# 16-bit Digital Signal Controllers (up to 256 KB Flash and 30 KB SRAM) with Motor Control and Advanced Analog

# **Operating Conditions**

- 3.0V to 3.6V, -40°C to +150°C, DC to 20 MIPS
- 3.0V to 3.6V, -40°C to +125°C, DC to 40 MIPS

#### Core: 16-bit dsPIC33F CPU

- · Code-efficient (C and Assembly) architecture
- · Two 40-bit wide accumulators
- · Single-cycle (MAC/MPY) with dual data fetch
- Single-cycle mixed-sign MUL plus hardware divide

#### **Clock Management**

- ±2% internal oscillator
- · Programmable PLLs and oscillator clock sources
- Fail-Safe Clock Monitor (FSCM)
- Independent Watchdog Timer (WDT)
- · Fast wake-up and start-up

#### **Power Management**

- Low-power management modes (Sleep, Idle, Doze)
- · Integrated Power-on Reset and Brown-out Reset
- 1.35 mA/MHz dynamic current (typical)
- 55 μA IPD current (typical)

#### **Motor Control PWM**

- Up to four PWM generators with eight outputs
- · Dead Time for rising and falling edges
- 12.5 ns PWM resolution
- PWM support for Motor Control: BLDC, PMSM, ACIM, and SRM
- · Programmable Fault inputs
- · Flexible trigger for ADC conversions and configurations

#### **Advanced Analog Features**

- · Two ADC modules:
  - Configurable as 10-bit, 1.1 Msps with four S&H or 12-bit, 500 ksps with one S&H
  - 18 analog inputs on 64-pin devices and up to 32 analog inputs on 100-pin devices
- Flexible and independent ADC trigger sources

# **Timers/Output Compare/Input Capture**

- Up to nine 16-bit timers/counters. Can pair up to make four 32-bit timers.
- Eight Output Compare modules configurable as timers/counters
- · Eight Input Capture modules

## **Communication Interfaces**

- Two UART modules (10 Mbps)
  - With support for LIN 2.0 protocols and IrDA<sup>®</sup>
- Two 4-wire SPI modules (15 Mbps)
- Up to two I<sup>2</sup>C<sup>™</sup> modules (up to 1 Mbaud) with SMBus support
- Up to two Enhanced CAN (ECAN) modules (1 Mbaud) with 2.0B support
- · Quadrature Encoder Interface (QEI) module
- Data Converter Interface (DCI) module with I<sup>2</sup>S codec support

#### Input/Output

- Sink/Source up to 10 mA (pin specific) for standard VOH/VOL, up to 16 mA (pin specific) for nonstandard VOH1
- 5V-tolerant pins
- · Selectable open drain, pull-ups, and pull-downs
- · Up to 5 mA overvoltage clamp current
- · External interrupts on all I/O pins

#### **Qualification and Class B Support**

- AEC-Q100 REVG (Grade 1 -40°C to +125°C)
- AEC-Q100 REVG (Grade 0 -40°C to +150°C)
- Class B Safety Library, IEC 60730

## **Debugger Development Support**

- · In-circuit and in-application programming
- Two program and two complex data breakpoints
- IEEE 1149.2-compatible (JTAG) boundary scan
- · Trace and run-time watch

#### **Packages**

Туре	QFN	TQFP	TQFP	TQFP
Pin Count	64	64	80	100
Contact Lead/Pitch	0.50	0.50	0.50	0.40
I/O Pins	53	53	69	85
Dimensions	9x9x0.9	10x10x1	12x12x1	14x14x1

Note: All dimensions are in millimeters (mm) unless specified.

## dsPIC33F PRODUCT FAMILIES

The dsPIC33FJXXXMCX06A/X08A/X10A family of devices supports a variety of motor control applications, such as brushless DC motors, single and 3-phase induction motors and switched reluctance motors. The dsPIC33F Motor Control products are also well-suited for Uninterrupted Power Supply (UPS), inverters, Switched mode power supplies, power factor correction and also for controlling the power management module in servers, telecommunication equipment and other industrial equipment.

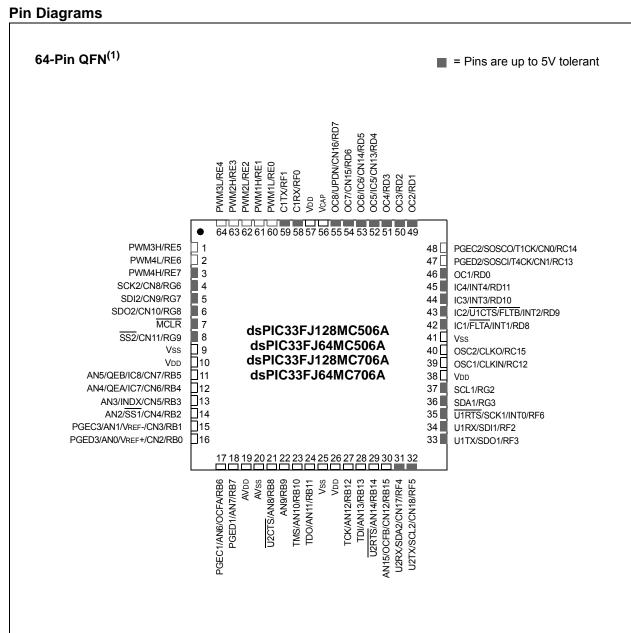
The device names, pin counts, memory sizes and peripheral availability of each device are listed below. The following pages show their pinout diagrams.

#### dsPIC33FJXXXMCX06A/X08A/X10A Controller Families

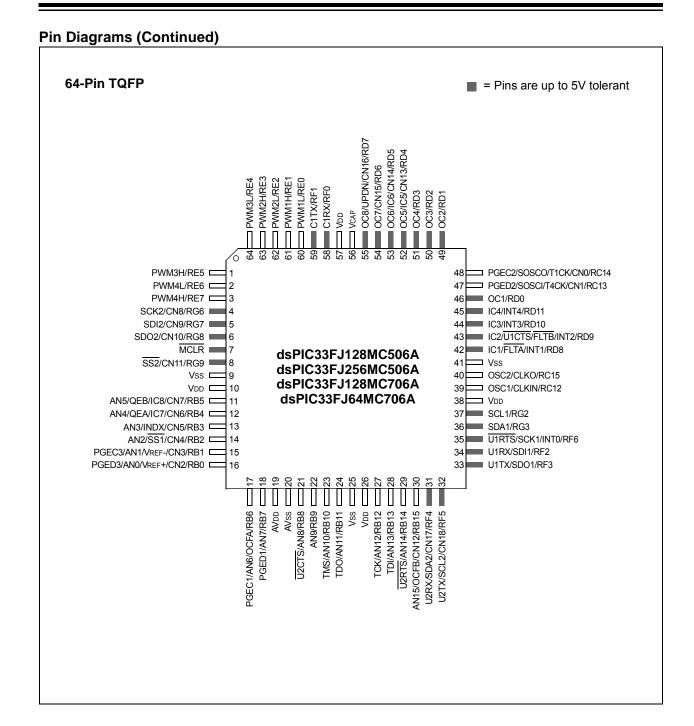
Device	Pins	Program Flash Memory (Kbyte)	RAM (Kbyte) <sup>(1)</sup>	Timer 16-bit	Input Capture	Output Compare Std. PWM	Motor Control PWM	Quadrature Encoder Interface	Codec Interface	ADC	UART	SPI	I²С™	Enhanced CAN	I/O Pins (Max) <sup>(2)</sup>	Packages
dsPIC33FJ64MC506A	64	64	8	9	8	8	8 ch	1	0	1 ADC, 16 ch	2	2	2	1	53	PT, MR
dsPIC33FJ64MC508A	80	64	8	9	8	8	8 ch	1	0	1 ADC, 18 ch	2	2	2	1	69	PT
dsPIC33FJ64MC510A	100	64	8	9	8	8	8 ch	1	0	1 ADC, 24 ch	2	2	2	1	85	PF, PT
dsPIC33FJ64MC706A	64	64	16	9	8	8	8 ch	1	0	2 ADC, 16 ch	2	2	2	1	53	PT, MR
dsPIC33FJ64MC710A	100	64	16	9	8	8	8 ch	1	0	2 ADC, 24 ch	2	2	2	2	85	PF, PT
dsPIC33FJ128MC506A	64	128	8	9	8	8	8 ch	1	0	1 ADC, 16 ch	2	2	2	1	53	PT, MR
dsPIC33FJ128MC510A	100	128	8	9	8	8	8 ch	1	0	1 ADC, 24 ch	2	2	2	1	85	PF, PT
dsPIC33FJ128MC706A	64	128	16	9	8	8	8 ch	1	0	2 ADC, 16 ch	2	2	2	1	53	PT, MR
dsPIC33FJ128MC708A	80	128	16	9	8	8	8 ch	1	0	2 ADC, 18 ch	2	2	2	2	69	PT
dsPIC33FJ128MC710A	100	128	16	9	8	8	8 ch	1	0	2 ADC, 24 ch	2	2	2	2	85	PF, PT
dsPIC33FJ256MC510A	100	256	16	9	8	8	8 ch	1	0	1 ADC, 24 ch	2	2	2	1	85	PF, PT
dsPIC33FJ256MC710A	100	256	30	9	8	8	8 ch	1	0	2 ADC, 24 ch	2	2	2	2	85	PF, PT

