	$Vss \leq Vpin \leq Vdd$
DI56		OSCI/CLKI	_	_	±1	μΑ	$\label{eq:VSS} \begin{array}{l} \text{VSS} \leq \text{VPIN} \leq \text{VDD,} \\ \text{EC, XT and HS modes} \end{array}$

Note 1: Data in the "Typ" column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

^{2:} Negative current is defined as current sourced by the pin.

^{3:} Refer to Table 1-3 for I/O pin buffer types.

TABLE 33-9: DC CHARACTERISTICS: I/O PIN OUTPUT SPECIFICATIONS

DC CHA	ARACTER	ISTICS	Standard C (unless oth Operating t	nerwise	stated) ure -4	O°C ≤ TA	oV to 3.6V ≤ +85°C for Industrial ≤ +125°C for Extended		
Param No.	Symbol	Characteristic	Min Typ ⁽¹⁾ Max Units Conditions						
	Vol	Output Low Voltage							
DO10		I/O Ports	_	_	0.4	V	IOL = 6.6 mA, VDD = 3.6V		
			_	_	0.4	V	IOL = 5.0 mA, VDD = 2V		
DO16		OSCO/CLKO	_	_	0.4	V	IOL = 6.6 mA, VDD = 3.6V		
			_	_	0.4	V	IOL = 5.0 mA, VDD = 2V		
	Vон	Output High Voltage							
DO20		I/O Ports	3.0	_	_	V	IOH = -3.0 mA, VDD = 3.6V		
			2.4	_	_	V	IOH = -6.0 mA, VDD = 3.6V		
			1.65	_	_	V	IOH = -1.0 mA, VDD = 2V		
			1.4	_	_	V	IOH = -3.0 mA, VDD = 2V		
DO26		OSCO/CLKO	2.4	_	_	V	IOH = -6.0 mA, VDD = 3.6V		
			1.4	_		V	IOH = -1.0 mA, VDD = 2V		

Note 1: Data in the "Typ" column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

TABLE 33-10: DC CHARACTERISTICS: PROGRAM MEMORY

DC CH	ARACTE	Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature $ -40^{\circ}C \leq TA \leq +85^{\circ}C \text{ for Industrial } \\ -40^{\circ}C \leq TA \leq +125^{\circ}C \text{ for Extended} $								
Param No.	Symbol	Characteristic	Min	Typ ⁽¹⁾	Max	Units Conditions				
		Program Flash Memory								
D130	EP	Cell Endurance	20000	_	_	E/W	-40°C to +125°C			
D131	VPR	VDD for Read	VMIN	_	3.6	V	VміN = Minimum Operating Voltage			
D132B		VDD for Self-Timed Write	VMIN	_	3.6	V	Vмін = Minimum Operating Voltage			
D133A	Tiw	Self-Timed Word Write Cycle Time	_	20	_	μS				
		Self-Timed Row Write Cycle Time	_	1.5	_	ms				
D133B	TIE	Self-Timed Page Erase Time	20	_	40	ms				
D134	TRETD	Characteristic Retention	20		_	Year	If no other specifications are violated			
D135	IDDP	Supply Current during Programming	_	5	_	mA				
D136	VOTP	OTP Programming	3.1	_	3.6	V				
D137	Тотр	OTP Memory Write/Bit	_	500	_	μS				

Note 1: Data in the "Typ" column is at 3.3V, +25°C unless otherwise stated.

TABLE 33-11: INTERNAL VOLTAGE REGULATOR SPECIFICATIONS

Operating Conditions:

Operating temperature $-40^{\circ}C \le TA \le +85^{\circ}C$ for Industrial $-40^{\circ}C \le TA \le +125^{\circ}C$ for Extended

Param No.	Symbol	Characteristics	Min	Тур	Max	Units	Comments
DVR	TVREG	Voltage Regulator Start-up Time	_	10	_	μS	VREGS = 1 with any POR or BOR
DVR10	VBG	Internal Band Gap Reference	_	1.2	_	V	
DVR11	TBG	Band Gap Reference Start-up Time	-	1	_	ms	
DVR20	VRGOUT	Regulator Output Voltage	_	1.8	_	V	VDD > 1.9V
DVR21	CEFC	External Filter Capacitor Value	4.7	10	_	μF	Series resistance < 3Ω recommended; < 5Ω required
DVR30	VLVR	Low-Voltage Regulator Output Voltage	_	1.2	_	V	RETEN = 1, LPCFG = 0

TABLE 33-12: HIGH/LOW-VOLTAGE DETECT CHARACTERISTICS

Operating Conditions:

Operating temperature $-40^{\circ}\text{C} \le \text{Ta} \le +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \le \text{Ta} \le +125^{\circ}\text{C}$ for Extended

Param No.	Symbol	Charac	cteristic	Min	Тур	Max	Units	Conditions
DC18	VHLVD	HLVD Voltage on VDD	HLVDL<3:0> = 0100 ⁽¹⁾	3.45	3.59	3.74	V	
		Transition	HLVDL<3:0> = 0101	3.33	3.45	3.58	V	
			HLVDL<3:0> = 0110	3.0	3.125	3.25	V	
			HLVDL<3:0> = 0111	2.8	2.92	3.04	V	
			HLVDL<3:0> = 1000	2.7	2.81	2.93	V	
			HLVDL<3:0> = 1001	2.50	2.6	2.70	V	
			HLVDL<3:0> = 1010	2.4	2.52	2.64	V	
			HLVDL<3:0> = 1011	2.30	2.4	2.50	V	
			HLVDL<3:0> = 1100	2.20	2.29	2.39	V	
			HLVDL<3:0> = 1101	2.1	2.19	2.28	V	
			HLVDL<3:0> = 1110	2.0	2.08	2.17	V	
DC101	VTHL	HLVD Voltage on HLVDIN Pin Transition	HLVDL<3:0> = 1111	_	1.2	_	V	

Note 1: Trip points for values of HLVD<3:0>, from '0000' to '0011', are not implemented.

TABLE 33-13: COMPARATOR DC SPECIFICATIONS

Operating Conditions: 2.0V < V_{DD} < 3.6V

Operating temperature $-40^{\circ}C \le TA \le +85^{\circ}C$ for Industrial $-40^{\circ}C \le TA \le +125^{\circ}C$ for Extended

Param No.	Symbol	Characteristic	Min	Тур	Max	Units	Comments
D300	VIOFF	Input Offset Voltage	_	20	±40	mV	(Note 1)
D301	VICM	Input Common-Mode Voltage	0	_	VDD	V	(Note 1)
D302	CMRR	Common-Mode Rejection Ratio	55	_	_	dB	(Note 1)
D306	IQCMP	AVDD Quiescent Current per Comparator	_	27	_	μs	Comparator enabled
D307	TRESP	Response Time	_	300	_	ns	(Note 2)
D308	Тмс2о∨	Comparator Mode Change to Valid Output	_		10	μs	

Note 1: Parameters are characterized but not tested.

2: Measured with one input at VDD/2 and the other transitioning from Vss to VDD, 40 mV step, 15 mV overdrive.

TABLE 33-14: COMPARATOR VOLTAGE REFERENCE DC SPECIFICATIONS

Operating Conditions: 2.0V < VDD < 3.6V

Operating temperature $-40^{\circ}\text{C} \le \text{TA} \le +85^{\circ}\text{C}$ for Industrial

 $-40^{\circ}C \le TA \le +125^{\circ}C$ for Extended

Param No.	Symbol	Characteristic	Min	Тур	Max	Units	Comments
VR310	TSET	Settling Time	_	_	10	μs	(Note 1)
VRD311	CVRAA	Absolute Accuracy	-100	_	100	mV	
VRD312	CVRur	Unit Resistor Value (R)	_	4.5	_	kΩ	

Note 1: Measures the interval while CVR<4:0> transitions from '11111' to '00000'.

TABLE 33-15: VBAT OPERATING VOLTAGE SPECIFICATIONS

Param No.	Symbol	Characteristic	Min	Тур	Max	Units	Comments
DVB01	VBT	Operating Voltage	1.6	_	3.6	V	Battery connected to the VBAT pin
DVB10		VBAT A/D Monitoring Voltage Specification ⁽¹⁾	1.6	_	3.6		A/D monitoring the VBAT pin using the internal A/D channel

Note 1: Measuring the A/D value using the A/D is represented by the equation: Measured Voltage = ((VBAT/2)/VDD) * 4096) for 12-bit A/D

TABLE 33-16: CTMU CURRENT SOURCE SPECIFICATIONS

DC CHA	RACTE	ERISTICS	Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{TA} \le +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \le \text{TA} \le +125^{\circ}\text{C}$ for Extended							
Param No.	Sym	Characteristic	Min	Typ ⁽¹⁾	Conditions					
DCT10	IOUT1	CTMU Current Source, Base Range	208	550	797	nA	CTMUICON<9:8> = 00			
DCT11	lout2	CTMU Current Source, 10x Range	3.32	5.5	7.67	μΑ	CTMUICON<9:8> = 01	2 5V < VDD < VDDMAX		
DCT12	Іоит3	CTMU Current Source, 100x Range	32.22	55	77.78	μА	CTMUICON<9:8> = 10	2.5V \ VDD \ VDDIVIAX		
DCT13	lout4	CTMU Current Source, 1000x Range	322	550	777	μΑ	CTMUICON<9:8> = 11 ⁽²⁾			
DCT21	VΔ	Temperature Diode Voltage Change per Degree Celsius	_	-3	_	mV/°C				

- Note 1: Nominal value at the center point of the current trim range (CTMUICON<15:10> = 000000).
 - 2: Do not use this current range with a temperature sensing diode.
 - 3: Maximum values are tested for +85°C.

TABLE 33-17: USB ON-THE-GO MODULE SPECIFICATIONS

DC CHA	RACTERI	Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{Ta} \le +85^{\circ}\text{C}$ for Industrial								
Param No.	Symbol	Characteristic	Min	Тур	Max	Units	Comments			
Operating Voltage										
DUS01	VUSB3V3	USB Supply Voltage	Greater of: 3.0 or (VDD – 0.3V)	3.3	3.6	V	USB module enabled			
			(VDD – 0.3V) ⁽¹⁾	_	3.6	V	USB disabled, D+/D- are unused and externally pulled low or left in a high-impedance state			
			(VDD – 0.3V)	VDD	3.6	V	USB disabled, D+/D- are used as general purpose I/Os			

Note 1: The VUSB3V3 pin may also be left in a high-impedance state under these conditions. However, if the voltage floats below (VDD – 0.3V), this may result in higher IPD currents than specified.

33.2 AC Characteristics and Timing Parameters

The information contained in this section defines the PIC24FJ128GB204 family AC characteristics and timing parameters.

TABLE 33-18: TEMPERATURE AND VOLTAGE SPECIFICATIONS - AC

	Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated)						
AC CHARACTERISTICS	Operating temperature -40°C ≤ TA ≤ +85°C for Industrial						
AC CHARACTERISTICS	-40°C ≤ TA ≤ +125°C for Extended						
	Operating voltage VDD range as described in Section 33.1 "DC Characteristics" .						

FIGURE 33-2: LOAD CONDITIONS FOR DEVICE TIMING SPECIFICATIONS

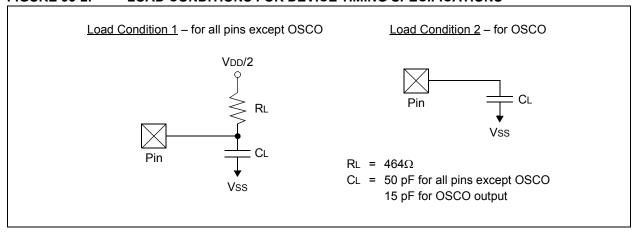


TABLE 33-19: CAPACITIVE LOADING REQUIREMENTS ON OUTPUT PINS

Param No.	Symbol	Characteristic	Min	Typ ⁽¹⁾	Max	Units	Conditions
DO50	Cosco	OSCO/CLKO Pin	_	_	15	pF	In XT and HS modes when external clock is used to drive OSCI
DO56	Cio	All I/O Pins and OSCO	_	_	50	pF	EC mode
DO58	Св	SCLx, SDAx	_	_	400	pF	In I ² C™ mode

Note 1: Data in the "Typ" column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

FIGURE 33-3: EXTERNAL CLOCK TIMING

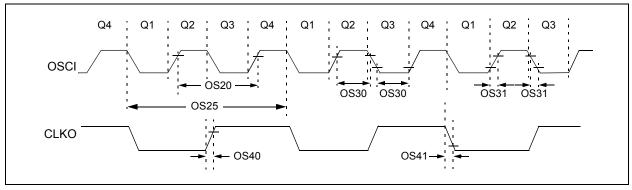


TABLE 33-20: EXTERNAL CLOCK TIMING REQUIREMENTS

40.011	A D A OTE	DIOTION	Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated)							
AC CH	ARACTE	RISTICS	Operating ten	nperature	-40 °C \leq TA \leq +85°C for Industrial -40 °C \leq TA \leq +125°C for Extended					
Param No.	Symbol	Characteristic	Min Typ ⁽¹⁾ Max Ur				Conditions			
OS10	Fosc	External CLKI Frequency (External clocks allowed only in EC mode)	DC 4		32 48	MHz MHz	EC ECPLL (Note 2)			
		Oscillator Frequency	3.5 4 10 12 31		10 8 32 32 33	MHz MHz MHz MHz kHz	XT XTPLL HS HSPLL SOSC			
OS20	Tosc	Tosc = 1/Fosc	_		_	_	See Parameter OS10 for Fosc value			
OS25	Tcy	Instruction Cycle Time ⁽³⁾	62.5	_	DC	ns				
OS30	TosL, TosH	External Clock in (OSCI) High or Low Time	0.45 x Tosc		_	ns	EC			
OS31	TosR, TosF	External Clock in (OSCI) Rise or Fall Time	_		20	ns	EC			
OS40	TckR	CLKO Rise Time ⁽⁴⁾	_	6	10	ns				
OS41	TckF	CLKO Fall Time(4)	_	6	10	ns				

- **Note 1:** Data in the "Typ" column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.
 - 2: Represents input to the system clock prescaler. PLL dividers and postscalers must still be configured so that the system clock frequency does not exceed the maximum frequency shown in Figure 33-1.
 - 3: Instruction cycle period (TcY) equals two times the input oscillator time base period. All specified values are based on characterization data for that particular oscillator type, under standard operating conditions, with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at "Min." values with an external clock applied to the OSCI/CLKI pin. When an external clock input is used, the "Max." cycle time limit is "DC" (no clock) for all devices.
 - **4:** Measurements are taken in EC mode. The CLKO signal is measured on the OSCO pin. CLKO is low for the Q1-Q2 period (1/2 TcY) and high for the Q3-Q4 period (1/2 TcY).