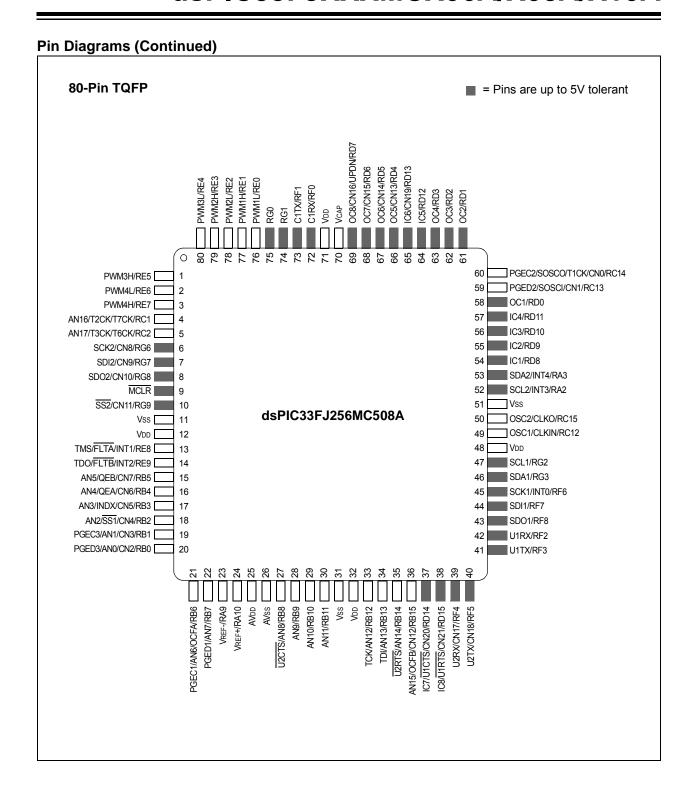
Note 1: RAM size is inclusive of 2 Kbytes DMA RAM.

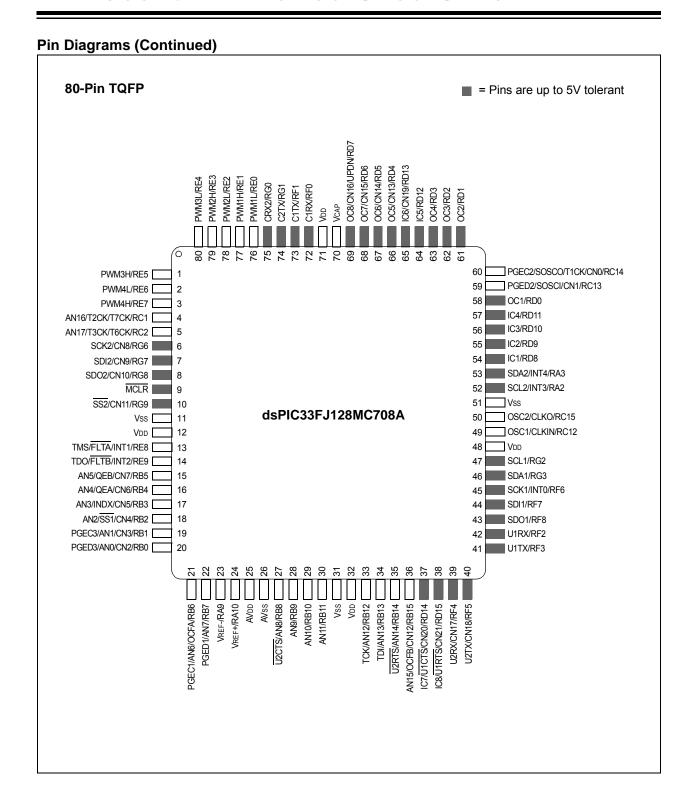
<sup>2:</sup> Maximum I/O pin count includes pins shared by the peripheral functions.

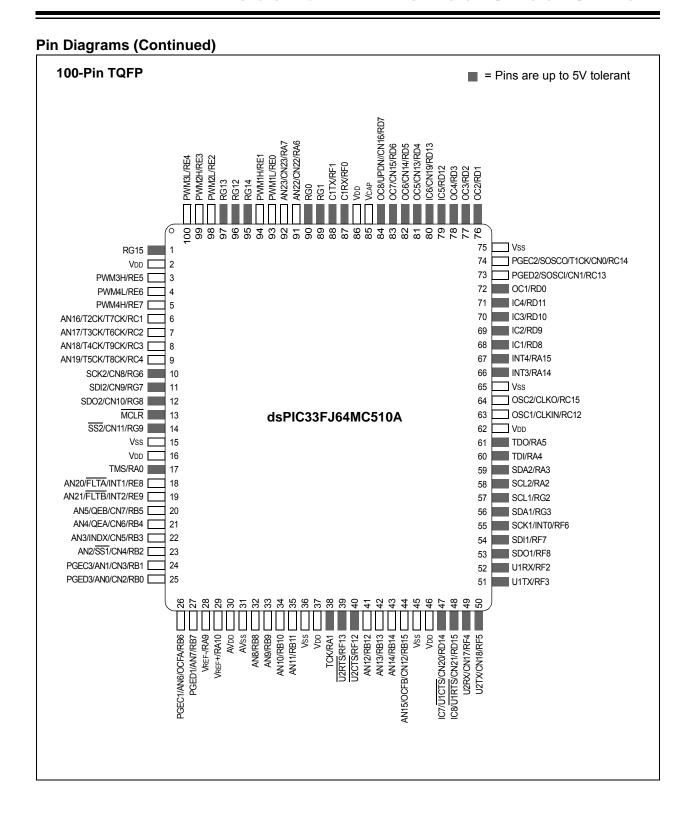


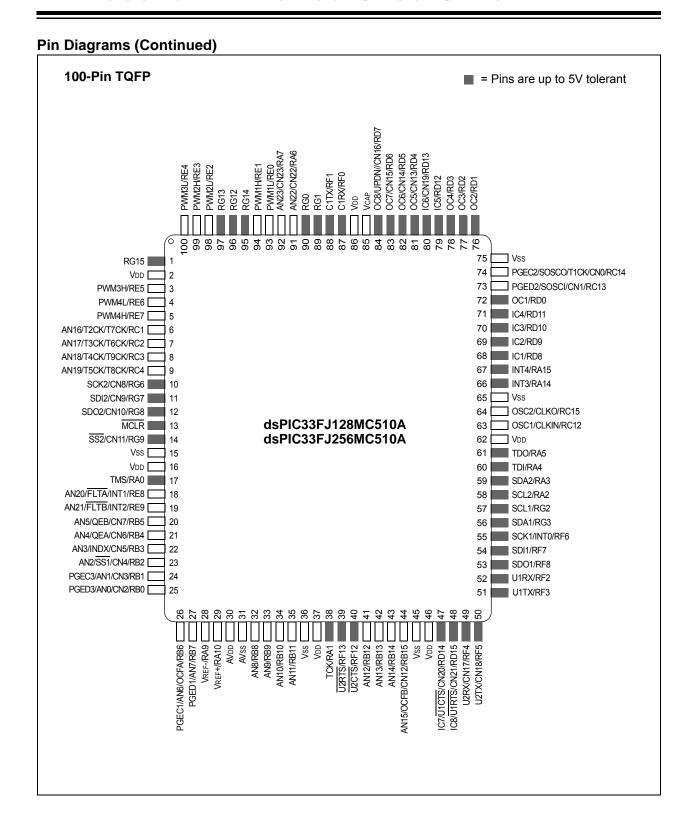
**Note 1:** The metal plane at the bottom of the device is not connected to any pins and should be connected to Vss externally.

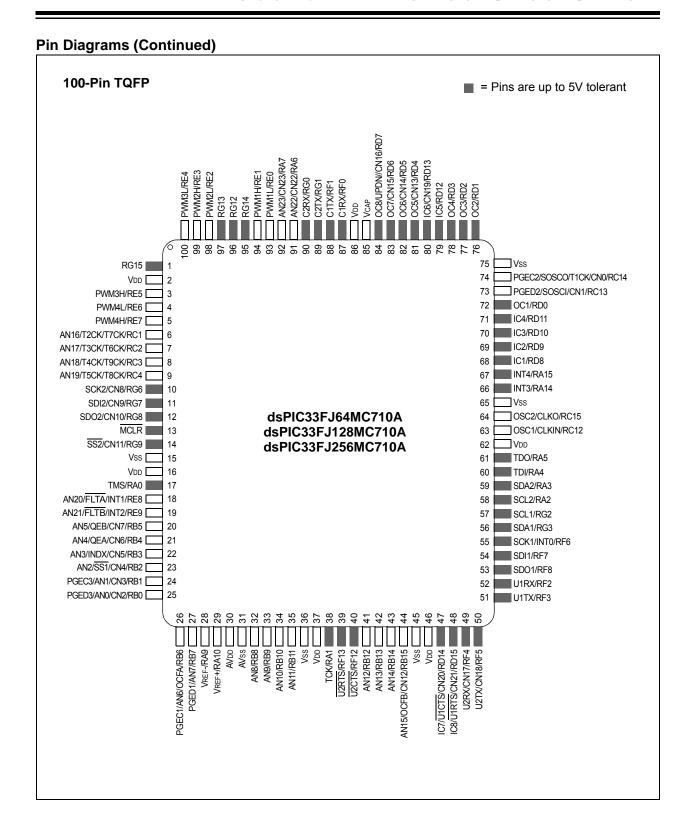












# **Table of Contents**

dsPl	C33F Product Families	2
1.0	Device Overview	
2.0	Guidelines for Getting Started with 16-bit Digital Signal Controllers	19
3.0	CPU	23
4.0	Memory Organization	35
5.0	Flash Program Memory	73
6.0	Reset	79
7.0	Interrupt Controller	85
8.0	Direct Memory Access (DMA)	133
9.0	Oscillator Configuration	143
10.0	Power-Saving Features	153
11.0	I/O Ports	
12.0	Timer1	165
13.0	Timer2/3, Timer4/5, Timer6/7 and Timer8/9	167
14.0	Input Capture	173
15.0	Output Compare	175
16.0	Motor Control PWM Module	179
17.0	Quadrature Encoder Interface (QEI) Module	193
18.0	Serial Peripheral Interface (SPI)	197
19.0	Inter-Integrated Circuit (I <sup>2</sup> C™)	203
20.0	Universal Asynchronous Receiver Transmitter (UART)	211
21.0	Enhanced CAN Module	217
22.0	10-bit/12-bit Analog-to-Digital Converter (ADC)	245
23.0	Special Features	259
24.0	Instruction Set Summary	267
25.0	Development Support	275
	Electrical Characteristics	
27.0	High Temperature Electrical Characteristics	329
28.0	DC and AC Device Characteristics Graphs	339
	Packaging Information	
Appe	endix A: Migrating from dsPIC33FJXXXMCX06/X08/X10 Devices to dsPIC33FJXXXMCX06A/X08A/X10A Devices	357
	endix B: Revision History	
Index	C	363
The I	Microchip Web Site	369
Cust	omer Change Notification Service	369
	omer Support	
	ler Response	
	uct Identification System	371

#### 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** or fax the **Reader Response Form** in the back of this data sheet to (480) 792-4150. 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., DS30000A is version A of document DS30000).

#### **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.

#### **Referenced Sources**

This device data sheet is based on the following individual chapters of the "dsPIC33F/PIC24H Family Reference Manual". These documents should be considered as the general reference for the operation of a particular module or device feature.

Note:

To access the documents listed below, browse to the documentation section of the dsPIC33FJ256MC710A product page on the Microchip web site (www.microchip.com) or select a family reference manual section from the following list.

In addition to parameters, features, and other documentation, the resulting page provides links to the related family reference manual sections.

- Section 1. "Introduction" (DS70197)
- Section 2. "CPU" (DS70204)
- Section 3. "Data Memory" (DS70202)
- Section 4. "Program Memory" (DS70203)
- Section 5. "Flash Programming" (DS70191)
- Section 6. "Interrupts" (DS70184)
- Section 7. "Oscillator" (DS70186)
- Section 8. "Reset" (DS70192)
- Section 9. "Watchdog Timer and Power-Saving Modes" (DS70196)
- Section 10. "I/O Ports" (DS70193)
- Section 11. "Timers" (DS70205)
- Section 12. "Input Capture" (DS70198)
- Section 13. "Output Compare" (DS70209)
- Section 14. "Motor Control PWM" (DS70187)
- Section 15. "Quadrature Encoder Interface (QEI)" (DS70208)
- Section 16. "Analog-to-Digital Converter (ADC)" (DS70183)
- Section 17. "UART" (DS70188)
- Section 18. "Serial Peripheral Interface (SPI)" (DS70206)
- Section 19. "Inter-Integrated Circuit™ (I2C™)" (DS70195)
- Section 20. "Data Converter Interface (DCI)" (DS70288)
- Section 21. "Enhanced Controller Area Network (ECAN™)" (DS70185)
- Section 22. "Direct Memory Access (DMA)" (DS70182)
- Section 23. "CodeGuard™ Security" (DS70199)
- Section 24. "Programming and Diagnostics" (DS70207)
- Section 25. "Device Configuration" (DS70194)

## 1.0 DEVICE OVERVIEW

Note 1: This data sheet summarizes the features of the dsPIC33FJXXXMCX06A/X08A/X10A family of devices. However, it is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to the "dsPIC33F/PIC24H Family Reference Manual". Please see the Microchip web site (www.microchip.com) for the latest dsPIC33F/PIC24H Family Reference Manual sections.

2: Some registers and associated bits described in this section may not be available on all devices. Refer to Section 4.0 "Memory Organization" in this data sheet for device-specific register and bit information.

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

- dsPIC33FJ64MC506A
- dsPIC33FJ64MC508A
- dsPIC33FJ64MC510A
- dsPIC33FJ64MC706A
- dsPIC33FJ64MC710A
- dsPIC33FJ128MC506A
- dsPIC33FJ128MC510A
- dsPIC33FJ128MC706A
- dsPIC33FJ128MC708A
- dsPIC33FJ128MC710A
- dsPIC33FJ256MC510A
- dsPIC33FJ256MC710A

The dsPIC33FJXXXMCX06A/X08A/X10A includes devices with a wide range of pin counts (64, 80 and 100), different program memory sizes (64 Kbytes, 128 Kbytes and 256 Kbytes) and different RAM sizes (8 Kbytes, 16 Kbytes and 30 Kbytes).