TABLE 33-21: PLL CLOCK TIMING SPECIFICATIONS

AC CHARACTERISTICS				Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{TA} \le +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \le \text{TA} \le +125^{\circ}\text{C}$ for Extended							
Param No.	Symbol	Characteristic	Min	Typ ⁽¹⁾	Max	Units	Conditions				
OS50	FPLLI	USB PLL Input	2	4	4	MHz	ECPLL mode				
		Frequency Range	2	4	4	MHz	HSPLL mode				
			2	4	4	MHz	XTPLL mode				
OS52	TLOCK	USB PLL Start-up Time (Lock Time)	_	_	128	μS					
OS53	DCLK	CLKO Stability (Jitter)	-0.25	_	0.25	%					
OS54	F4xPLL	4x PLL Input Frequency Range	2	_	8	MHz	4xPLL				
OS55	F6xPLL	6x PLL Input Frequency Range	2	_	5	MHz	6xPLL				
OS56	F8xpll	8x PLL Input Frequency Range	2	_	4	MHz	8xPLL				
OS57	Txplllock	PLL Start-up Time (Lock Time)	_	_	24	μS					
OS58	Dxpllclk	PLL CLKO Stability (Jitter)	-2		2	%					

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

TABLE 33-22: INTERNAL RC ACCURACY

AC CHARACTERISTICS		Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{TA} \le +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \le \text{TA} \le +125^{\circ}\text{C}$ for Extended							
Param No.	Characteristic	Min	Тур	Max	Units	Conditions			
F20	FRC Accuracy @ 8 MHz	-1	±0.15	1	%	$2.0V \le VDD \le 3.6V$, $0^{\circ}C \le TA \le +85^{\circ}C$ (Note 1)			
		1.5	_	1.5	%	$2.0V \le V$ DD $\le 3.6V$, -40 °C $\le T$ A ≤ 0 °C			
		-0.20	±0.05	-0.20	%	$2.0V \le V_{DD} \le 3.6V$, $-40^{\circ}C \le T_A \le +85^{\circ}C$, self-tune enabled and locked (Note 2)			
		_	3	5	%	$2.0V \le VDD \le 3.6V$, TA = +125°C			
F21	LPRC @ 31 kHz	-20	_	20	%	VCAP Output Voltage = 1.8V			
F22	OSCTUN Step-Size	_	0.05	_	%/bit				
F23	FRC Self-Tune Lock Time	_	<5	8	ms	(Note 3)			

- **Note 1:** To achieve this accuracy, physical stress applied to the microcontroller package (ex., by flexing the PCB) must be kept to a minimum.
 - 2: Accuracy is measured with respect to the reference source.
 - 3: Time from reference clock stable, and in range, to FRC tuned within range specified by F20 (with self-tune).

TABLE 33-23: RC OSCILLATOR START-UP TIME

AC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{TA} \le +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \le \text{TA} \le +125^{\circ}\text{C}$ for Extended						
Param No.	Symbol	Characteristic	Min	Тур	Max	Units	Conditions		
FR0	TFRC	FRC Oscillator Start-up Time	_	15	_	μS			
FR1	TLPRC	Low-Power RC Oscillator Start-up Time	_	50	_	μS			

FIGURE 33-4: CLKO AND I/O TIMING CHARACTERISTICS

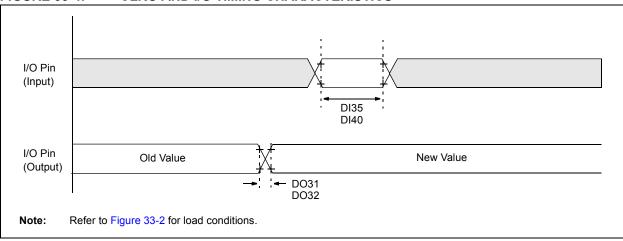


TABLE 33-24: CLKO AND I/O TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{Ta} \le +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \le \text{Ta} \le +125^{\circ}\text{C}$ for Extended					
Param No.	Symbol	Characteristic	Min	Conditions				
DO31	TioR	Port Output Rise Time	_	10	25	ns		
DO32	TioF	Port Output Fall Time	_	10	25	ns		
DI35	TINP	INTx Pin High or Low Time (input)	20	_	_	ns		
DI40	TRBP	CNxx High or Low Time (input)	2	_	_	Tcy		

Note 1: Data in the "Typ" column is at 3.3V, +25°C unless otherwise stated.

TABLE 33-25: RESET AND BROWN-OUT RESET REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{Ta} \le +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \le \text{Ta} \le +125^{\circ}\text{C}$ for Extended						
Param No.	Symbol	Characteristic	Min	Тур	Max	Units	Conditions		
SY10	Тмсь	MCLR Pulse Width (Low)	2	_	_	μS			
SY12	TPOR	Power-on Reset Delay	_	2	_	μS			
SY13	Tıoz	I/O High-Impedance from MCLR Low or Watchdog Timer Reset	Lesser of: (3 Tcy + 2) or 700	_	(3 Tcy + 2)	μS			
SY25	TBOR	Brown-out Reset Pulse Width	1	_	_	μS	VDD ≤ VBOR		
SY45	TRST	Internal State Reset Time	_	50	_	μS			
SY70	Toswu	Deep Sleep Wake-up Time	_	200	_	μS	VCAP fully discharged before wake-up		
SY71	ТРМ	Program Memory Wake-up Time	_	20	_	μS	Sleep wake-up with VREGS = 0		
			_	1	_	μ\$	Sleep wake-up with VREGS = 1		
SY72	TLVR	Low-Voltage Regulator Wake-up Time	_	90	_	μS	Sleep wake-up with VREGS = 0		
			_	70	_	μS	Sleep wake-up with VREGS = 1		

FIGURE 33-5: TIMER1, 2, 3, 4 AND 5 EXTERNAL CLOCK TIMING CHARACTERISTICS

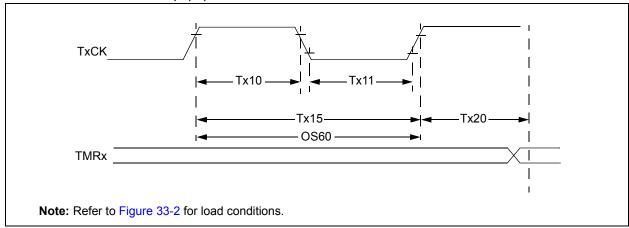


TABLE 33-26: TIMER1 EXTERNAL CLOCK TIMING REQUIREMENTS⁽¹⁾

AC CHARACTERISTICS				Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{TA} \le +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \le \text{TA} \le +125^{\circ}\text{C}$ for Extended						
Param No.	Symbol	Charac	teristic	Min	Тур	Max	Units	Conditions		
TA10	ТтхН	T1CK High Time	Synchronous, No Prescaler	0.5 Tcy + 20		_	ns	Must also meet Parameter TA15		
			Synchronous, with Prescaler	10	_	_	ns			
			Asynchronous	10	_	_	ns			
TA11	ΤτxL	T1CK Low Time	Synchronous, No Prescaler	0.5 Tcy + 20	_	_	ns	Must also meet Parameter TA15		
			Synchronous, with Prescaler	10	-	_	ns			
			Asynchronous	10	_	_	ns			
TA15	ТтхР	T1CK Input Period	Synchronous, No Prescaler	Tcy + 40		_	ns			
			Synchronous, with Prescaler	Greater of: 20 ns or (Tcy + 40)/N		_		N = Prescale Value (1, 8, 64, 256)		
			Asynchronous	20	_	_	ns			
OS60	FT1	SOSC1/T1CK Oscillator Input Frequency Range (oscillator enabled by setting bit, TCS (T1CON<1>))		DC	_	50	kHz			
TA20	TCKEXTMRL	Delay from Exter Edge to Timer In		0.5 TcY		1.5 TCY				

Note 1: Timer1 is a Type A.

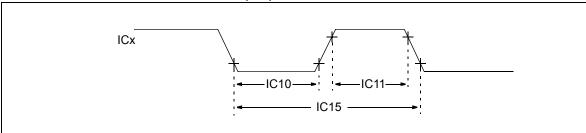
TABLE 33-27: TIMER2 AND TIMER4 EXTERNAL CLOCK TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Ope (unless other Operating terr	wise sta	ated) e -40°C	S ≤ TA ≤ +	to 3.6V -85°C for Industrial 125°C for Extended	
Param No.	Symbol	Charact	Min	Тур	Max	Units	Conditions	
TB10	ТтхН	TxCK High Time	Synchronous, No Prescaler	0.5 Tcy + 20	_	_	ns	Must also meet Parameter TB15
			Synchronous, with Prescaler	10	-	_	ns	
TB11	TTXL	TxCK Low Time	Synchronous, No Prescaler	0.5 Tcy + 20	-	_	ns	Must also meet Parameter TB15
			Synchronous, with Prescaler	10	_		ns	
TB15	ТтхР	TxCK Input Period	Synchronous, No Prescaler	Tcy + 40	_	_	ns	N = Prescale Value (1, 8, 64, 256)
			Synchronous, with Prescaler	Greater of: 20 ns or (Tcy + 40)/N				
TB20	TCKEXTMRL	Delay from Extern Edge to Timer Inc		0.5 TcY	_	1.5 Tcy	_	

TABLE 33-28: TIMER3 AND TIMER5 EXTERNAL CLOCK TIMING REQUIREMENTS

AC CH	ARACTERIS	STICS	Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{TA} \le +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \le \text{TA} \le +125^{\circ}\text{C}$ for Extended					
Param No.	Symbol	Characte	eristic	Min	Тур	Max	Units	Conditions
TC10	ТтхН	TxCK High Time	Synchronous	0.5 Tcy + 20	_	_	ns	Must also meet Parameter TC15
TC11	TTXL	TxCK Low Time	Synchronous	0.5 Tcy + 20	_	_	ns	Must also meet Parameter TC15
TC15	ТтхР	TxCK Input Period	Synchronous, No Prescaler	Tcy + 40	_	_	ns	N = Prescale Value (1, 8, 64, 256)
			Synchronous, with Prescaler	Greater of: 20 ns or (Tcy + 40)/N				
TC20	TCKEXTMRL	Delay from External TxCK Clock Edge to Timer Increment		0.5 TcY	_	1.5 TcY	_	

FIGURE 33-6: INPUT CAPTURE x (ICx) TIMING CHARACTERISTICS



Note: Refer to Figure 33-2 for load conditions.

TABLE 33-29: INPUT CAPTURE x TIMING REQUIREMENTS

AC CHAPACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{TA} \le +85^{\circ}\text{C}$ for Industrict $-40^{\circ}\text{C} \le \text{TA} \le +125^{\circ}\text{C}$ for Extenditions: 2.0V to 3.6V				
Param No.	Symbol Characteristic '/		Min	Max	Units	Conditions	
IC10	TccL	ICx Input Low Time	No Prescaler	0.5 Tcy + 20		ns	
			With Prescaler	10	_	ns	
IC11	TccH	ICx Input High	No Prescaler	0.5 Tcy + 20	_	ns	
Time With Prescaler		10		ns			
IC15	IC15 TccP ICx Input Period		(Tcy + 40)/N		ns	N = Prescale Value (1, 4, 16)	

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

FIGURE 33-7: OUTPUT COMPARE x MODULE (OCx) TIMING CHARACTERISTICS

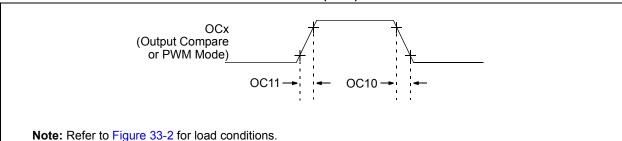


TABLE 33-30: OUTPUT COMPARE x MODULE TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{TA} \le +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \le \text{TA} \le +125^{\circ}\text{C}$ for Extended					
Param No.	Symbol	Characteristic ⁽¹⁾	Min Typ Max Units Conditions					
OC10	TccF	OCx Output Fall Time	_	_	_	ns	See Parameter DO32	
OC11	TccR	OCx Output Rise Time	ns See Parameter DO31					

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

FIGURE 33-8: OCx/PWM MODULE TIMING CHARACTERISTICS

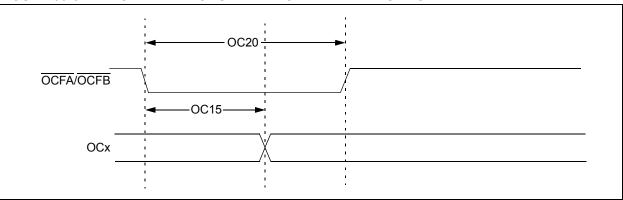


TABLE 33-31: SIMPLE OCx/PWM MODE TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated)					
Param No.	Symbol	Characteristic ⁽¹⁾	Min Typ Max Units Conditions					
OC15	TFD	Fault Input to PWM I/O Change	_	_	50	ns		
OC20	TFLT	Fault Input Pulse Width	50	_	_	ns		

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

FIGURE 33-9: SPIX MODULE MASTER MODE (CKE = 0) TIMING CHARACTERISTICS

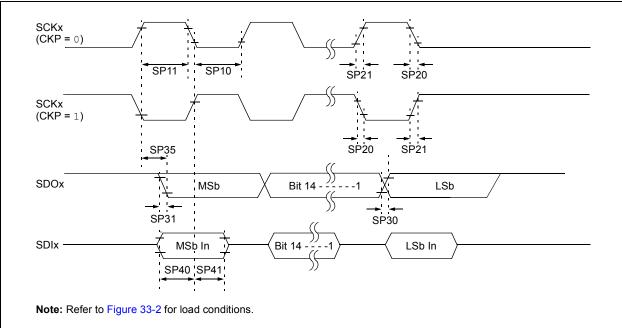


TABLE 33-32: SPIx MASTER MODE (CKE = 0) TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{TA} \le +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \le \text{TA} \le +125^{\circ}\text{C}$ for Extended					
Param No.	Symbol	Characteristic ⁽¹⁾	Min	Conditions				
SP10	TscL	SCKx Output Low Time	Tcy/2	_	_	ns	(Note 3)	
SP11	TscH	SCKx Output High Time	Tcy/2	_	_	ns	(Note 3)	
SP20	TscF	SCKx Output Fall Time	_	_	_	ns	See Parameter DO32 (Note 4)	
SP21	TscR	SCKx Output Rise Time	_			ns	See Parameter DO31 (Note 4)	
SP30	TdoF	SDOx Data Output Fall Time	_		_	ns	See Parameter DO32 (Note 4)	
SP31	TdoR	SDOx Data Output Rise Time	_	_	_	ns	See Parameter DO31 (Note 4)	
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid After SCKx Edge	_	6	20	ns		
SP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	23	_	_	ns		
SP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	30		_	ns		

- Note 1: These parameters are characterized but not tested in manufacturing.
 - 2: Data in "Typ" column is at 3.3V, +25°C unless otherwise stated.
 - **3:** The minimum clock period for SCKx is 100 ns. Therefore, the clock generated in Master mode must not violate this specification.
 - 4: Assumes 50 pF load on all SPIx pins.