These features make this family suitable for a wide variety of high-performance, digital signal control applications. The devices are pin compatible with the PIC24H family of devices, and also share a very high degree of compatibility with the dsPIC30F family devices. This allows easy migration between device families as may be necessitated by the specific functionality, computational resource and system cost requirements of the application.

The dsPIC33FJXXXMCX06A/X08A/X10A family of devices employs a powerful 16-bit architecture that seamlessly integrates the control features of a Microcontroller (MCU) with the computational capabilities of a Digital Signal Processor (DSP). The resulting functionality is ideal for applications that rely on high-speed, repetitive computations, as well as control.

The DSP engine, dual 40-bit accumulators, hardware support for division operations, barrel shifter, 17 x 17 multiplier, a large array of 16-bit working registers and a wide variety of data addressing modes, together, the dsPIC33FJXXXMCX06A/X08A/X10A Central Processing Unit (CPU) with extensive mathematical processing capability. Flexible and deterministic interrupt handling, coupled with a powerful array of peripherals, renders the dsPIC33FJXXXMCX06A/X08A/X10A devices suitable for control applications. Further, Direct Memory Access (DMA) enables overhead-free transfer of data between several peripherals and a dedicated DMA RAM. Reliable, field programmable Flash program memory ensures scalability of applications that use dsPIC33FJXXXMCX06A/X08A/X10A devices.

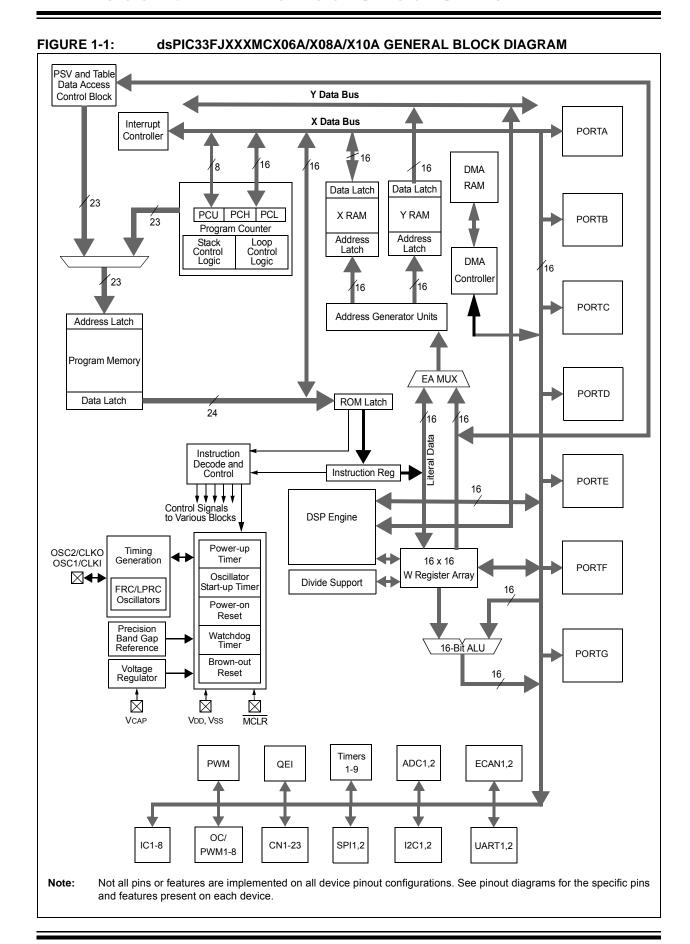


TABLE 1-1: PINOUT I/O DESCRIPTIONS

Pin Name	Pin	Buffer	Description Description
AN0-AN31	Туре	Type Analog	Analog input channels.
	1	_	<del>-</del> -
AVDD	Р	Р	Positive supply for analog modules. This pin must be connected at all times.
AVss	Р	Р	Ground reference for analog modules.
CLKI CLKO	0	ST/CMOS —	External clock source input. Always associated with OSC1 pin function. Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode. Optionally functions as CLKO in RC and EC modes. Always associated with OSC2 pin function.
CN0-CN23	I	ST	Input change notification inputs. Can be software programmed for internal weak pull-ups on all inputs.
C1RX	ı	ST	ECAN1 bus receive pin.
C1TX	0	_	ECAN1 bus transmit pin.
C2RX	i	ST	ECAN2 bus receive pin.
C2TX	0	_	ECAN2 bus transmit pin.
PGED1	I/O	ST	Data I/O pin for Programming/Debugging Communication Channel 1.
PGEC1	"0	ST	Clock input pin for Programming/Debugging Communication Channel 1.
PGED2	1/0	ST	Data I/O pin for Programming/Debugging Communication Channel 2.
PGEC2	1	ST	Clock input pin for Programming/Debugging Communication Channel 2.
PGED3	I/O	ST	Data I/O pin for Programming/Debugging Communication Channel 3.
PGEC3	Ī	ST	Clock input pin for Programming/Debugging Communication Channel 3.
IC1-IC8	I	ST	Capture Inputs 1 through 8.
INDX	1	ST	Quadrature Encoder Index Pulse input.
QEA	l i	ST	Quadrature Encoder Phase A input in QEI mode. Auxiliary timer external clock/
QZ, t	·	0.	gate input in Timer mode.
QEB	I	ST	Quadrature Encoder Phase A input in QEI mode. Auxiliary timer external clock/gate input in Timer mode.
UPDN	0	CMOS	Position up/down counter direction state.
INT0	I	ST	External Interrupt 0.
INT1	I	ST	External Interrupt 1.
INT2	I	ST	External Interrupt 2.
INT3	I	ST	External Interrupt 3.
INT4	I	ST	External Interrupt 4.
FLTA	I	ST	PWM Fault A input.
FLTB	I	ST	PWM Fault B input.
PWM1L	0	_	PWM1 low output.
PWM1H	0	-	PWM1 high output.
PWM2L	0	_	PWM2 low output.
PWM2H	0	_	PWM2 high output.
PWM3L	0	_	PWM3 low output.
PWM3H	0	_	PWM3 high output.
PWM4L	0	-	PWM4 low output.
PWM4H	0		PWM4 high output.
MCLR	I/P	ST	Master Clear (Reset) input. This pin is an active-low Reset to the device.
OCFA	!	ST	Compare Fault A input (for Compare Channels 1, 2, 3 and 4).
OCFB		ST	Compare Fault B input (for Compare Channels 5, 6, 7 and 8).
OC1-OC8	0	_	Compare outputs 1 through 8.
OSC1	I	ST/CMOS	Oscillator crystal input. ST buffer when configured in RC mode; CMOS otherwise.
OSC2	I/O	_	Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode. Optionally functions as CLKO in RC and EC modes.

**Legend:** CMOS = CMOS compatible input or output

ST = Schmitt Trigger input with CMOS levels (

Analog = Analog input

O = Output

P = Power I = Input

TABLE 1-1: PINOUT I/O DESCRIPTIONS (CONTINUED)

TABLE 1-1: PINOUT I/O DESCRIPTIONS (CONTINUED)					
Pin Name	Pin Type	Buffer Type	Description		
RA0-RA7 RA9-RA10 RA12-RA15	I/O I/O I/O	ST ST ST	PORTA is a bidirectional I/O port.		
RB0-RB15	I/O	ST	PORTB is a bidirectional I/O port.		
RC1-RC4 RC12-RC15	I/O I/O	ST ST	PORTC is a bidirectional I/O port.		
RD0-RD15	I/O	ST	PORTD is a bidirectional I/O port.		
RE0-RE9	I/O	ST	PORTE is a bidirectional I/O port.		
RF0-RF8 RF12-RF13	I/O	ST	PORTF is a bidirectional I/O port.		
RG0-RG3 RG6-RG9 RG12-RG15	I/O I/O I/O	ST ST ST	PORTG is a bidirectional I/O port.		
SCK1 SDI1 SDO1 SS1 SCK2 SDI2 SDO2 SS2	1/O 1 0 1/O 1/O 1/O 1	ST ST — ST ST ST — ST	Synchronous serial clock input/output for SPI1. SPI1 data in. SPI1 data out. SPI1 slave synchronization or frame pulse I/O. Synchronous serial clock input/output for SPI2. SPI2 data in. SPI2 data out. SPI2 slave synchronization or frame pulse I/O.		
SCL1 SDA1 SCL2 SDA2	I/O I/O I/O I/O	ST ST ST ST	Synchronous serial clock input/output for I2C1. Synchronous serial data input/output for I2C1. Synchronous serial clock input/output for I2C2. Synchronous serial data input/output for I2C2.		
SOSCI SOSCO	I 0	ST/CMOS	32.768 kHz low-power oscillator crystal input; CMOS otherwise. 32.768 kHz low-power oscillator crystal output.		
TMS TCK TDI TDO	 	ST ST ST	JTAG Test mode select pin. JTAG test clock input pin. JTAG test data input pin. JTAG test data output pin.		
T1CK T2CK T3CK T4CK T5CK T6CK T7CK T8CK T9CK		ST ST ST ST ST ST ST ST	Timer1 external clock input. Timer2 external clock input. Timer3 external clock input. Timer4 external clock input. Timer5 external clock input. Timer6 external clock input. Timer7 external clock input. Timer8 external clock input. Timer8 external clock input. Timer9 external clock input.		
U1CTS U1RTS U1RX U1TX U2CTS U2RTS U2RX U2TX	   0   0   0   0	ST — ST — ST — ST —	UART1 clear to send. UART1 ready to send. UART1 receive. UART1 transmit. UART2 clear to send. UART2 ready to send. UART2 receive. UART2 transmit.		
VDD	Р	_	Positive supply for peripheral logic and I/O pins.		
VCAP	Р	_	CPU logic filter capacitor connection.		

**Legend:** CMOS = CMOS compatible input or output ST = Schmitt Trigger input with CMOS levels

Analog = Analog input O = Output P = Power I = Input

# TABLE 1-1: PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name	Pin Type	Buffer Type	Description
Vss	Р	_	Ground reference for logic and I/O pins.
VREF+	I	Analog	Analog voltage reference (high) input.
VREF-	I	Analog	Analog voltage reference (low) input.

**Legend:** CMOS = CMOS compatible input or output

Analog = Analog input

P = Power

ST = Schmitt Trigger input with CMOS levels

O = Output

I = Input

NO	T	Ε	S	

# 2.0 GUIDELINES FOR GETTING STARTED WITH 16-BIT DIGITAL SIGNAL CONTROLLERS

- Note 1: This data sheet summarizes the features of the dsPIC33FJXXXMCX06A/X08A/X10A family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to the "dsPIC33F/PIC24H Family Reference Manual", which is available from the Microchip web site (www.microchip.com).
  - 2: Some registers and associated bits described in this section may not be available on all devices. Refer to Section 4.0 "Memory Organization" in this data sheet for device-specific register and bit information.

# 2.1 Basic Connection Requirements

Getting started with the dsPIC33FJXXXMCX06A/X08A/X10A family of 16-bit Digital Signal Controllers (DSC) requires attention to a minimal set of device pin connections before proceeding with development. The following is a list of pin names, which must always be connected:

- All VDD and Vss pins (see Section 2.2 "Decoupling Capacitors")
- All AVDD and AVSS pins (regardless if ADC module is not used)

(see Section 2.2 "Decoupling Capacitors")

- VCAP
   (see Section 2.3 "CPU Logic Filter Capacitor Connection (VCAP)")
- MCLR pin (see Section 2.4 "Master Clear (MCLR) Pin")
- PGECx/PGEDx pins used for In-Circuit Serial Programming™ (ICSP™) and debugging purposes (see Section 2.5 "ICSP Pins")
- OSC1 and OSC2 pins when 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 ADC module is implemented

**Note:** The AVDD and AVSS pins must be connected independent of the ADC voltage reference source.

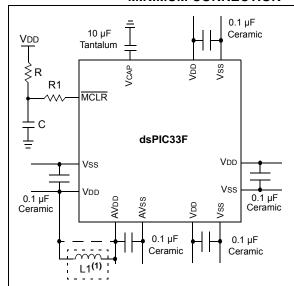
#### 2.2 Decoupling Capacitors

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

# FIGURE 2-1: RECOMMENDED MINIMUM CONNECTION



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

Where:

$$f=rac{FCNV}{2}$$
 (i.e., ADC conversion rate/2) 
$$f=rac{1}{(2\pi\sqrt{LC})}$$
  $L=\left(rac{1}{(2\pi f\sqrt{C})}
ight)^2$ 

#### 2.2.1 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 DSCs 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  $\mu F$  to 47  $\mu F$ .

# 2.3 CPU Logic Filter Capacitor Connection (VCAP)

A low-ESR (< 5 Ohms) capacitor is required on the VCAP pin, which is used to stabilize the voltage regulator output voltage. The VCAP pin must not be connected to VDD and must have a capacitor between 4.7  $\mu$ F and 10  $\mu$ F, 16V connected to ground. The type can be ceramic or tantalum. Refer to Section 26.0 "Electrical Characteristics" for additional information.

The placement of this capacitor should be close to the VCAP. It is recommended that the trace length not exceed one-quarter inch (6 mm). Refer to Section 23.2 "On-Chip Voltage Regulator" for details.

# 2.4 Master Clear (MCLR) Pin

The  $\overline{\text{MCLR}}$  pin provides for two specific device functions:

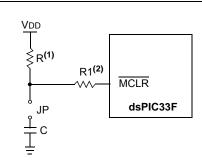
- · Device Reset
- · Device Programming and Debugging

During device programming and debugging, the resistance and capacitance that can be added to the pin must be considered. Device programmers and debuggers drive the  $\overline{\text{MCLR}}$  pin. Consequently, specific voltage levels (VIH and VIL) and fast signal transitions must not be adversely affected. Therefore, specific values of R and C will need to be adjusted based on the application and PCB requirements.

For example, as shown in Figure 2-2, it is recommended that the capacitor, C, be isolated from the  $\overline{\text{MCLR}}$  pin during programming and debugging operations.

Place the components shown in Figure 2-2 within one-quarter inch (6 mm) from the  $\overline{\text{MCLR}}$  pin.

# FIGURE 2-2: EXAMPLE OF MCLR PIN CONNECTIONS



- Note 1:  $R \le 10 \text{ k}\Omega$  is recommended. A suggested starting value is  $10 \text{ k}\Omega$ . Ensure that the MCLR pin VIH and VIL specifications are met.
  - 2:  $R1 \le 470\Omega$  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.

#### 2.5 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 100 Ohms.

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 "dsPIC33F/PIC24H Flash Programming Specification" (DS70152) for information on capacitive loading limits, and pin input voltage high (VIH) and input low (VIL) requirements.

Ensure that the "Communication Channel Select" (i.e., PGECx/PGEDx pins) programmed into the device matches the physical connections for the ICSP to the MPLAB® ICD 3 or REAL ICE  $^{\text{TM}}$  in-circuit emulator.

For more information on the ICD 3 and REAL ICE in-circuit emulator connection requirements, refer to the following documents that are available on the Microchip web site.

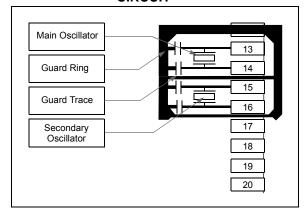
- "Using MPLAB® ICD 3" (poster) (DS51765)
- "MPLAB® ICD 3 Design Advisory" (DS51764)
- "MPLAB® REAL ICE™ In-Circuit Emulator User's Guide" (DS51616)
- "Using MPLAB® REAL ICE™ In-Circuit Emulator" (poster) (DS51749)

#### 2.6 External Oscillator Pins

Many DSCs 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. Also, place the oscillator circuit close to the respective oscillator pins, not exceeding one-half inch (12 mm) distance between them. 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 them 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. A suggested layout is shown in Figure 2-3.