SP36 SCKx (CKP = 0)SP21 SP20 SP11 SP10 SCKx(CKP = 1)SP35 SP21 SP20 SDOx Bit 14 LSb MSb SP30, SP31 ► SP40 SDIx-MSb In Bit 14 LSb In SP41 Note: Refer to Figure 33-2 for load conditions.

FIGURE 33-10: SPIX MODULE MASTER MODE (CKE = 1) TIMING CHARACTERISTICS

TABLE 33-33: SPIX MODULE MASTER MODE (CKE = 1) TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard (unless of Operating	therwise	stated) ture -40°	$^{\circ}C \leq TA \leq$	V to 3.6V +85°C for Industrial +125°C for Extended	
Param No.	Symbol	Characteristic ⁽¹⁾	Min Typ ⁽²⁾ Max Units Condition					
SP10	TscL	SCKx Output Low Time(3)	Tcy/2	_	_	ns		
SP11	TscH	SCKx Output High Time ⁽³⁾	Tcy/2	_	_	ns		
SP20	TscF	SCKx Output Fall Time(4)	_	_	_	ns	See Parameter DO32	
SP21	TscR	SCKx Output Rise Time(4)	_	_	_	ns	See Parameter DO31	
SP30	TdoF	SDOx Data Output Fall Time ⁽⁴⁾	ı	_		ns	See Parameter DO32	
SP31	TdoR	SDOx Data Output Rise Time ⁽⁴⁾				ns	See Parameter DO31	
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid After SCKx Edge		6	20	ns		
SP36	TdoV2sc, TdoV2scL	SDOx Data Output Setup to First SCKx Edge	30	_	_	ns		
SP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	23	_	_	ns		
SP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	30	_	_	ns		

- Note 1: These parameters are characterized but not tested in manufacturing.
 - 2: Data in "Typ" column is at 3.3V, +25°C unless otherwise stated.
 - 3: The minimum clock period for SCKx is 100 ns. Therefore, the clock generated in Master mode must not violate this specification.
 - 4: Assumes 50 pF load on all SPIx pins.

SSx SP52 SP50 SCKx (CKP = 0) SP71 SP70 SP72 SP73 SCKx (CKP = 1) SP73 SP72 SP35 SDOx MSb Bit 14 LSb SP30, SP31 SP51 SDIx Bit 14 LSb In MSb In _SP41 SP40 Note: Refer to Figure 33-2 for load conditions.

FIGURE 33-11: SPIX MODULE SLAVE MODE (CKE = 0) TIMING CHARACTERISTICS

TABLE 33-34: SPIX MODULE SLAVE MODE (CKE = 0) TIMING REQUIREMENTS

AC CH	ARACTERI	Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{TA} \le +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \le \text{TA} \le +125^{\circ}\text{C}$ for Extended					
Param No.	Symbol	Characteristic ⁽¹⁾	Min Typ ⁽²⁾ Max Units Conditi				
SP70	TscL	SCKx Input Low Time	30	_	_	ns	
SP71	TscH	SCKx Input High Time	30			ns	
SP72	TscF	SCKx Input Fall Time ⁽³⁾	_	10	25	ns	
SP73	TscR	SCKx Input Rise Time ⁽³⁾	_	10	25	ns	
SP30	TdoF	SDOx Data Output Fall Time ⁽³⁾	_		_	ns	See Parameter DO32
SP31	TdoR	SDOx Data Output Rise Time ⁽³⁾	_		l	ns	See Parameter DO31
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid After SCKx Edge			30	ns	
SP40		Setup Time of SDIx Data Input to SCKx Edge	20	1	1	ns	
SP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	20		l	ns	
SP50	TssL2scH, TssL2scL	SSx ↓ to SCKx ↑ or SCKx Input	120		_	ns	
SP51	TssH2doZ	SSx ↑ to SDOx Output High-Impedance(3)	10	_	50	ns	
SP52	TscH2ssH, TscL2ssH	SSx After SCKx Edge	1.5 Tcy + 40	_	_	ns	

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

2: Data in "Typ" column is at 3.3V, +25°C unless otherwise stated.

3: Assumes 50 pF load on all SPIx pins.

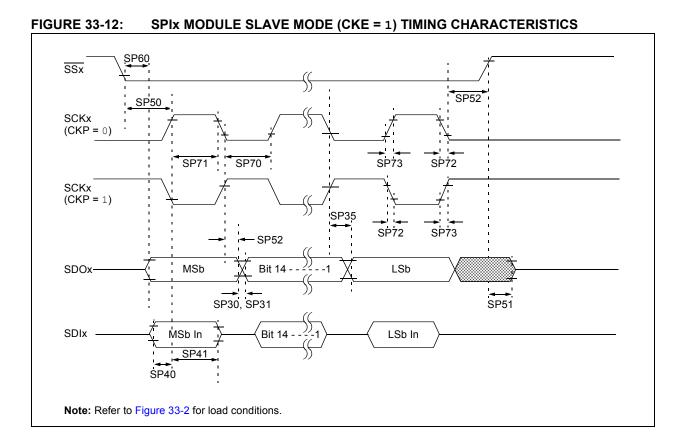


TABLE 33-35: SPIx MODULE SLAVE MODE (CKE = 1) TIMING REQUIREMENTS

AC CHA	RACTERIS'	TICS	Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{Ta} \le +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \le \text{Ta} \le +125^{\circ}\text{C}$ for Extended					
Param No.	Symbol	Characteristic ⁽¹⁾	Min	Min Typ ⁽²⁾ Max Units Cond				
SP70	TscL	SCKx Input Low Time	30		_	ns		
SP71	TscH	SCKx Input High Time	30			ns		
SP72	TscF	SCKx Input Fall Time ⁽³⁾	_	10	25	ns		
SP73	TscR	SCKx Input Rise Time ⁽³⁾	_	10	25	ns		
SP30	TdoF	SDOx Data Output Fall Time ⁽³⁾	_			ns	See Parameter DO32	
SP31	TdoR	SDOx Data Output Rise Time(3)	_		_	ns	See Parameter DO31	
SP35		SDOx Data Output Valid After SCKx Edge	_	1	30	ns		
SP40		Setup Time of SDIx Data Input to SCKx Edge	20			ns		
SP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	20	1		ns		
SP50		$\overline{\rm SSx} \downarrow {\rm to} \; {\rm SCKx} \downarrow {\rm or} \; {\rm SCKx} \uparrow {\rm Input}$	120	1	1	ns		
SP51	TssH2doZ	SSx ↑ to SDOx Output High-Impedance ⁽⁴⁾	10		50	ns		
SP52	TscH2ssH TscL2ssH	SSx ↑ After SCKx Edge	1.5 Tcy + 40		_	ns		
SP60	TssL2doV	SDOx Data Output Valid After SSx Edge	_		50	ns		

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

^{2:} Data in "Typ" column is at 3.3V, +25°C unless otherwise stated.

^{3:} The minimum clock period for SCKx is 100 ns. Therefore, the clock generated in Master mode must not violate this specification.

^{4:} Assumes 50 pF load on all SPIx pins.

FIGURE 33-13: I²C™ BUS START/STOP BITS TIMING CHARACTERISTICS (MASTER MODE)

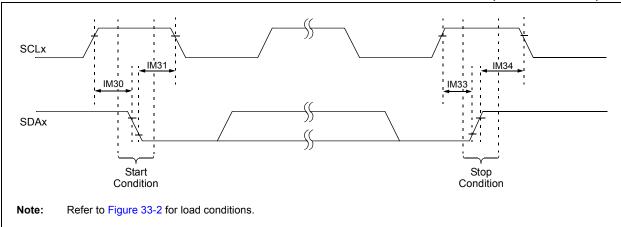


TABLE 33-36: I²C™ BUS START/STOP BIT TIMING REQUIREMENTS (MASTER MODE)

AC CHA	ARACTER	ISTICS		Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated)				
Param No.	Symbol	Charac	teristic	Min ⁽¹⁾	Max	Units	Conditions	
IM30	Tsu:sta	Start Condition	100 kHz mode	Tcy (BRG + 1)	_	μS	Only relevant for	
		Setup Time	400 kHz mode	Tcy (BRG + 1)	_	μS	Repeated Start	
			1 MHz mode ⁽²⁾	Tcy (BRG + 1)	_	μS	condition	
IM31	THD:STA	Start Condition	100 kHz mode	Tcy (BRG + 1)	_	μS	After this period, the	
		Hold Time	400 kHz mode	Tcy (BRG + 1)	_	μS	first clock pulse is	
			1 MHz mode ⁽²⁾	Tcy (BRG + 1)	_	μS	generated	
IM33	Tsu:sto	Stop Condition	100 kHz mode	Tcy (BRG + 1)	_	μS		
		Setup Time	400 kHz mode	Tcy (BRG + 1)	_	μS		
			1 MHz mode ⁽²⁾	Tcy (BRG + 1)	_	μS		
IM34	THD:STO	Stop Condition	100 kHz mode	Tcy (BRG + 1)	_	ns		
		Hold Time	400 kHz mode	Tcy (BRG + 1)	_	ns		
			1 MHz mode ⁽²⁾	Tcy (BRG + 1)	_	ns		

Note 1: BRG is the value of the I²C Baud Rate Generator. Refer to Section 17.2 "Setting Baud Rate When Operating as a Bus Master" for details.

2: Maximum Pin Capacitance = 10 pF for all I²C pins (for 1 MHz mode only).

FIGURE 33-14: I²C™ BUS DATA TIMING CHARACTERISTICS (MASTER MODE)

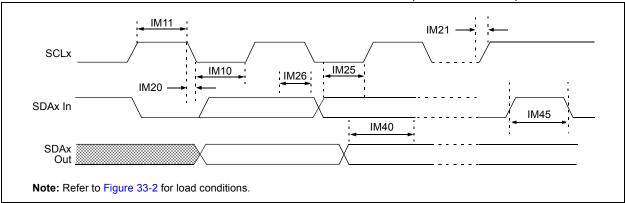


TABLE 33-37: I²C™ BUS DATA TIMING REQUIREMENTS (MASTER MODE)

AC CHA	ARACTERIS	STICS		Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{Ta} \le +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \le \text{Ta} \le +125^{\circ}\text{C}$ for Extended					
Param No.	Symbol	Charac	teristic	Min ⁽¹⁾	Max	Units	Conditions		
IM10	TLO:SCL	Clock Low Time	100 kHz mode	Tcy (BRG + 1)	_	μS			
			400 kHz mode	Tcy (BRG + 1)	_	μS			
			1 MHz mode ⁽²⁾	Tcy (BRG + 1)	_	μS			
IM11	THI:SCL	Clock High Time	100 kHz mode	Tcy (BRG + 1)	_	μS			
			400 kHz mode	Tcy (BRG + 1)	_	μS			
			1 MHz mode ⁽²⁾	Tcy (BRG + 1)	_	μS			
IM20	TF:SCL	SDAx and SCLx	100 kHz mode	_	300	ns	CB is specified to be		
		Fall Time	400 kHz mode	20 + 0.1 CB	300	ns	from 10 to 400 pF		
			1 MHz mode ⁽²⁾	_	100	ns			
IM21	TR:SCL	SDAx and SCLx	100 kHz mode	_	1000	ns	CB is specified to be		
		Rise Time	400 kHz mode	20 + 0.1 CB	300	ns	from 10 to 400 pF		
			1 MHz mode ⁽²⁾	_	300	ns			
IM25	TSU:DAT	Data Input	100 kHz mode	250		ns			
		Setup Time	400 kHz mode	100	_	ns			
			1 MHz mode ⁽²⁾	40	_	ns			
IM26	THD:DAT	Data Input	100 kHz mode	0	_	ns			
		Hold Time	400 kHz mode	0	0.9	μS			
			1 MHz mode ⁽²⁾	0.2	_	ns			
IM40	TAA:SCL	Output Valid	100 kHz mode	_	3500	ns			
		From Clock	400 kHz mode	_	1000	ns			
			1 MHz mode ⁽²⁾	_	400	ns			
IM45	TBF:SDA	Bus Free Time	100 kHz mode	4.7	_	μS	Time the bus must be		
			400 kHz mode	1.3		μS	free before a new		
			1 MHz mode ⁽²⁾	0.5	_	μS	transmission can start		
IM50	Св	Bus Capacitive L	oading	_	400	pF			

Note 1: BRG is the value of the I²C Baud Rate Generator. Refer to Section 17.2 "Setting Baud Rate When Operating as a Bus Master" for details.

2: Maximum Pin Capacitance = 10 pF for all I²C pins (for 1 MHz mode only).

FIGURE 33-15: I²C™ BUS START/STOP BITS TIMING CHARACTERISTICS (SLAVE MODE)

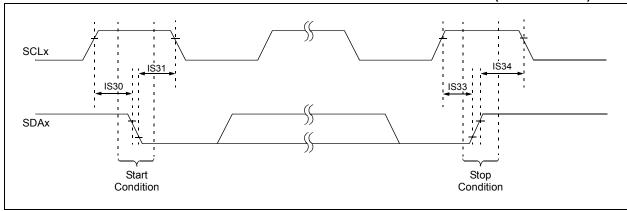


TABLE 33-38: I²C™ BUS START/STOP BITS TIMING REQUIREMENTS (SLAVE MODE)

AC CHA	ARACTER	ISTICS		Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{Ta} \le +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \le \text{Ta} \le +125^{\circ}\text{C}$ for Extended				
Param No.	Symbol	Chara	cteristic	Min	Max	Units	Conditions	
IS30	Tsu:sta	Start Condition	100 kHz mode	4.7	T —	μS	Only relevant for Repeated	
		Setup Time	400 kHz mode	0.6	_	μS	Start condition	
			1 MHz mode ⁽¹⁾	0.25	_	μS		
IS31	THD:STA	Start Condition Hold Time	100 kHz mode	4.0	_	μS	After this period, the first clock	
			400 kHz mode	0.6	_	μS	pulse is generated	
			1 MHz mode ⁽¹⁾	0.25	_	μS		
IS33	Tsu:sto	Stop Condition	100 kHz mode	4.7	_	μS		
		Setup Time	400 kHz mode	0.6	_	μS		
			1 MHz mode ⁽¹⁾	0.6	_	μS		
IS34	THD:STO	Stop Condition	100 kHz mode	4000	_	ns		
		Hold Time	400 kHz mode	600	_	ns		
			1 MHz mode ⁽¹⁾	250	_	ns		

Note 1: Maximum Pin Capacitance = 10 pF for all I^2 C pins (for 1 MHz mode only).

FIGURE 33-16: I²C™ BUS DATA TIMING CHARACTERISTICS (SLAVE MODE)

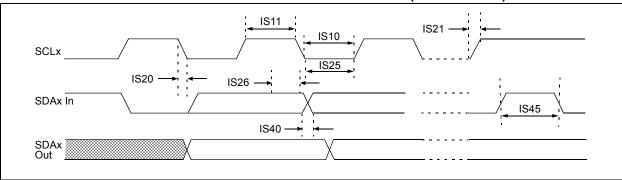


TABLE 33-39: I²C™ BUS DATA TIMING REQUIREMENTS (SLAVE MODE)

AC CHADACTEDISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{TA} \le +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \le \text{TA} \le +125^{\circ}\text{C}$ for Extended				
Param No.	Symbol	Charact	eristic	Min	Max	Units	Conditions
IS10	TLO:SCL	Clock Low Time	100 kHz mode	4.7	_	μS	Device must operate at a minimum of 1.5 MHz
			400 kHz mode	1.3	_	μS	Device must operate at a minimum of 10 MHz
			1 MHz mode ⁽¹⁾	0.5	_	μS	
IS11	THI:SCL	Clock High Time	100 kHz mode	4.0	_	μS	Device must operate at a minimum of 1.5 MHz
			400 kHz mode	0.6	_	μS	Device must operate at a minimum of 10 MHz
			1 MHz mode ⁽¹⁾	0.5	_	μS	
IS20	TF:SCL	SDAx and SCLx	100 kHz mode	_	300	ns	CB is specified to be from
		Fall Time	400 kHz mode	20 + 0.1 CB	300	ns	10 to 400 pF
			1 MHz mode ⁽¹⁾	_	100	ns	
IS21	TR:SCL	SDAx and SCLx	100 kHz mode	_	1000	ns	CB is specified to be from
		Rise Time	400 kHz mode	20 + 0.1 CB	300	ns	10 to 400 pF
			1 MHz mode ⁽¹⁾	_	300	ns	
IS25	TSU:DAT	Data Input	100 kHz mode	250	_	ns	
		Setup Time	400 kHz mode	100		ns	
			1 MHz mode ⁽¹⁾	100		ns	
IS26	THD:DAT	Data Input	100 kHz mode	0		ns	
		Hold Time	400 kHz mode	0	0.9	μS	
			1 MHz mode ⁽¹⁾	0	0.3	μS	
IS40	TAA:SCL	Output Valid From	100 kHz mode	0	3500	ns	
		Clock	400 kHz mode	0	1000	ns	
			1 MHz mode ⁽¹⁾	0	350	ns	
IS45	TBF:SDA	Bus Free Time	100 kHz mode	4.7	_	μS	Time the bus must be free
			400 kHz mode	1.3	_	μS	before a new transmission can start
			1 MHz mode ⁽¹⁾	0.5	_	μS	our start
IS50	Св	Bus Capacitive Lo	ading	_	400	pF	

Note 1: Maximum Pin Capacitance = 10 pF for all I²C pins (for 1 MHz mode only).

TABLE 33-40: A/D MODULE SPECIFICATIONS

AC CHA	ARACTERI	STICS	(unless oth	erwise s	erating Conditions: 2.0V to 3.6V wise stated) perature $-40^{\circ}\text{C} \le \text{TA} \le +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \le \text{TA} \le +125^{\circ}\text{C}$ for Extended			
Param No.	Symbol	Characteristic	Min.	Тур	Max.	Units	Conditions	
			Device S	Supply				
AD01	AVDD	Module VDD Supply	Greater of: VDD – 0.3 or 2.2	_	Lesser of: VDD + 0.3 or 3.6	V		
AD02	AVss	Module Vss Supply	Vss - 0.3	_	Vss + 0.3	V		
			Reference	e Inputs	i			
AD05	VREFH	Reference Voltage High	AVss + 1.7	_	AVDD	V		
AD06	VREFL	Reference Voltage Low	AVss	_	AVDD - 1.7	V		
AD07	VREF	Absolute Reference Voltage	AVss - 0.3	_	AVDD + 0.3	V		
Analog Input								
AD10	VINH-VINL	Full-Scale Input Span	VREFL	_	VREFH	V	(Note 2)	
AD11	VIN	Absolute Input Voltage	AVss - 0.3	_	AVDD + 0.3	V		
AD12	VINL	Absolute VINL Input Voltage	AVss - 0.3	_	AVDD/3	V		
AD13		Leakage Current	_	±1.0	±610	nA	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V, Source Impedance = $2.5 \text{ k}\Omega$	
AD17	Rin	Recommended Impedance of Analog Voltage Source	_	_	2.5K	Ω	10-bit	
			A/D Acc	uracy				
AD20B	Nr	Resolution	_	12	_	bits		
AD21B	INL	Integral Nonlinearity	_	±1	<±2	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V	
AD22B	DNL	Differential Nonlinearity	_	_	<±1	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V	
AD23B	GERR	Gain Error	_	±1	±3	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V	
AD24B	EOFF	Offset Error	_	±1	±2	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V	
AD25B		Monotonicity ⁽¹⁾	_		_		Guaranteed	

Note 1: The A/D conversion result never decreases with an increase in the input voltage and has no missing codes.

^{2:} Measurements are taken with the external VREF+ and VREF- used as the A/D voltage reference.

TABLE 33-41: A/D CONVERSION TIMING REQUIREMENTS⁽¹⁾

AC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{TA} \le +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \le \text{TA} \le +125^{\circ}\text{C}$ for Extended					
Param No.	Symbol	Characteristic	Min. Typ Max. Units Cor			Conditions		
Clock Parameters								
AD50	TAD	A/D Clock Period	75	_	_	ns	Tcy = 75 ns, AD1CON3 in default state	
AD51	trc	A/D Internal RC Oscillator Period	_	250	_	ns		
		Con	version R	ate				
AD55	tconv	Conversion Time	_	14	_	TAD		
AD56	FCNV	Throughput Rate	_	_	200	ksps	AVDD > 2.7V	
AD57	tsamp	Sample Time	_	1	_	TAD		
		Cloc	k Paramet	ters				
AD61	tpss	Sample Start Delay from Setting Sample bit (SAMP)	2	_	3	TAD		

Note 1: Because the sample caps will eventually lose charge, clock rates below 10 kHz can affect linearity performance, especially at elevated temperatures.

34.0 PACKAGING INFORMATION

34.1 Package Marking Information

28-Lead QFN-S



Example



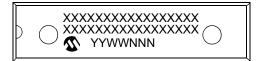
28-Lead SOIC (.300")



Example



28-Lead SPDIP



Example



28-Lead SSOP



Example



 Legend:
 XX...X
 Customer-specific information

 Y
 Year code (last digit of calendar year)

 YY
 Year code (last 2 digits of calendar year)

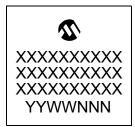
 WW
 Week code (week of January 1 is week '01')

 NNN
 Alphanumeric traceability code

Note: In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line, thus limiting the number of available characters for customer-specific information.

34.1 Package Marking Information (Continued)

44-Lead QFN



Example



PIC24FJ128 GB204

1510017

44-Lead TQFP



Example



MICROCHIP

PIC24FJ128 GB204

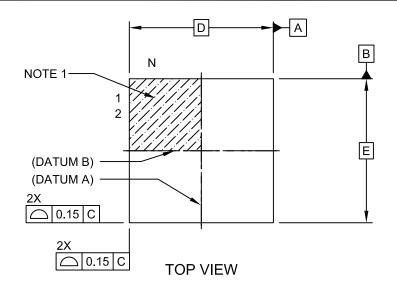
1510017

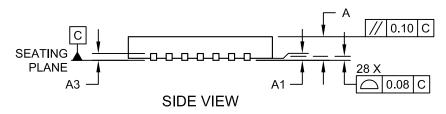
34.2 Package Details

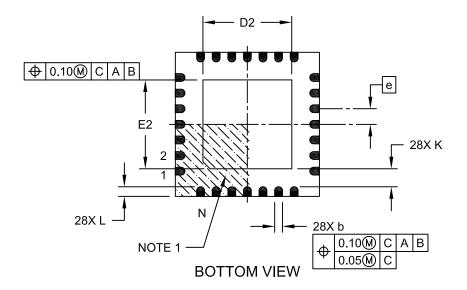
The following sections give the technical details of the packages.

28-Lead Plastic Quad Flat, No Lead Package (MM) - 6x6x0.9mm Body [QFN-S] With 0.40 mm Terminal Length

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



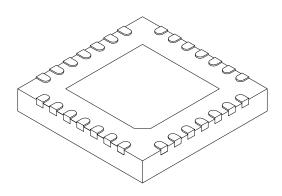




Microchip Technology Drawing C04-124C Sheet 1 of 2

28-Lead Plastic Quad Flat, No Lead Package (MM) - 6x6x0.9mm Body [QFN-S] With 0.40 mm Terminal Length

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



	Units			S
Dimension	Limits	MIN	NOM	MAX
Number of Pins	Ν		28	
Pitch	е		0.65 BSC	
Overall Height	Α	0.80	0.90	1.00
Standoff	A1	0.00	0.02	0.05
Terminal Thickness	A3	0.20 REF		
Overall Width	Е		6.00 BSC	
Exposed Pad Width	E2	3.65	3.70	4.70
Overall Length	D		6.00 BSC	
Exposed Pad Length	D2	3.65	3.70	4.70
Terminal Width	b	0.23	0.30	0.35
Terminal Length	L	0.30	0.40	0.50
Terminal-to-Exposed Pad	K	0.20	-	-

Notes:

- 1. Pin 1 visual index feature may vary, but must be located within the hatched area.
- 2. Package is saw singulated
- 3. Dimensioning and tolerancing per ASME Y14.5M

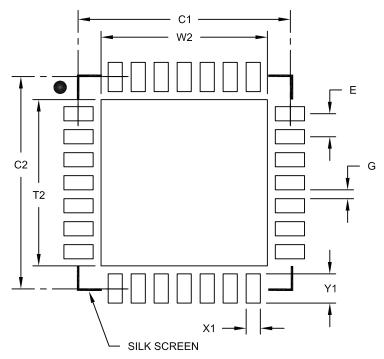
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

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

Microchip Technology Drawing C04-124C Sheet 2 of 2

28-Lead Plastic Quad Flat, No Lead Package (MM) – 6x6x0.9 mm Body [QFN-S] with 0.40 mm Contact Length

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



RECOMMENDED LAND PATTERN

	MILLIMETERS			
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E		0.65 BSC	
Optional Center Pad Width	W2			4.70
Optional Center Pad Length	T2			4.70
Contact Pad Spacing	C1		6.00	
Contact Pad Spacing	C2		6.00	
Contact Pad Width (X28)	X1			0.40
Contact Pad Length (X28)	Y1			0.85
Distance Between Pads	G	0.25		

Notes:

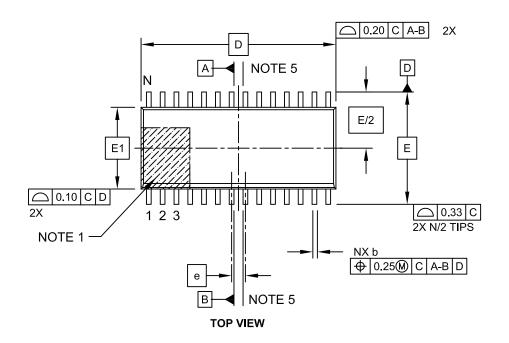
1. Dimensioning and tolerancing per ASME Y14.5M

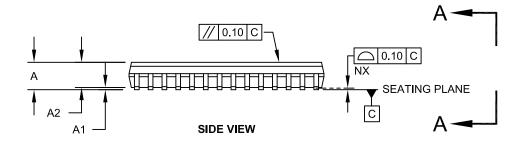
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

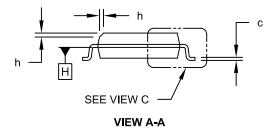
Microchip Technology Drawing No. C04-2124A

28-Lead Plastic Small Outline (SO) - Wide, 7.50 mm Body [SOIC]

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



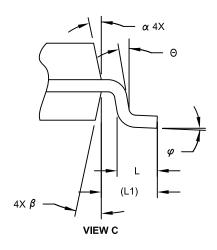


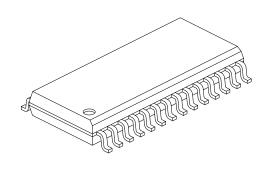


Microchip Technology Drawing C04-052C Sheet 1 of 2

28-Lead Plastic Small Outline (SO) - Wide, 7.50 mm Body [SOIC]

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





	MILLIMETERS			
Dimensior	MIN	NOM	MAX	
Number of Pins	N		28	
Pitch	е		1.27 BSC	
Overall Height	Α	-	-	2.65
Molded Package Thickness	A2	2.05	i	-
Standoff §	A1	0.10	-	0.30
Overall Width	Е		10.30 BSC	
Molded Package Width	E1	7.50 BSC		
Overall Length	D	17.90 BSC		
Chamfer (Optional)	h	0.25	ı	0.75
Foot Length	L	0.40	ı	1.27
Footprint	L1		1.40 REF	
Lead Angle	Θ	0°	i	-
Foot Angle	φ	0° - 8°		
Lead Thickness	С	0.18 - 0.33		
Lead Width	b	0.31	-	0.51
Mold Draft Angle Top	α	5°	-	15°
Mold Draft Angle Bottom	β	5°	-	15°

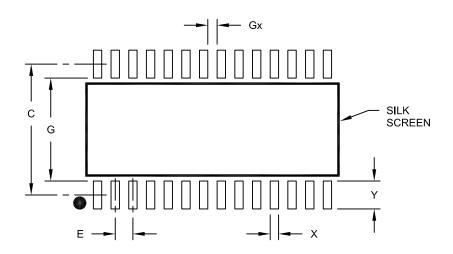
Notes:

- 1. Pin 1 visual index feature may vary, but must be located within the hatched area.
- 2. § Significant Characteristic
- Dimension D does not include mold flash, protrusions or gate burrs, which shall not exceed 0.15 mm per end. Dimension E1 does not include interlead flash or protrusion, which shall not exceed 0.25 mm per side.
- 4. 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.
- 5. Datums A & B to be determined at Datum H.