FIGURE 2-3: SUGGESTED PLACEMENT
OF THE OSCILLATOR
CIRCUIT



# 2.7 Oscillator Value Conditions on Device Start-up

If the PLL of the target device is enabled and configured for the device start-up oscillator, the maximum oscillator source frequency must be limited to  $\leq 8$  MHz for start-up with PLL enabled to comply with device PLL start-up conditions. This means that if the external oscillator frequency is outside this range, the application must start-up in the FRC mode first. The default PLL settings after a POR with an oscillator frequency outside this range will violate the device operating speed.

Once the device powers up, the application firmware can initialize the PLL SFRs, CLKDIV and PLLDBF to a suitable value, and then perform a clock switch to the oscillator + PLL clock source. Note that clock switching must be enabled in the device Configuration Word.

# 2.8 Configuration of Analog and Digital Pins During ICSP Operations

If the MPLAB ICD 3 or REAL ICE in-circuit emulator is selected as a debugger, it automatically initializes all of the A/D input pins (ANx) as "digital" pins by setting all bits in the AD1PCFGL register.

The bits in this register that correspond to the A/D pins that are initialized by the MPLAB ICD 3 or REAL ICE in-circuit emulator, must not be cleared 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 clear the corresponding bits in the AD1PCFGL register during initialization of the ADC module.

When the MPLAB ICD 3 or REAL ICE in-circuit emulator is used as a programmer, the user application firmware must correctly configure the AD1PCFGL register. 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.

#### 2.9 Unused I/Os

Unused I/O pins should be configured as outputs and driven to a logic low state.

Alternatively, connect a 1k to 10k resistor between Vss and the unused pins.

## 3.0 CPU

# Note 1: This data sheet summarizes the features of the dsPIC33FJXXXMCX06A/X08A/X10A family of devices. However, it is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to Section 2. "CPU" (DS70204) in the "dsPIC33F/PIC24H Family Reference Manual", which is available from the Microchip web site (www.microchip.com).

2: Some registers and associated bits described in this section may not be available on all devices. Refer to Section 4.0 "Memory Organization" in this data sheet for device-specific register and bit information.

The dsPIC33FJXXXMCX06A/X08A/X10A CPU module has a 16-bit (data) modified Harvard architecture with an enhanced instruction set, including significant support for DSP. The CPU has a 24-bit instruction word with a variable length opcode field. The Program Counter (PC) is 23 bits wide and addresses up to 4M x 24 bits of user program memory space. The actual amount of program memory implemented varies by device. 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 DO and REPEAT instructions, both of which are interruptible at any point.

The dsPIC33FJXXXMCX06A/X08A/X10A devices have sixteen, 16-bit working registers in the programmer's model. Each of the working registers can serve as a data, address or address offset register. The 16th working register (W15) operates as a software Stack Pointer (SP) for interrupts and calls.

The dsPIC33FJXXXMCX06A/X08A/X10A instruction set has two classes of instructions: MCU and DSP. These two instruction classes are seamlessly integrated into a single CPU. The instruction set includes many addressing modes and is designed for optimum 'C' compiler efficiency. For most instructions, the dsPIC33FJXXXMCX06A/X08A/X10A devices are 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 A + B = C operations to be executed in a single cycle.

A block diagram of the CPU is shown in Figure 3-1 and the programmer's model for the dsPIC33FJXXXMCX06A/X08A/X10A is shown in Figure 3-2.

# 3.1 Data Addressing Overview

The data space can be addressed as 32K words or 64 Kbytes, and is split into two blocks referred to as X and Y data memory. Each memory block has its own independent Address Generation Unit (AGU). The MCU class of instructions operates solely through the X memory AGU, which accesses the entire memory map as one linear data space. Certain DSP instructions operate through the X and Y AGUs to support dual operand reads, which splits the data address space into two parts. The X and Y data space boundary is device-specific.

Overhead-free circular buffers (Modulo Addressing mode) are supported in both X and Y address spaces. The Modulo Addressing removes the software boundary checking overhead for DSP algorithms. Furthermore, the X AGU circular addressing can be used with any of the MCU class of instructions. The X AGU also supports Bit-Reversed Addressing to greatly simplify input or output data reordering for radix-2 FFT algorithms.

The upper 32 Kbytes of the data space memory map can optionally be mapped into program space at any 16K program word boundary defined by the 8-bit Program Space Visibility Page register (PSVPAG). The program to data space mapping feature lets any instruction access program space as if it were data space.

The data space also includes 2 Kbytes of DMA RAM, which is primarily used for DMA data transfers but may be used as general purpose RAM.

#### 3.2 DSP Engine Overview

The DSP engine features a high-speed, 17-bit by 17-bit multiplier, a 40-bit ALU, two 40-bit saturating accumulators and a 40-bit bidirectional barrel shifter. The barrel shifter is capable of shifting a 40-bit value up to 16 bits right or left in a single cycle. The DSP instructions operate seamlessly with all other instructions and have been designed for optimal real-time performance. The MAC instruction and other associated instructions can concurrently fetch two data operands from memory while multiplying two W registers, and accumulating and optionally saturating the result in the same cycle. This instruction functionality requires that the RAM memory data space be split for these instructions and linear for all others. Data space partitioning is achieved in a transparent and flexible manner through dedicating certain working registers to each address space.

#### 3.3 Special MCU Features

The dsPIC33FJXXXMCX06A/X08A/X10A devices feature a 17-bit by 17-bit, single-cycle multiplier that is shared by both the MCU ALU and DSP engine. The multiplier can perform signed, unsigned and mixed sign multiplication. Using a 17-bit by 17-bit multiplier for 16-bit by 16-bit multiplication not only allows you to perform mixed sign multiplication, it also achieves accurate results for special operations, such as (-1.0) x (-1.0).

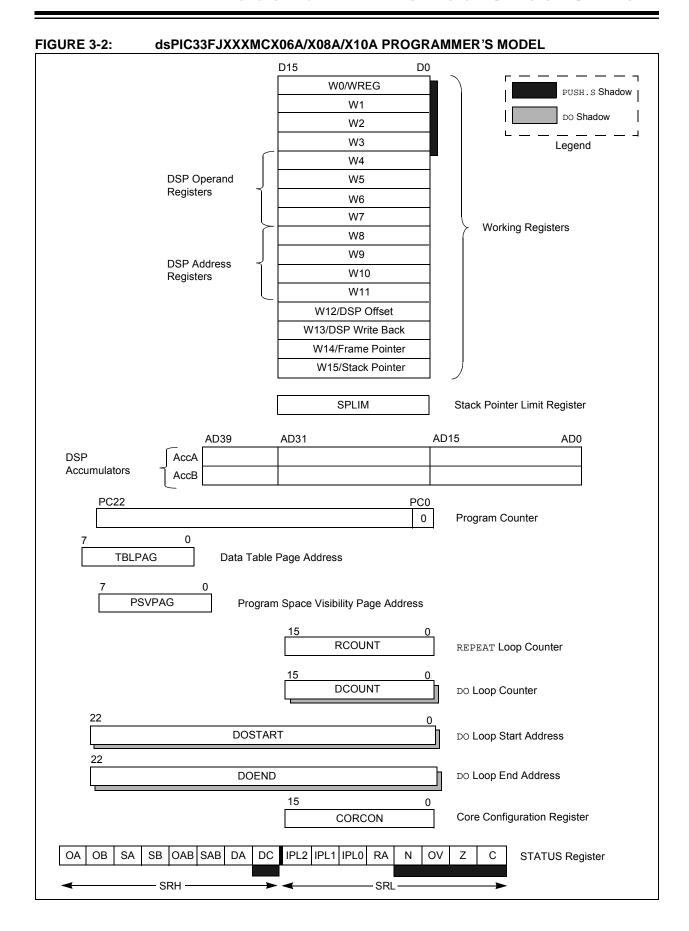
The dsPIC33FJXXXMCX06A/X08A/X10A devices support 16/16 and 32/16 divide operations, both fractional and integer. All divide instructions are iterative operations. They must be executed within a REPEAT loop, resulting in a total execution time of 19 instruction cycles. The divide operation can be interrupted during any of those 19 cycles without a loss of data.