Microchip Technology Drawing C04-052C Sheet 2 of 2

28-Lead Plastic Small Outline (SO) - Wide, 7.50 mm Body [SOIC]

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



RECOMMENDED LAND PATTERN

	MILLIMETERS			
Dimension Limits		MIN	NOM	MAX
Contact Pitch	Е	1.27 BSC		
Contact Pad Spacing	С		9.40	
Contact Pad Width (X28)	Х			0.60
Contact Pad Length (X28)	Υ			2.00
Distance Between Pads	Gx	0.67		
Distance Between Pads	G	7.40		

Notes:

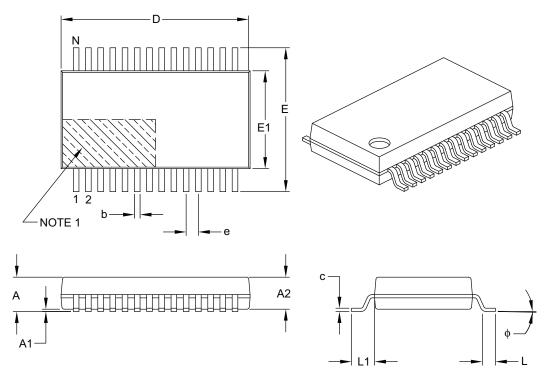
1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2052A

28-Lead Plastic Shrink Small Outline (SS) - 5.30 mm Body [SSOP]

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



	Units			MILLIMETERS			
Dimension	n Limits	MIN	NOM	MAX			
Number of Pins	N		28				
Pitch	е		0.65 BSC				
Overall Height	Α	_	_	2.00			
Molded Package Thickness	A2	1.65	1.75	1.85			
Standoff	A1	0.05	_	_			
Overall Width	E	7.40	7.80	8.20			
Molded Package Width	E1	5.00	5.30	5.60			
Overall Length	D	9.90	10.20	10.50			
Foot Length	L	0.55	0.75	0.95			
Footprint	L1		1.25 REF				
Lead Thickness	С	0.09	_	0.25			
Foot Angle	ф	0°	4°	8°			
Lead Width	b	0.22	_	0.38			

Notes:

- 1. Pin 1 visual index feature may vary, but must be located within the hatched area.
- 2. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.20 mm per side.
- 3. Dimensioning and tolerancing per ASME Y14.5M.

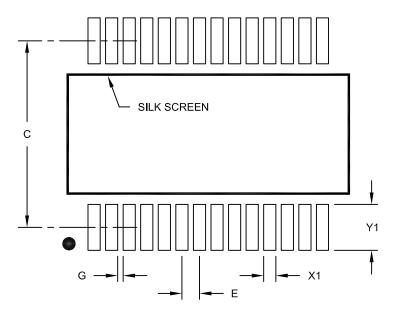
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

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

Microchip Technology Drawing C04-073B

28-Lead Plastic Shrink Small Outline (SS) - 5.30 mm Body [SSOP]

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



RECOMMENDED LAND PATTERN

	MILLIMETERS				
Dimension Limits		MIN	NOM	MAX	
Contact Pitch	Е	0.65 BSC			
Contact Pad Spacing	С		7.20		
Contact Pad Width (X28)	X1			0.45	
Contact Pad Length (X28)	Y1			1.75	
Distance Between Pads	G	0.20			

Notes:

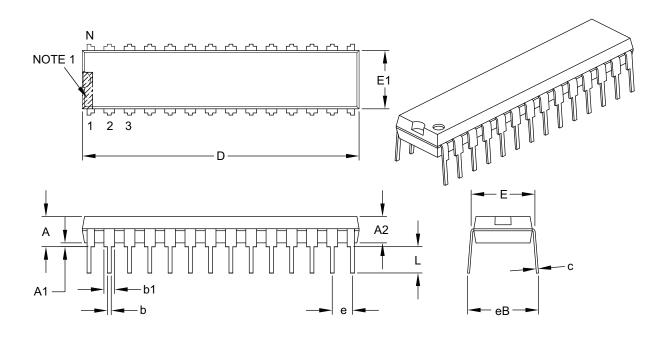
1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2073A

28-Lead Skinny Plastic Dual In-Line (SP) - 300 mil Body [SPDIP]

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



	Units		INCHES	
	Dimension Limits	MIN	NOM	MAX
Number of Pins	N		28	
Pitch	е		.100 BSC	
Top to Seating Plane	A	-	_	.200
Molded Package Thickness	A2	.120	.135	.150
Base to Seating Plane	A1	.015	_	_
Shoulder to Shoulder Width	E	.290	.310	.335
Molded Package Width	E1	.240	.285	.295
Overall Length	D	1.345	1.365	1.400
Tip to Seating Plane	L	.110	.130	.150
Lead Thickness	С	.008	.010	.015
Upper Lead Width	b1	.040	.050	.070
Lower Lead Width	b	.014	.018	.022
Overall Row Spacing §	eB	_	_	.430

Notes:

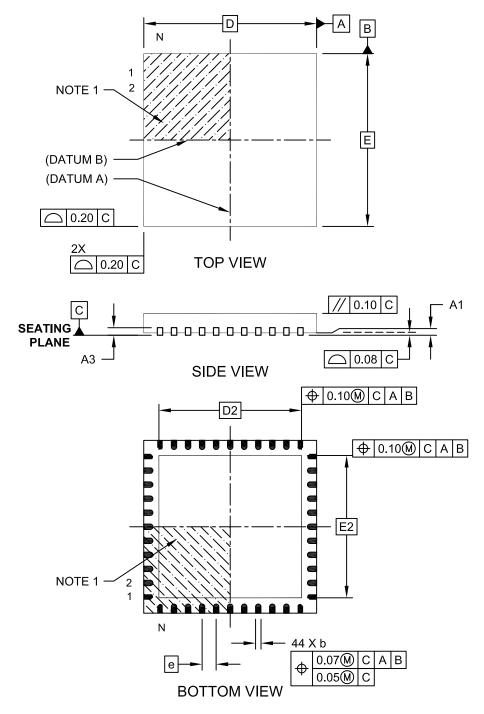
- 1. Pin 1 visual index feature may vary, but must be located within the hatched area.
- 2. § Significant Characteristic.
- 3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" per side.
- 4. Dimensioning and tolerancing per ASME Y14.5M.

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

Microchip Technology Drawing C04-070B

44-Lead Plastic Quad Flat, No Lead Package (ML) - 8x8 mm Body [QFN]

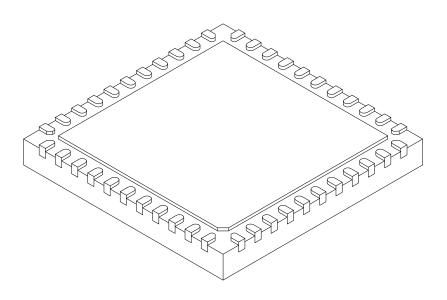
Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Microchip Technology Drawing C04-103C Sheet 1 of 2

44-Lead Plastic Quad Flat, No Lead Package (ML) - 8x8 mm Body [QFN]

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



	Units			S	
Dimension	Limits	MIN	NOM	MAX	
Number of Pins	N		44		
Pitch	е		0.65 BSC		
Overall Height	Α	0.80	0.90	1.00	
Standoff	A1	0.00	0.02	0.05	
Terminal Thickness	A3		0.20 REF		
Overall Width	E		8.00 BSC		
Exposed Pad Width	E2	6.25	6.45	6.60	
Overall Length	D		8.00 BSC		
Exposed Pad Length	D2	6.25	6.45	6.60	
Terminal Width	b	0.20	0.30	0.35	
Terminal Length	Ĺ	0.30	0.40	0.50	
Terminal-to-Exposed-Pad	K	0.20	-	-	

Notes:

- 1. Pin 1 visual index feature may vary, but must be located within the hatched area.
- 2. Package is saw singulated
- 3. Dimensioning and tolerancing per ASME Y14.5M

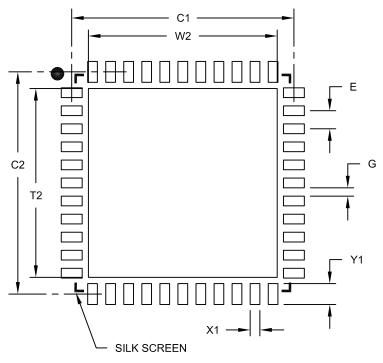
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

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

Microchip Technology Drawing C04-103C Sheet 2 of 2

44-Lead Plastic Quad Flat, No Lead Package (ML) - 8x8 mm Body [QFN]

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



RECOMMENDED LAND PATTERN

	MILLIMETERS			
Dimension Limits		MIN	NOM	MAX
Contact Pitch	Е		0.65 BSC	
Optional Center Pad Width	W2			6.60
Optional Center Pad Length	T2			6.60
Contact Pad Spacing	C1		8.00	
Contact Pad Spacing	C2		8.00	
Contact Pad Width (X44)	X1			0.35
Contact Pad Length (X44)	Y1			0.85
Distance Between Pads	G	0.25		

Notes:

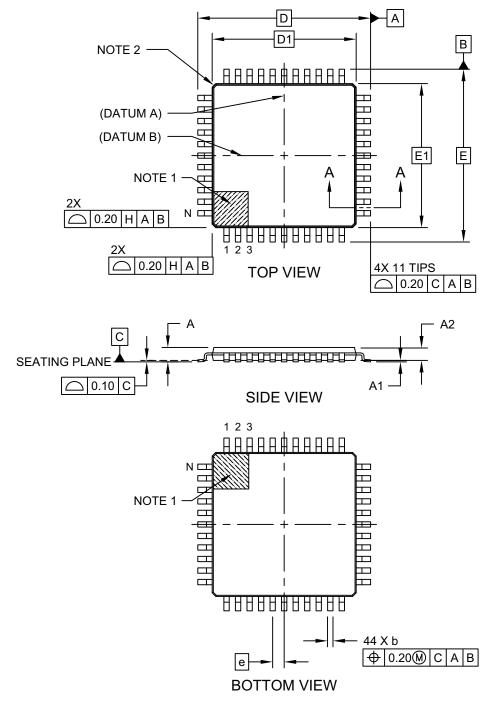
1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2103B

44-Lead Plastic Thin Quad Flatpack (PT) - 10x10x1.0 mm Body [TQFP]

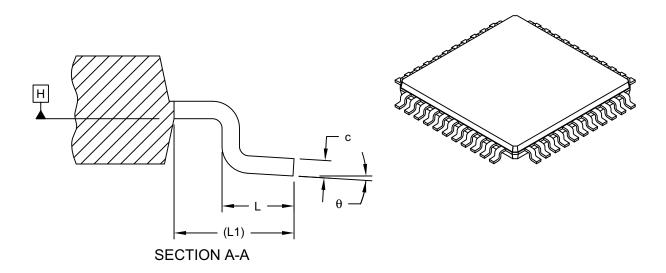
Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Microchip Technology Drawing C04-076C Sheet 1 of 2

44-Lead Plastic Thin Quad Flatpack (PT) - 10x10x1.0 mm Body [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
Number of Leads	N	44		
Lead Pitch	е	0.80 BSC		
Overall Height	Α		-	1.20
Standoff	A1	0.05	-	0.15
Molded Package Thickness	A2	0.95	1.00	1.05
Overall Width	E	12.00 BSC		
Molded Package Width	E1	10.00 BSC		
Overall Length	D	12.00 BSC		
Molded Package Length	D1	10.00 BSC		
Lead Width	b	0.30	0.37	0.45
Lead Thickness	С	0.09	-	0.20
Lead Length	L	0.45	0.60	0.75
Footprint	L1	1.00 REF		
Foot Angle	θ	0°	3.5°	7°

Notes:

- 1. Pin 1 visual index feature may vary, but must be located within the hatched area.
- 2. Exact shape of each corner is optional.
- 3. Dimensioning and tolerancing per ASME Y14.5M

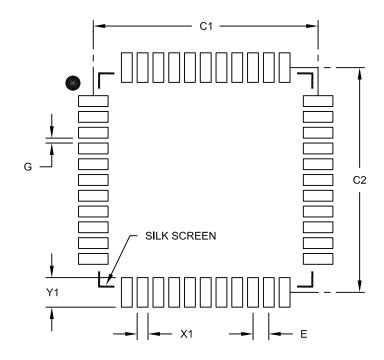
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

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

Microchip Technology Drawing C04-076C Sheet 2 of 2

44-Lead Plastic Thin Quad Flatpack (PT) 10X10X1 mm 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		0.80 BSC	
Contact Pad Spacing	C1		11.40	
Contact Pad Spacing	C2		11.40	
Contact Pad Width (X44)	X1			0.55
Contact Pad Length (X44)	Y1			1.50
Distance Between Pads	G	0.25		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2076B

PICZ4FJ IZOGBZU4 FAIVIIL I					
NOTES:					

APPENDIX A: REVISION HISTORY

Revision A (May 2013)

Original data sheet for the PIC24FJ128GB204 family of devices.

Revision B (May 2014)

This revision incorporates the following updates:

- · Sections:
 - Inserted new bulleted list in "Cryptographic Engine"
 - Updated a unit in "Analog Features"
 - Updated note in Section 16.0 "Serial Peripheral Interface (SPI)", Section 17.0 "Inter-Integrated Circuit™ (I²C™)", Section 18.0 "Universal Asynchronous Receiver Transmitter (UART)", Section 23.0 "Cryptographic Engine" and Section 27.0 "Comparator Voltage Reference"
 - Updated Section 17.3 "Slave Address Masking"
 - Updated Section 23.0 "Cryptographic Engine"
 - Inserted new Section 23.5.6 "Generating a Random Number"
 - Updated Section 30.3.1 "Windowed Operation"
 - Updated packaging information in Section 34.0 "Packaging Information"
- · Registers:
 - Updated Register 8-45, Register 16-1, Register 16-4, Register 17-1, Register 18-1, Register 20-1, Register 23-1 and Register 23-5
 - Updated the title of Register 18-2 and Register 18-4
 - Updated bit 10 register description in Register 23-5
- · Tables:
 - Updated Table 1-3, Table 4-5, Table 4-9, Table 4-10, Table 4-11, Table 4-12, Table 4-13, Table 4-14, Table 4-29, Table 33-1, Table 33-3, Table 33-4, Table 33-5, Table 33-6, Table 33-7, Table 33-8, Table 33-10, Table 33-12, Table 33-13, Table 33-14, Table 33-15, Table 33-16, Table 33-21
 - Added Table 33-26 through Table 33-39
- · Figures:
 - Added Figure 9-1, Figure 33-5, Figure 33-6,
 Figure 33-7, Figure 33-8, Figure 33-9,
 Figure 33-10, Figure 33-11 and Figure 33-12.
- · Examples:
 - Example 22-1

Revision C (March 2015)

This revision incorporates the following updates:

- · Registers:
 - Register 26-1
- · Tables:
 - Table 33-4, Table 33-5, Table 33-6 and Table 33-21
- Package marking examples and package diagrams in Section 34.0 "Packaging Information" were updated

NOTES:

INDEX

A	CTMU Connections and Internal Configuration	
A/D	for Capacitance Measurement	380
Control Registers	CTMU Connections and Internal Configuration	
Extended DMA Operations	for Pulse Delay Generation	381
Operation	CTMU Connections and Internal Configuration	
Transfer Functions	for Time Measurement	381
10-Bit370	Data Access from Program Space	
12-Bit	Address Generation	66
AC Characteristics	Data Signal Modulator	299
A/D Conversion Timing Requirements446	Direct Memory Access (DMA)	
A/D Specifications	EDS Address Generation for Read	62
and Timing Parameters	EDS Address Generation for Write	63
Capacitive Loading on Output Pins	Extended Data Space (EDS)	
CLKO and I/O Timing Requirements	High/Low-Voltage Detect (HLVD) Module	387
External Clock Timing Requirements428	I ² C Module	246
I ² C Bus Data (Master Mode) Requirements	Individual Comparator Configurations,	
I ² C Bus Data (Slave Mode) Requirements	CREF = 0	372
I ² C Bus Start/Stop Bit Timing Requirements	Individual Comparator Configurations,	
(Master Mode)441	CREF = 1, CVREFP = 0	373
I ² C Bus Start/Stop Bits (Slave Mode)	Individual Comparator Configurations,	
Requirements443	CREF = 1, CVREFP = 1	373
Input Capture x Requirements	Input Capture x	211
Internal RC Accuracy429	MCLR Pin Connections	
·	On-Chip Regulator Connections	399
Load Conditions and Requirements for Specifications427	Oscillator Circuit Placement	
•	Output Compare x (16-Bit Mode)	218
Output Compare x Requirements	Output Compare x (Double-Buffered,	
RC Oscillator Start-up Time	16-Bit PWM Mode)	220
	PIC24F CPU Core	30
Reset and Brown-out Reset Requirements	PIC24FJ128GB204 Family (General)	14
Simple OCx/PWM Mode Requirements	PSV Operation Access (Lower Word)	69
SPIx Master Mode (CKE = 0) Requirements	PSV Operation Access (Upper Word)	
SPIx Master Mode (CKE = 1) Requirements	Recommended Minimum Connections	
SPIx Slave Mode (CKE = 0) Requirements	Reset System	85
SPIx Slave Mode (CKE = 1) Requirements	RTCC Module	
Timer1 External Clock Requirements432 Timer2 and Timer4 External Clock	Shared I/O Port Structure	173
Requirements433	Smart Card Subsystem Connection	257
Timer3 and Timer5 External Clock	SPIx Master, Frame Master Connection	
Requirements433	SPIx Master, Frame Slave Connection	
Alternate Interrupt Vector Table (AIVT)91	SPIx Master/Slave Connection	
Assembler	(Enhanced Buffer Modes)	242
MPASM Assembler404	SPIx Master/Slave Connection	
WIFASIW ASSETIBLET404	(Standard Mode)	241
В	SPIx Module (Enhanced Mode)	
Block Diagrams	SPIx Module (Standard Mode)	
10-Bit A/D Converter Analog Input Model	SPIx Slave, Frame Master Connection	
12-Bit A/D Converter Arialog Input Model	SPIx Slave, Frame Slave Connection	243
16-Bit Asynchronous Timer3 and Timer5	System Clock	147
16-Bit Synchronous Timer2 and Timer4	Timer2/3 and Timer4/5 (32-Bit)	
16-Bit Timer1 Module201	Triple Comparator Module	
Accessing Program Space Using	UARTx (Simplified)	254
Table Instructions67	USB Host Interface Example	
Addressing for Table Registers79	USB OTG Bus Power Only	
· ·	USB OTG Dual Power Example	
BDT Mapping for Endpoint Buffering Modes	USB OTG External Pull-up for Full-Speed	
	Device Mode	267
CALL Stack Frame	USB OTG Interface Example	
Comparator Voltage Reference Module	USB OTG Interrupt Funnel	
CPU Programmer's Model	USB OTG Module	
CRC Module	USB OTG Self-Power Only	
CRC Shift Engine Detail	USB PLL	
Cryptographic Engine329	Watchdog Timer (WDT)	
		+00

C		CTMU	
C Compilers		Measuring Capacitance	379
MPLAB XC	404	Measuring Time	
Charge Time Measurement Unit. See CTMU.		Pulse Delay and Generation	381
Code Examples		Customer Change Notification Service	
Basic Clock Switching Sequence	154	Customer Notification Service	
Configuring UART1 Input/Output Functions		Customer Support	473
EDS Read from Program Memory in Assembl		D	
EDS Read in Assembly		_	
EDS Write in Assembly		Data Memory	
Erasing a Program Memory Block (Assembly)	82	Address Space	
Erasing a Program Memory Block (C Language		Extended Data Space (EDS)	
Initiating a Programming Sequence	83	Memory Map	
Loading the Write Buffers	83	Near Data Space	
Port Read/Write in Assembly		SFR SpaceSoftware Stack	
Port Read/Write in C	177	Space Organization, Alignment	
PWRSAV Instruction Syntax	162	Data Signal Modulator (DSM)	
Repeat Sequence	164	Data Signal Modulator. See DSM.	200
Setting the RTCWREN Bit		DC Characteristics	
Single-Word Flash Programming		Comparator Specifications	425
Single-Word Flash Programming (C Language		Comparator Voltage Reference Specifications	
Code Protection		CTMU Current Source Specifications	
Code Segment Protection		Δ Current (BOR, WDT, DSBOR, DSWDT)	
Configuration Options		High/Low-Voltage Detect	
Configuration Register Protection		I/O Pin Input Specifications	
General Segment Protection		I/O Pin Output Specifications	
Comparator Voltage Reference		Idle Current (IDLE)	
Configuring		Internal Voltage Regulator Specifications	
Configuration Bits		Operating Current (IDD)	
Core Features	9	Power-Down Current (IPD)	
CPU		Program Memory	
Arithmetic Logic Unit (ALU)		Temperature and Voltage Specifications	
Control Registers		Thermal Operating Conditions	
Core Registers		Thermal Packaging	
Programmer's Model	29	USB On-The-Go Specifications	
CRC	0.47	VBAT Operating Voltage Specifications	
Interrupt Operation		Development Support	
Polynomials		Device Features	
Setup Examples for 16/32-Bit Polynomials		28-Pin Devices	13
Typical Operation		44-Pin Devices	12
User Interface		Device Programmer	
Cryptographic Engine Control Registers		MPLAB PM3	405
•		Direct Memory Access. See DMA.	
Data Register Spaces Decrypting Data		DMA	
		Channel Trigger Sources	78
Enabling Encrypting Data		Control Registers	
Key Source Selection		Peripheral Module Disable (PMD)	74
AES	3/13	Summary of Operations	72
DES/3DES		Types of Data Transfers	73
Operation Modes		Typical Setup	74
Idle		DMA Controller	10
Sleep		DSM	
Programming		E	
CFGPAGE Configuration Bits	334	_	
Keys		Electrical Characteristics	
Verifying Keys		Absolute Maximum Ratings	
Pseudorandom Number (PRN) Generation		V/F Graph (Industrial)	
Random Number Generation		Enhanced Parallel Master Port (EPMP)	303
Session Keys		Enhanced Parallel Master Port. See EPMP.	
Encrypting	332	EPMP	
Receiving		Key Features	
Testing Key Source Configuration		Memory Addressable in Different Modes	
. Journa 107 Journal Configuration		Pin Descriptions	305

Equations		Input Capture	
16-Bit, 32-Bit CRC Polynomials	346	32-Bit Cascaded Mode	212
A/D Conversion Clock Period	368	Operations	
Baud Rate Reload Calculation		Synchronous and Trigger Modes	
Calculating the PWM Period		Input Capture with Dedicated Timers	211
Calculation for Maximum PWM Resolution	221	Input Voltage Levels for Port or Pin Tolerated	
Estimating USB Transceiver		Description Input	174
Current Consumption		Instruction Set	400
Fractional Divisor for ROTRIMx Bits		Overview	
Relationship Between Device and SPIx Clock Speed	242	Summary Symbols Used in Opcode Descriptions	
UARTx Baud Rate with BRGH = 0		Interfacing Program and Data Spaces	
UARTX Baud Rate with BRGH = 1		Inter-Integrated Circuit. See I ² C.	05
Errata		Internet Address	473
Extended Data Space (EDS)		Interrupt Vector Table (IVT)	
• • • •		Interrupts	
F		Control and Status Registers	95
Flash Configuration Word Locations	389	Implemented Vectors	
Flash Configuration Words	36	Reset Sequence	
Flash Program Memory		Setup and Service Procedures	
and Table Instructions	79	Trap Vectors	
Control Registers		Vector Table	
Enhanced ICSP Operation		Peripheral Pin Select (PPS)	
JTAG Operation		Selectable Input Sources	179
Programming Algorithm		1	
Programming Operations		J	
RTSP Operation		JTAG Interface	402
Single-Word Programming	84	K	
G		••	000
-	00	Key Features	389
Getting Started with 16-Bit MCUs	23	L	
Analog/Digital Pins Configuration During ICSP Operations	20	Low-Voltage/Retention Regulator	163 300
Connection Requirements		•	100, 000
External Oscillator Pins		M	
ICSP Programming Pins		Memory Organization	35
Master Clear (MCLR) Pin		Microchip Internet Web Site	
Power Supply Pins		MPLAB Assembler, Linker, Librarian	
Unused I/Os		MPLAB X Integrated Development	
Voltage Regulator Pins		Environment Software	403
		MPLINK Object Linker/MPLIB Object Librarian	404
Н		N	
High/Low-Voltage Detect (HLVD)	387		
High/Low-Voltage Detect. See HLVD.		Near Data Space	38
1		0	
1		•	200
I/O Ports		On-Chip Voltage Regulator	
Analog Port Pins Configuration (ANSx)		POR	
Analog/Digital Function of an I/O Pin		Standby Mode On-The-Go. See OTG.	399
Input Change Notification (ICN)		Oscillator	
Open-Drain Configuration		Clock Switching	152
Parallel (PIO)		Sequence	
Peripheral Pin Select		Configuration Bit Values for Clock Selection	
Pull-ups and Pull-Downs		Control Registers	
Write/Read Timing	174	FRC Self-Tuning	
		Initial Configuration on POR	
Communicating as Master in Single	245	Initial CPU Clocking Scheme	
Master Environment		On-Chip PLL	
Reserved Addresses		Reference Clock Output	
Setting Baud Rate as Bus Master		USB Operation	
Slave Address Masking	241	Special Considerations	
In-Circuit Debuggers MPLAB ICD 3	405	opena. Consideration	
PICkit 3			
In-Circuit Emulator System			
MPLAB REAL ICE	405		

Output Compare		Output Compare	45
32-Bit Cascaded Mode	217	Pad Configuration (PADCFG1)	
Compare Mode Operations		Peripheral Module Disable (PMD)	
Synchronous and Trigger Modes		Peripheral Pin Select (PPS)	
Output Compare with Dedicated Timers		PORTA	
		PORTB	
P		PORTC	50
Packaging	447	Real-Time Clock and Calendar (RTCC)	56
Details	449	SPI1	48
Marking	447	SPI2	48
Peripheral Pin Select (PPS)	178	SPI3	49
Available Peripherals and Pins	178	System Control (Clock and Reset)	58
Configuration Control	181	Timers	
Considerations for Use	182	UART	47
Control Registers	183	USB OTG	54
Input Mapping	179	Registers	
Mapping Exceptions	181	AD1CHITL (ADC1 Scan Compare Hit,	
Output Mapping	180	Low Word)	365
Peripheral Priority	178	AD1CHS (ADC1 Sample Select)	
Selectable Output Sources		AD1CON1 (ADC1 Control 1)	
Pinout Description		AD1CON2 (ADC1 Control 2)	
Power-Saving Features		AD1CON3 (ADC1 Control 3)	
Clock Frequency, Clock Switching		AD1CON4 (ADC1 Control 4)	
Doze Mode		AD1CON5 (ADC1 Control 5)	
Instruction-Based Modes		AD1CSSH (ADC1 Input Scan Select,	502
Deep Sleep		High Word)	366
Idle		AD1CSSL (ADC1 Input Scan Select,	500
Sleep		Low Word)	366
Overview of Modes		AD1CTMENL (ADC1 CTMU Enable,	300
VBAT Mode		Low Word)	267
Product Identification System		•	
Program Memory	473	ALCFGRPT (Alarm Configuration)	320
Access Using Table Instructions	67	· ·	224
Address Space		Seconds Value)	
		ALMTHDY (Alarm Month and Day Value)	
Addressing Program Space		ALWDHR (Alarm Weekday and Hours Value)	323
Flash Configuration Words		ANCFG (A/D Band Gap Reference	004
Hard Memory Vectors		Configuration)	
Memory Maps		ANSA (PORTA Analog Function Selection)	
Organization		ANSB (PORTB Analog Function Selection)	
Reading from Program Memory Using EDS		ANSC (PORTC Analog Function Selection)	176
Program Verification		BDnSTAT (Buffer Descriptor n Status Prototype,	
Pulse-Width Modulation (PWM) Mode	219	CPU Mode)	273
Pulse-Width Modulation. See PWM. PWM		BDnSTAT (Buffer Descriptor n Status Prototype, USB Mode)	272
Duty Cycle and Period	220	CFGPAGE (Secure Array Configuration Bits)	340
R		CLKDIV (Clock Divider)	
		CMSTAT (Comparator Status)	375
Real-Time Clock and Calendar (RTCC)	315	CMxCON (Comparator x Control,	
Real-Time Clock and Calendar. See RTCC.		Comparators 1-3)	374
Register Maps		CORCON (CPU Core Control)	
A/D Converter	51	CRCCON1 (CRC Control 1)	348
Analog Configuration		CRCCON2 (CRC Control 2)	349
Comparators		CRCXORH (CRC XOR Polynomial, High Byte)	
CPU Core	39	CRCXORL (CRC XOR Polynomial, Low Byte)	
CRC	57	CRYCONH (Cryptographic Control High)	
Cryptographic Engine	59	CRYCONL (Cryptographic Control Low)	335
CTMU	52	CRYOTP (Cryptographic OTP Page	
Data Signal Modulator (DSM)	56	Program Control)	339
Deep Sleep		CRYSTAT (Cryptographic Status)	
DMA		CTMUCON1 (CTMU Control 1)	
Enhanced Parallel Master/Slave Port	55	CTMUCON2 (CTMU Control 2)	
I ² C	46	CTMUICON (CTMU Current Control)	
ICN		CVRCON (Comparator Voltage	500
Input Capture		Reference Control)	379
Interrupt Controller		resolution dominal	570
NVM			

CW1 (Flash Configuration Word 1)	390	MTHDY (RTCC Month and Day Value)	. 321
CW2 (Flash Configuration Word 2)	392	NVMCON (Flash Memory Control)	81
CW3 (Flash Configuration Word 3)	394	OCxCON1 (Output Compare x Control 1)	. 222
CW4 (Flash Configuration Word 4)	396	OCxCON2 (Output Compare x Control 2)	. 224
DEVID (Device ID)	398	OSCCON (Oscillator Control)	. 149
DEVREV (Device Revision)	398	OSCTUN (FRC Oscillator Tune)	. 152
DMACHn (DMA Channel n Control)		PADCFG1 (Pad Configuration Control)	
DMACON (DMA Engine Control)		PMCON1 (EPMP Control 1)	
DMAINTn (DMA Channel n Interrupt)		PMCON2 (EPMP Control 2)	
DSCON (Deep Sleep Control)		PMCON3 (EPMP Control 3)	
DSWAKE (Deep Sleep Wake-up Source)		PMCON4 (EPMP Control 4)	
HLVDCON (High/Low-Voltage Detect Control)		PMCSxBS (Chip Select x Base Address)	
I2CxCONH (I2Cx Control High)		PMCSxCF (EPMP Chip Select x Configuration)	
I2CxCONL (I2Cx Control Low)		PMCSxMD (EPMP Chip Select x Mode)	
I2CxMSK (I2Cx Slave Mode Address Mask)		PMSTAT (EPMP Status, Slave Mode)	
I2CxSTAT (I2Cx Status)		RCFGCAL (RTCC Calibration	. 510
,		· · · · · · · · · · · · · · · · · · ·	247
ICxCON1 (Input Capture x Control 1)		and Configuration)	
ICxCON2 (Input Capture x Control 2)		RCON (Reset Control)	
IEC0 (Interrupt Enable Control 0)		RCON2 (Reset and System Control 2)	, 170
IEC1 (Interrupt Enable Control 1)		REFOCONH (Reference Oscillator	
IEC2 (Interrupt Enable Control 2)		Control High)	. 158
IEC3 (Interrupt Enable Control 3)		REFOCONL (Reference Oscillator	
IEC4 (Interrupt Enable Control 4)		Control Low)	
IEC5 (Interrupt Enable Control 5)		REFOTRIML (Reference Oscillator Trim)	
IEC6 (Interrupt Enable Control 6)	121	RPINR0 (PPS Input 0)	. 183
IEC7 (Interrupt Enable Control 7)	121	RPINR1 (PPS Input 1)	. 183
IFS0 (Interrupt Flag Status 0)	100	RPINR11 (PPS Input 11)	. 186
IFS1 (Interrupt Flag Status 1)	102	RPINR17 (PPS Input 17)	. 186
IFS2 (Interrupt Flag Status 2)	104	RPINR18 (PPS Input 18)	. 187
IFS3 (Interrupt Flag Status 3)	106	RPINR19 (PPS Input 19)	
IFS4 (Interrupt Flag Status 4)		RPINR2 (PPS Input 2)	
IFS5 (Interrupt Flag Status 5)		RPINR20 (PPS Input 20)	
IFS6 (Interrupt Flag Status 6)		RPINR21 (PPS Input 21)	
IFS7 (Interrupt Flag Status 7)		RPINR22 (PPS Input 22)	
INTCON1 (Interrupt Control 1)		RPINR23 (PPS Input 23)	
INTCON2 (Interrupt Control 2)		RPINR27 (PPS Input 27)	
INTTREG (Interrupt Controller Test)		RPINR28 (PPS Input 28)	
IPC0 (Interrupt Priority Control 0)		RPINR29 (PPS Input 29)	
IPC1 (Interrupt Priority Control 1)		RPINR30 (PPS Input 30)	
IPC10 (Interrupt Priority Control 10)		RPINR31 (PPS Input 31)	
` ' '		RPINR7 (PPS Input 7)	
IPC11 (Interrupt Priority Control 11)			
IPC12 (Interrupt Priority Control 12)		RPINR8 (PPS Input 8)	
IPC13 (Interrupt Priority Control 13)		RPINR9 (PPS Input 9)	
IPC14 (Interrupt Priority Control 14)		RPOR0 (PPS Output 0)	
IPC15 (Interrupt Priority Control 15)		RPOR1 (PPS Output 1)	
IPC16 (Interrupt Priority Control 16)		RPOR10 (PPS Output 10)	
IPC18 (Interrupt Priority Control 18)		RPOR11 (PPS Output 11)	
IPC19 (Interrupt Priority Control 19)		RPOR12 (PPS Output 12)	
IPC2 (Interrupt Priority Control 2)		RPOR2 (PPS Output 2)	
IPC20 (Interrupt Priority Control 20)	140	RPOR3 (PPS Output 3)	
IPC21 (Interrupt Priority Control 21)	141	RPOR4 (PPS Output 4)	. 195
IPC22 (Interrupt Priority Control 22)	142	RPOR5 (PPS Output 5)	. 195
IPC26 (Interrupt Priority Control 26)	143	RPOR6 (PPS Output 6)	. 196
IPC29 (Interrupt Priority Control 29)	143	RPOR7 (PPS Output 7)	. 196
IPC3 (Interrupt Priority Control 3)	125	RPOR8 (PPS Output 8)	
IPC4 (Interrupt Priority Control 4)		RPOR9 (PPS Output 9)	
IPC5 (Interrupt Priority Control 5)		RTCCSWT (RTCC Power Control and	
IPC6 (Interrupt Priority Control 6)		Sample Window Timer)	. 325
IPC7 (Interrupt Priority Control 7)		RTCPWC (RTCC Power Control)	
IPC8 (Interrupt Priority Control 8)		SPIxCON1H (SPIx Control 1 High)	
IPC9 (Interrupt Priority Control 9)		SPIxCON1L (SPIx Control 1 Low)	
MDCAR (DSM Carrier Control)		SPIxCON1L (SPIx Control 2 Low)	
MDCON (DSM Control)		SPIxIMSKH (SPIx Interrupt Mask High)	
MDSRC (DSM Source Control)		SPIXIMSKI (GPIX Interrupt Mask Low)	
MINSEC (RTCC Minutes and Seconds Value)	322	SPIxSTATH (SPIx Status High)	. ∠ 30

	SPIxSTATL (SPIx Status Low)	236
	SR (ALU STATUS)	
	T1CON (Timer1 Control)	
	TxCON (Timer2 and Timer4 Control)	
	TyCON (Timer3 and Timer5 Control)	
	U1ADDR (USB Address)	
	U1CNFG1 (USB Configuration 1)	
	U1CNFG2 (USB Configuration 2)	
	U1CON (USB Control, Device Mode)	
	U1CON (USB Control, Host Mode)	
	U1EIE (USB Error Interrupt Enable)	
	U1EIR (USB Error Interrupt Status)	
	U1EPn (USB Endpoint n Control)	
	U1IE (USB Interrupt Enable, All Modes)	
	U1IR (USB Interrupt Status, Device Mode)	
	U1IR (USB Interrupt Status, Host Mode)	
	U10TGCON (USB OTG Control)	
		20 1
	U1OTGIE (USB OTG Interrupt Enable,	204
	Host Mode)	291
	U1OTGIR (USB OTG Interrupt Status,	000
	Host Mode)	
	U1OTGSTAT (USB OTG Status, Host Mode)	
	U1PWRC (USB Power Control)	
	U1SOF (USB OTG SOF Count, Host Mode)	
	U1STAT (USB Status)	
	U1TOK (USB Token, Host Mode)	
	UxADMD (UARTx Address Match Detect)	262
	UxMODE (UARTx Mode)	
	UxSCCON (UARTx Smart Card Control)	
	UxSCINT (UARTx Smart Card Interrupt)	
	LIVETA (LIADTy Status and Control)	
	UxSTA (UARTx Status and Control)	
	UxTXREG (UARTx Transmit)	262
	UXTXREG (UARTX Transmit)WKDYHR (RTCC Weekday and Hours Value)	262 322
	UxTXREG (UARTx Transmit)	262 322
Res	UxTXREG (UARTx Transmit)	262 322
Res	UXTXREG (UARTx Transmit)	262 322 321
Res	UxTXREG (UARTx Transmit)	262 322 321
Res	UXTXREG (UARTx Transmit)	262 322 321 90
Res	UXTXREG (UARTx Transmit)	262 322 90 85 89
Res	UXTXREG (UARTx Transmit)	262 321 90 85 89 89
Res	UXTXREG (UARTx Transmit)	262 321 90 85 89 89
Res	UXTXREG (UARTx Transmit)	262 322 90 85 89 89
Res	UXTXREG (UARTx Transmit)	262 321 90 85 89 89 85
Res	UXTXREG (UARTx Transmit)	262 321 90 85 89 89 85 90
Res	UXTXREG (UARTx Transmit)	262 321 90 85 89 89 85 90 85
Res	UXTXREG (UARTx Transmit)	262 321 90 85 89 89 85 90 89 85
Res	UXTXREG (UARTx Transmit)	262 321 90 85 89 85 90 85 85 85
Res	UXTXREG (UARTx Transmit)	262 321 90 85 89 89 85 90 85 85 85
Res	UXTXREG (UARTx Transmit)	262 321 90 85 89 85 85 85 85 85 85
Res	UXTXREG (UARTx Transmit)	262 321 90 85 89 85 85 85 85 85
Res	UXTXREG (UARTx Transmit)	262 321 90 85 89 85 85 85 85 85 85
	UXTXREG (UARTx Transmit)	262 321 90 85 89 85 85 85 85 85 85
	UXTXREG (UARTx Transmit)	262 321 90 85 89 85 85 85 85 85 85
Revi	UXTXREG (UARTx Transmit) WKDYHR (RTCC Weekday and Hours Value) YEAR (RTCC Year Value) ets and Fail-Safe Clock Monitor BOR (Brown-out Reset) Brown-out Reset (BOR) Clock Source Selection CM (Configuration Mismatch Reset) Delay Times Device Times IOPUWR (Illegal Opcode Reset) MCLR (Master Clear Pin Reset) POR (Power-on Reset) RCON Flags, Operation SFR States SWR (RESET Instruction) TRAPR (Trap Conflict Reset) UWR (Uninitialized W Register Reset) WDT (Watchdog Timer Reset)	262 321 90 85 89 85 85 85 85 85 85 85 85
Revi	UXTXREG (UARTx Transmit) WKDYHR (RTCC Weekday and Hours Value) YEAR (RTCC Year Value) ets and Fail-Safe Clock Monitor BOR (Brown-out Reset) Brown-out Reset (BOR) Clock Source Selection CM (Configuration Mismatch Reset) Delay Times Device Times IOPUWR (Illegal Opcode Reset) MCLR (Master Clear Pin Reset) POR (Power-on Reset) RCON Flags, Operation SFR States SWR (RESET Instruction) TRAPR (Trap Conflict Reset) UWR (Uninitialized W Register Reset) WDT (Watchdog Timer Reset) ision History	262 321 90 85 89 85 85 85 85 85 85 85 85 85
Revi	UXTXREG (UARTx Transmit) WKDYHR (RTCC Weekday and Hours Value) YEAR (RTCC Year Value) ets and Fail-Safe Clock Monitor. BOR (Brown-out Reset) Brown-out Reset (BOR) Clock Source Selection CM (Configuration Mismatch Reset) Delay Times Device Times IOPUWR (Illegal Opcode Reset) MCLR (Master Clear Pin Reset) POR (Power-on Reset) RCON Flags, Operation SFR States. SWR (RESET Instruction) TRAPR (Trap Conflict Reset) UWR (Uninitialized W Register Reset) WDT (Watchdog Timer Reset) ision History CC Alarm Configuration Alarm Mask Settings (figure)	262 321 90 85 89 85 85 85 85 85 85 85 85 85 85 85
Revi	UXTXREG (UARTx Transmit)	
Revi	UXTXREG (UARTx Transmit) WKDYHR (RTCC Weekday and Hours Value) YEAR (RTCC Year Value) ets and Fail-Safe Clock Monitor. BOR (Brown-out Reset) Brown-out Reset (BOR) Clock Source Selection CM (Configuration Mismatch Reset) Delay Times Device Times IOPUWR (Illegal Opcode Reset) MCLR (Master Clear Pin Reset) POR (Power-on Reset) RCON Flags, Operation SFR States SWR (RESET Instruction) TRAPR (Trap Conflict Reset) UWR (Uninitialized W Register Reset) WDT (Watchdog Timer Reset) ision History CC Alarm Configuration Alarm Mask Settings (figure) ALRMVAL Register Mappings Calibration	
Revi	UXTXREG (UARTx Transmit) WKDYHR (RTCC Weekday and Hours Value) YEAR (RTCC Year Value) ets and Fail-Safe Clock Monitor. BOR (Brown-out Reset) Brown-out Reset (BOR) Clock Source Selection CM (Configuration Mismatch Reset) Delay Times Device Times IOPUWR (Illegal Opcode Reset) MCLR (Master Clear Pin Reset) POR (Power-on Reset) RCON Flags, Operation SFR States SWR (RESET Instruction) TRAPR (Trap Conflict Reset) UWR (Uninitialized W Register Reset) WDT (Watchdog Timer Reset) ision History CC Alarm Configuration Alarm Mask Settings (figure) ALRMVAL Register Mappings Calibration Clock Source Selection	
Revi	UXTXREG (UARTx Transmit) WKDYHR (RTCC Weekday and Hours Value) YEAR (RTCC Year Value) ets and Fail-Safe Clock Monitor. BOR (Brown-out Reset) Brown-out Reset (BOR) Clock Source Selection CM (Configuration Mismatch Reset) Delay Times Device Times IOPUWR (Illegal Opcode Reset) MCLR (Master Clear Pin Reset) POR (Power-on Reset) RCON Flags, Operation SFR States SWR (RESET Instruction) TRAPR (Trap Conflict Reset) UWR (Uninitialized W Register Reset) WDT (Watchdog Timer Reset) ision History CC Alarm Configuration Alarm Mask Settings (figure) ALRMVAL Register Mappings Calibration Clock Source Selection Control Registers	
Revi	UXTXREG (UARTx Transmit) WKDYHR (RTCC Weekday and Hours Value) YEAR (RTCC Year Value) ets and Fail-Safe Clock Monitor BOR (Brown-out Reset) Brown-out Reset (BOR) Clock Source Selection CM (Configuration Mismatch Reset) Delay Times Delay Times IOPUWR (Illegal Opcode Reset) MCLR (Master Clear Pin Reset) POR (Power-on Reset) RCON Flags, Operation SFR States SWR (RESET Instruction) TRAPR (Trap Conflict Reset) UWR (Uninitialized W Register Reset) WDT (Watchdog Timer Reset) ision History CC Alarm Configuration Alarm Mask Settings (figure) ALRMVAL Register Mappings Calibration Clock Source Selection Control Registers Module Registers	262 321 90 85 89 85 90 85 85 85 85 85 85 85 85 85 85 85 85 85 85 85 85 85 85
Revi	UXTXREG (UARTx Transmit) WKDYHR (RTCC Weekday and Hours Value) YEAR (RTCC Year Value) ets and Fail-Safe Clock Monitor. BOR (Brown-out Reset) Brown-out Reset (BOR) Clock Source Selection CM (Configuration Mismatch Reset) Delay Times Device Times IOPUWR (Illegal Opcode Reset) MCLR (Master Clear Pin Reset) POR (Power-on Reset) RCON Flags, Operation SFR States SWR (RESET Instruction) TRAPR (Trap Conflict Reset) UWR (Uninitialized W Register Reset) WDT (Watchdog Timer Reset) ision History CC Alarm Configuration Alarm Mask Settings (figure) ALRMVAL Register Mappings Calibration Clock Source Selection Control Registers	