A 40-bit barrel shifter is used to perform up to a 16-bit left or right shift in a single cycle. The barrel shifter can be used by both MCU and DSP instructions.

FIGURE 3-1: dsPIC33FJXXXMCX06A/X08A/X10A CPU CORE BLOCK DIAGRAM PSV and Table **Data Access** Control Block Y Data Bus X Data Bus Interrupt Controlle **1**16 . 16 16 Data Latch Data Latch DMA PCU PCH PCL 16 X RAM Y RAM RAM Program Counter Address Stack Control Loop Control Address Latch Latch Logic Logic 23 16 DMA Controller Address Generator Units Address Latch **Program Memory** EA MUX Data Latch ROM Latch 24 16 Instruction Decode and Instruction Reg Control 16  $\downarrow$   $\downarrow$   $\downarrow$   $\downarrow$ Control Signals to Various Blocks **DSP** Engine 16 x 16 W Register Array Divide Support 16 16-Bit ALU 16

DS70594D-page 24

To Peripheral Modules



# 3.4 CPU Control Registers

#### REGISTER 3-1: SR: CPU STATUS REGISTER

R-0	R-0	R/C-0	R/C-0	R-0	R/C-0	R -0	R/W-0
OA	ОВ	SA <sup>(1)</sup>	SB <sup>(1)</sup>	OAB	SAB <sup>(4)</sup>	DA	DC
bit 15							bit 8

R/W-0 <sup>(3)</sup>	R/W-0 <sup>(3)</sup>	R/W-0 <sup>(3)</sup>	R-0	R/W-0	R/W-0	R/W-0	R/W-0
	IPL<2:0> <sup>(2)</sup>		RA	N	OV	Z	С
bit 7							bit 0

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

bit 15 OA: Accumulator A Overflow Status bit

1 = Accumulator A overflowed

0 = Accumulator A has not overflowed

bit 14 **OB:** Accumulator B Overflow Status bit

1 = Accumulator B overflowed

0 = Accumulator B has not overflowed

bit 13 SA: Accumulator A Saturation 'Sticky' Status bit (1)

1 = Accumulator A is saturated or has been saturated at some time

0 = Accumulator A is not saturated

bit 12 SB: Accumulator B Saturation 'Sticky' Status bit<sup>(1)</sup>

1 = Accumulator B is saturated or has been saturated at some time

0 = Accumulator B is not saturated

bit 11 OAB: OA || OB Combined Accumulator Overflow Status bit

1 = Accumulators A or B have overflowed

0 = Neither Accumulators A or B have overflowed

bit 10 SAB: SA || SB Combined Accumulator 'Sticky' Status bit (4)

1 = Accumulators A or B are saturated or have been saturated at some time in the past

0 = Neither Accumulator A or B are saturated

bit 9 DA: DO Loop Active bit

1 = DO loop in progress

0 = DO loop not in progress

bit 8 **DC:** MCU 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 low-order bit (for byte-sized data) or 8th low-order bit (for word-sized data) of the result occurred

Note 1: This bit may be read or cleared (not set).

2: The IPL<2:0> bits are concatenated with the IPL<3> bit (CORCON<3>) to form the CPU interrupt priority level. The value in parentheses indicates the IPL if IPL<3> = 1. User interrupts are disabled when IPL<3> = 1.

3: The IPL<2:0> Status bits are read only when NSTDIS = 1 (INTCON1<15>).

4: This bit may be read or cleared (not set). Clearing this bit will clear SA and SB.

#### REGISTER 3-1: SR: CPU STATUS REGISTER (CONTINUED)

IPL<2:0>: CPU Interrupt Priority Level Status bits<sup>(2)</sup> bit 7-5 111 = CPU interrupt priority level is 7 (15), user interrupts 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 in progress 0 = REPEAT loop not in progress bit 3 N: MCU ALU Negative bit

1 = Result was negative

0 = Result was non-negative (zero or positive)

bit 2 **OV:** MCU ALU Overflow bit

This bit is used for signed arithmetic (2's complement). It indicates an overflow of the magnitude that causes the sign bit to change state.

1 = Overflow occurred for signed arithmetic (in this arithmetic operation)

0 = No overflow occurred

bit 1 **Z:** MCU 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: MCU ALU Carry/Borrow bit

1 = A carry-out from the Most Significant bit of the result occurred

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

Note 1: This bit may be read or cleared (not set).

- 2: The IPL<2:0> bits are concatenated with the IPL<3> bit (CORCON<3>) to form the CPU interrupt priority level. The value in parentheses indicates the IPL if IPL<3> = 1. User interrupts are disabled when IPL<3> = 1.
- 3: The IPL<2:0> Status bits are read only when NSTDIS = 1 (INTCON1<15>).
- 4: This bit may be read or cleared (not set). Clearing this bit will clear SA and SB.

#### REGISTER 3-2: CORCON: CORE CONTROL REGISTER

U-0	U-0	U-0	R/W-0	R/W-0	R-0	R-0	R-0
_	_	_	US	EDT <sup>(1)</sup>		DL<2:0>	
bit 15	_	_					bit 8

R/W-0	R/W-0	R/W-1	R/W-0	R/C-0	R/W-0	R/W-0	R/W-0
SATA	SATB	SATDW	ACCSAT	IPL3 <sup>(2)</sup>	PSV	RND	IF
bit 7							bit 0

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

bit 15-13 Unimplemented: Read as '0'

bit 12 US: DSP Multiply Unsigned/Signed Control bit

1 = DSP engine multiplies are unsigned0 = DSP engine multiplies are signed

bit 11 EDT: Early DO Loop Termination Control bit<sup>(1)</sup>

1 = Terminate executing DO loop at end of current loop iteration

0 = No effect

bit 10-8 DL<2:0>: DO Loop Nesting Level Status bits

111 = 7 DO loops active

:

001 = 1 DO loop active 000 = 0 DO loops active

bit 7 SATA: AccA Saturation Enable bit

1 = Accumulator A saturation enabled0 = Accumulator A saturation disabled

bit 6 SATB: AccB Saturation Enable bit

1 = Accumulator B saturation enabled0 = Accumulator B saturation disabled

bit 5 SATDW: Data Space Write from DSP Engine Saturation Enable bit

1 = Data space write saturation enabled0 = Data space write saturation disabled

bit 4 ACCSAT: Accumulator Saturation Mode Select bit

1 = 9.31 saturation (super saturation) 0 = 1.31 saturation (normal saturation)

bit 3 IPL3: CPU Interrupt Priority Level Status bit 3<sup>(2)</sup>

1 = CPU interrupt priority level is greater than 7 0 = CPU interrupt priority level is 7 or less

bit 2 **PSV:** Program Space Visibility in Data Space Enable bit

1 = Program space visible in data space0 = Program space not visible in data space

bit 1 RND: Rounding Mode Select bit

1 = Biased (conventional) rounding enabled0 = Unbiased (convergent) rounding enabled

bit 0 IF: Integer or Fractional Multiplier Mode Select bit

1 = Integer mode enabled for DSP multiply ops0 = Fractional mode enabled for DSP multiply ops

Note 1: This bit will always read as '0'.

2: The IPL3 bit is concatenated with the IPL<2:0> bits (SR<7:5>) to form the CPU interrupt priority level.