RTCVAL Register Mappings	321
Source Clock	315
VBAT Operation	
Write Lock	316
S	
	474
Selective Peripheral Module Control	
Serial Peripheral Interface (SPI)	221
Serial Peripheral Interface. See SPI. SFR Space	20
Software Simulator	30
MPLAB X SIM	405
Software Stack	
Special Features	
SPI	
Audio Mode	230
Control Registers	
Enhanced Master Mode	
Enhanced Slave Mode	
Standard Master Mode	
Standard Slave Mode	
_	
Т	
Timer1	201
Timer2/3 and Timer4/5	205
Timing Diagrams	
CLKO and I/O Characteristics	430
External Clock	
I ² C Bus Data (Master Mode)	
I ² C Bus Data (Slave Mode)	444
I ² C Bus Start/Stop Bits (Master Mode)	
I ² C Bus Start/Stop Bits (Slave Mode)	
Input Capture x (ICx)	434
OCx/PWM Characteristics	435
Output Compare x (OCx)	434
SPIx Master Mode (CKE = 0)	436
SPIx Master Mode (CKE = 1)	437
SPIx Slave Mode (CKE = 0)	438
SPIx Slave Mode (CKE = 1)	439
Timer1, 2, 3, 4, 5 External Clock	432
Triple Comparator	371
Triple Comparator Module	371
U	
UART (PPO)	
Baud Rate Generator (BRG)	
Control Registers	
Infrared Support	
Operation of UxCTS and UxRTS Pins	256
Receiving	050
8-Bit or 9-Bit Data Mode	
Smart Card ISO 7816 Support	257
Transmitting	
8-Bit Data Mode	
9-Bit Data Mode	
Break and Sync Sequence	
Universal Asynchronous Receiver Transmitter. See UA	NRT.
Universal Serial Bus. See USB OTG.	
USB On-The-Go (OTG)	10

USB OTG	265
Buffer Descriptors	
Assignment in Different Buffering Modes	271
Buffer Descriptors and BDT	270
Control Registers	279
Controller-Centric Data Direction for Host	
or Target	265
Device Mode Operation	
DMA Interface	271
Hardware Configuration	267
Calculating Transceiver Power	
Requirements	269
Device Mode	267
Host and OTG Modes	268
Power Modes	
Bus Power Only	267
Dual Power with Self-Power	
Dominance	267
Self-Power Only	267
VBUS Voltage Generation	269
Host Mode Operation	276
Interrupts	274
and USB Transactions	275
Operation	
Host Negotiation Protocol (HNP)	
Session Request Protocol (SRP)	278

W	
Watchdog Timer (WDT)	400
Control Register	400
Windowed Operation	400
WWW Address	473
WWW, On-Line Support	7

NOTES:

THE MICROCHIP WEB SITE

Microchip provides online support via our WWW site at www.microchip.com. This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- Product Support Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- General Technical Support Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- Business of Microchip Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at www.microchip.com. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- · Distributor or Representative
- · Local Sales Office
- Field Application Engineer (FAE)
- · Technical Support

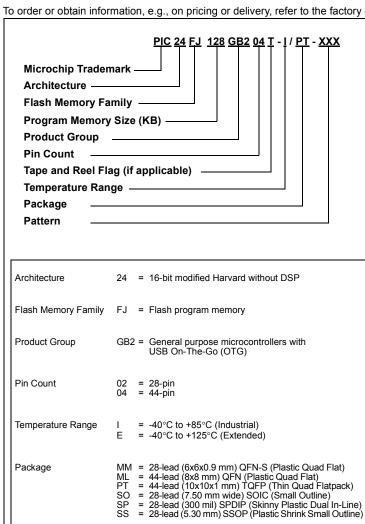
Customers should contact their distributor, representative or Field Application Engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the web site at: http://microchip.com/support

NOTES:

PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.



Three-digit QTP, SQTP, Code or Special Requirements

(blank otherwise) ES = Engineering Sample

Examples:

- PIC24FJ128GB202-I/MM: PIC24F device with USB On-The-Go, 128-Kbyte program memory, 8-Kbyte data memory, 28-pin, Industrial temp., QFN-S package.
- PIC24FJ128GB204-I/PT: PIC24F device with USB On-The-Go, 128-Kbyte program memory, 8-Kbyte data memory, 44-pin, Industrial temp., TQFP package.

Pattern

NOTES:

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.

Trademarks

The Microchip name and logo, the Microchip logo, dsPIC, FlashFlex, flexPWR, JukeBlox, KEELoQ, KEELoQ logo, Kleer, LANCheck, 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.

The Embedded Control Solutions Company and mTouch are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Analog-for-the-Digital Age, BodyCom, chipKIT, chipKIT logo, CodeGuard, dsPICDEM, dsPICDEM.net, ECAN, In-Circuit Serial Programming, ICSP, Inter-Chip Connectivity, KleerNet, KleerNet logo, MiWi, MPASM, MPF, MPLAB Certified logo, MPLIB, MPLINK, MultiTRAK, NetDetach, Omniscient Code Generation, PICDEM, PICDEM.net, PICkit, PICtail, RightTouch logo, REAL ICE, SQI, Serial Quad I/O, 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.

 $\ensuremath{\mathsf{SQTP}}$ 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.

© 2013-2015, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

ISBN: 978-1-63277-204-6

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

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.



Worldwide Sales and Service

AMERICAS

Corporate Office 2355 West Chandler Blvd. Chandler, AZ 85224-6199

Tel: 480-792-7200 Fax: 480-792-7277 Technical Support:

http://www.microchip.com/

support
Web Address:

www.microchip.com

Atlanta

Duluth, GA Tel: 678-957-9614 Fax: 678-957-1455

Austin, TX Tel: 512-257-3370

Boston

Westborough, MA Tel: 774-760-0087 Fax: 774-760-0088

Chicago Itasca, IL

Tel: 630-285-0071 Fax: 630-285-0075

Cleveland

Independence, OH Tel: 216-447-0464 Fax: 216-447-0643

Dallas

Addison, TX Tel: 972-818-7423 Fax: 972-818-2924

Detroit Novi. MI

Tel: 248-848-4000

Houston, TX Tel: 281-894-5983

Indianapolis

Noblesville, IN Tel: 317-773-8323 Fax: 317-773-5453

Los Angeles

Mission Viejo, CA Tel: 949-462-9523 Fax: 949-462-9608

New York, NY Tel: 631-435-6000

San Jose, CA Tel: 408-735-9110

Canada - Toronto Tel: 905-673-0699 Fax: 905-673-6509

ASIA/PACIFIC

Asia Pacific Office

Suites 3707-14, 37th Floor Tower 6, The Gateway Harbour City, Kowloon Hong Kong

Tel: 852-2943-5100 Fax: 852-2401-3431

Australia - Sydney Tel: 61-2-9868-6733 Fax: 61-2-9868-6755

China - Beijing

Tel: 86-10-8569-7000 Fax: 86-10-8528-2104

China - Chengdu Tel: 86-28-8665-5511

Fax: 86-28-8665-7889 China - Chongqing

Tel: 86-23-8980-9588 Fax: 86-23-8980-9500

China - Dongguan

Tel: 86-769-8702-9880

China - Hangzhou Tel: 86-571-8792-8115 Fax: 86-571-8792-8116

China - Hong Kong SAR Tel: 852-2943-5100

Fax: 852-2401-3431

China - Nanjing Tel: 86-25-8473-2460

Fax: 86-25-8473-2470
China - Qingdao

Tel: 86-532-8502-7355 Fax: 86-532-8502-7205

China - Shanghai Tel: 86-21-5407-5533

Tel: 86-21-5407-5533 Fax: 86-21-5407-5066

China - Shenyang Tel: 86-24-2334-2829 Fax: 86-24-2334-2393

China - Shenzhen

Tel: 86-755-8864-2200 Fax: 86-755-8203-1760

China - Wuhan

Tel: 86-27-5980-5300 Fax: 86-27-5980-5118

China - Xian

Tel: 86-29-8833-7252 Fax: 86-29-8833-7256

ASIA/PACIFIC

China - Xiamen

Tel: 86-592-2388138 Fax: 86-592-2388130

China - Zhuhai

Tel: 86-756-3210040 Fax: 86-756-3210049

India - Bangalore

Tel: 91-80-3090-4444 Fax: 91-80-3090-4123

India - New Delhi

Tel: 91-11-4160-8631 Fax: 91-11-4160-8632

India - Pune

Tel: 91-20-3019-1500

Japan - Osaka

Tel: 81-6-6152-7160 Fax: 81-6-6152-9310

Japan - Tokyo

Tel: 81-3-6880- 3770 Fax: 81-3-6880-3771

Korea - Daegu

Tel: 82-53-744-4301 Fax: 82-53-744-4302

Korea - Seoul

Tel: 82-2-554-7200 Fax: 82-2-558-5932 or 82-2-558-5934

Malaysia - Kuala Lumpur

Tel: 60-3-6201-9857 Fax: 60-3-6201-9859

Malaysia - Penang

Tel: 60-4-227-8870 Fax: 60-4-227-4068

Philippines - Manila

Tel: 63-2-634-9065

Fax: 63-2-634-9069

Singapore

Tel: 65-6334-8870 Fax: 65-6334-8850

Taiwan - Hsin Chu

Tel: 886-3-5778-366 Fax: 886-3-5770-955

Taiwan - Kaohsiung Tel: 886-7-213-7828

Taiwan - Taipei

Tel: 886-2-2508-8600 Fax: 886-2-2508-0102

Thailand - Bangkok

Tel: 66-2-694-1351 Fax: 66-2-694-1350

EUROPE

Austria - Wels

Tel: 43-7242-2244-39 Fax: 43-7242-2244-393

Denmark - Copenhagen

Tel: 45-4450-2828 Fax: 45-4485-2829

France - Paris

Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79

Germany - Dusseldorf

Tel: 49-2129-3766400

Germany - Munich Tel: 49-89-627-144-0

Fax: 49-89-627-144-44

Germany - Pforzheim Tel: 49-7231-424750

Italy - Milan

Tel: 39-0331-742611 Fax: 39-0331-466781

Italy - Venice

Tel: 39-049-7625286

Netherlands - Drunen

Tel: 31-416-690399 Fax: 31-416-690340

Poland - Warsaw

Tel: 48-22-3325737

Spain - Madrid

Tel: 34-91-708-08-90 Fax: 34-91-708-08-91

Sweden - Stockholm

Tel: 46-8-5090-4654

UK - Wokingham Tel: 44-118-921-5800 Fax: 44-118-921-5820

01/27/15

Mouser Electronics

Authorized Distributor

Click to View Pricing, Inventory, Delivery & Lifecycle Information:

Microchip:

 PIC24FJ128GB204-I/PT
 PIC24FJ128GB202-I/MM
 PIC24FJ128GB204-I/ML
 PIC24FJ128GB202-I/SS

 PIC24FJ128GB202-I/SO
 PIC24FJ128GB202-I/SP
 PIC24FJ164GB204-E/PT
 PIC24FJ128GB202-E/SO

 PIC24FJ128GB204-E/PT
 PIC24FJ64GB202-E/SO
 PIC24FJ128GB202-E/MM
 PIC24FJ64GB202-E/SP

 PIC24FJ64GB204-E/ML
 PIC24FJ64GB202-E/SS
 PIC24FJ128GB202-E/SP
 PIC24FJ128GB204-E/ML

 PIC24FJ64GB202-E/SS
 PIC24FJ64GB204-I/PT
 PIC24FJ128GB204T-I/ML
 PIC24FJ64GB202-I/SP
 PIC24FJ64GB202-I/SO

 PIC24FJ64GB202-I/SS
 PIC24FJ64GB202T-I/SO
 PIC24FJ64GB202-I/MM
 PIC24FJ128GB204T-I/PT
 PIC24FJ128GB202T-I/SS
 PIC24FJ64GB204T-I/SS
 PIC24FJ64GB204T-I/ML

 PIC24FJ64GB204T-I/ML
 PIC24FJ64GB204T-I/ML
 PIC24FJ64GB204T-I/ML
 PIC24FJ64GB204T-I/ML
 PIC24FJ64GB204T-I/ML



OOO «ЛайфЭлектроникс" "LifeElectronics" LLC

ИНН 7805602321 КПП 780501001 P/C 40702810122510004610 ФАКБ "АБСОЛЮТ БАНК" (ЗАО) в г.Санкт-Петербурге К/С 3010181090000000703 БИК 044030703

Компания «Life Electronics» занимается поставками электронных компонентов импортного и отечественного производства от производителей и со складов крупных дистрибьюторов Европы, Америки и Азии.

С конца 2013 года компания активно расширяет линейку поставок компонентов по направлению коаксиальный кабель, кварцевые генераторы и конденсаторы (керамические, пленочные, электролитические), за счёт заключения дистрибьюторских договоров

Мы предлагаем:

- Конкурентоспособные цены и скидки постоянным клиентам.
- Специальные условия для постоянных клиентов.
- Подбор аналогов.
- Поставку компонентов в любых объемах, удовлетворяющих вашим потребностям.
- Приемлемые сроки поставки, возможна ускоренная поставка.
- Доставку товара в любую точку России и стран СНГ.
- Комплексную поставку.
- Работу по проектам и поставку образцов.
- Формирование склада под заказчика.
- Сертификаты соответствия на поставляемую продукцию (по желанию клиента).
- Тестирование поставляемой продукции.
- Поставку компонентов, требующих военную и космическую приемку.
- Входной контроль качества.
- Наличие сертификата ISO.

В составе нашей компании организован Конструкторский отдел, призванный помогать разработчикам, и инженерам.

Конструкторский отдел помогает осуществить:

- Регистрацию проекта у производителя компонентов.
- Техническую поддержку проекта.
- Защиту от снятия компонента с производства.
- Оценку стоимости проекта по компонентам.
- Изготовление тестовой платы монтаж и пусконаладочные работы.



Тел: +7 (812) 336 43 04 (многоканальный) Email: org@lifeelectronics.ru