#### 3.5 Arithmetic Logic Unit (ALU)

The dsPIC33FJXXXMCX06A/X08A/X10A 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 Borrow and 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.

Refer to the "16-bit MCU and DSC Programmer's Reference Manual" (DS70157) for information on the SR bits affected by each instruction.

The dsPIC33FJXXXMCX06A/X08A/X10A CPU incorporates hardware support for both multiplication and division. This includes a dedicated hardware multiplier and support hardware for 16-bit-divisor division.

## 3.5.1 MULTIPLIER

Using the high-speed, 17-bit x 17-bit multiplier of the DSP engine, the ALU supports unsigned, signed or mixed sign operation in several MCU multiplication modes:

- 1. 16-bit x 16-bit signed
- 2. 16-bit x 16-bit unsigned
- 16-bit signed x 5-bit (literal) unsigned
- 4. 16-bit unsigned x 16-bit unsigned
- 5. 16-bit unsigned x 5-bit (literal) unsigned
- 6. 16-bit unsigned x 16-bit signed
- 7. 8-bit unsigned x 8-bit unsigned

#### 3.5.2 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. 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.

# 3.6 DSP Engine

The DSP engine consists of a high-speed, 17-bit x 17-bit multiplier, a barrel shifter and a 40-bit adder/ subtracter (with two target accumulators, round and saturation logic).

The dsPIC33FJXXXMCX06A/X08A/X10A devices are a single-cycle, instruction flow architecture; therefore, concurrent operation of the DSP engine with MCU instruction flow is not possible. However, some MCU ALU and DSP engine resources may be used concurrently by the same instruction (e.g., ED, EDAC).

The DSP engine also has the capability to perform inherent accumulator-to-accumulator operations which require no additional data. These instructions are ADD, SUB and NEG.

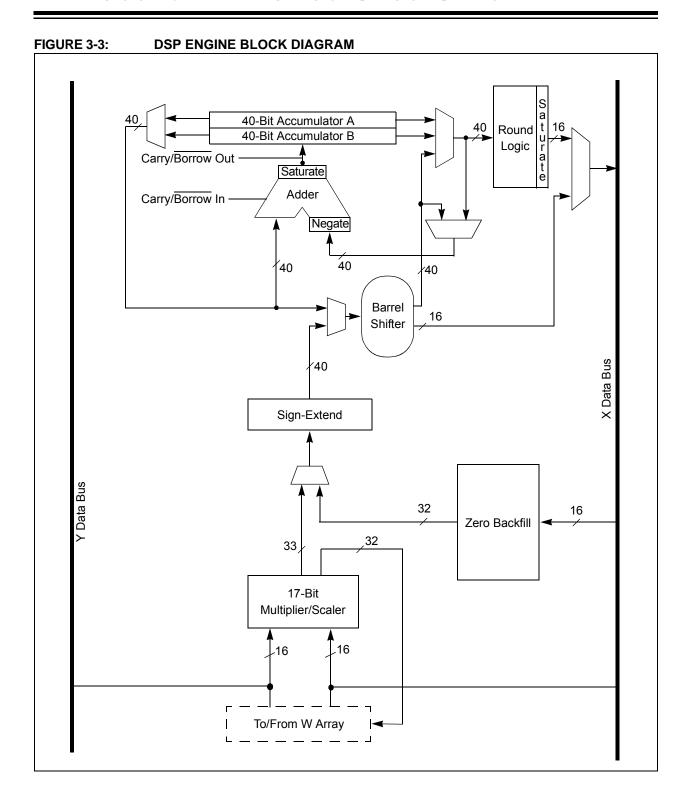
The DSP engine has various options selected through various bits in the CPU Core Control register (CORCON), as listed below:

- 1. Fractional or integer DSP multiply (IF)
- 2. Signed or unsigned DSP multiply (US)
- 3. Conventional or convergent rounding (RND)
- 4. Automatic saturation on/off for AccA (SATA)
- 5. Automatic saturation on/off for AccB (SATB)
- Automatic saturation on/off for writes to data memory (SATDW)
- 7. Accumulator Saturation mode selection (ACCSAT)

Table 2-1 provides a summary of DSP instructions. A block diagram of the DSP engine is shown in Figure 3-3.

TABLE 3-1: DSP INSTRUCTIONS SUMMARY

Instruction	Algebraic Operation	ACC Write Back
CLR	A = 0	Yes
ED	$A = (x - y)^2$	No
EDAC	$A = A + (x - y)^2$	No
MAC	$A = A + (x \bullet y)$	Yes
MAC	$A = A + x^2$	No
MOVSAC	No change in A	Yes
MPY	$A = x \cdot y$	No
MPY	$A = x^2$	No
MPY.N	$A = -x \bullet y$	No
MSC	$A = A - x \bullet y$	Yes



#### 3.6.1 **MULTIPLIER**

The 17-bit x 17-bit multiplier is capable of signed or unsigned operation and can multiplex its output using a scaler to support either 1.31 fractional (Q31) or 32-bit integer results. Unsigned operands are zero-extended into the 17th bit of the multiplier input value. Signed operands are sign-extended into the 17th bit of the multiplier input value. The output of the 17-bit x 17-bit multiplier/scaler is a 33-bit value which is sign-extended to 40 bits. Integer data is inherently represented as a signed two's complement value, where the MSb is defined as a sign bit. Generally speaking, the range of an N-bit two's complement integer is  $-2^{N-1}$ to  $2^{N-1}$  – 1. For a 16-bit integer, the data range is -32768 (0x8000) to 32767 (0x7FFF) including 0. For a 32-bit integer, the data range is -2,147,483,648 (0x8000 0000) to 2,147,483,647 (0x7FFF FFFF).

When the multiplier is configured for fractional multiplication, the data is represented as a two's complement fraction, where the MSb is defined as a sign bit and the radix point is implied to lie just after the sign bit (QX format). The range of an N-bit two's complement fraction with this implied radix point is -1.0 to  $(1 - 2^{1-N})$ . For a 16-bit fraction, the Q15 data range is -1.0 (0x8000) to 0.999969482 (0x7FFF) including 0 and has a precision of 3.01518 x 10<sup>-5</sup>. In Fractional mode, the 16 x 16 multiply operation generates a 1.31 product which has a precision of  $4.65661 \times 10^{-10}$ .

The same multiplier is used to support the MCU multiply instructions which include integer 16-bit signed, unsigned and mixed sign multiplies.

The MUL instruction may be directed to use byte or word-sized operands. Byte operands will direct a 16-bit result, and word operands will direct a 32-bit result to the specified register(s) in the W array.

#### 3.6.2 DATA ACCUMULATORS AND ADDER/SUBTRACTER

The data accumulator consists of a 40-bit adder/ subtracter with automatic sign extension logic. It can select one of two accumulators (A or B) as its pre-accumulation source and post-accumulation destination. For the ADD and LAC instructions, the data to be accumulated or loaded can be optionally scaled via the barrel shifter prior to accumulation.

#### 3.6.2.1 Adder/Subtracter, Overflow and Saturation

The adder/subtracter is a 40-bit adder with an optional zero input into one side, and either true, or complement data into the other input. In the case of addition, the Carry/Borrow input is active-high and the other input is true data (not complemented); whereas in the case of subtraction, the Carry/Borrow input is active-low and the other input is complemented. The adder/subtracter generates Overflow Status bits, SA/SB and OA/OB, which are latched and reflected in the STATUS register:

- Overflow from bit 39: this is a catastrophic overflow in which the sign of the accumulator is destroyed.
- · Overflow into guard bits 32 through 39: this is a recoverable overflow. This bit is set whenever all the guard bits are not identical to each other.

The adder has an additional saturation block which controls accumulator data saturation, if selected. It uses the result of the adder, the Overflow Status bits described above and the SAT<A:B> (CORCON<7:6>) and ACCSAT (CORCON<4>) mode control bits to determine when and to what value to saturate.

Six STATUS register bits have been provided to support saturation and overflow; they are:

- 1. OA: AccA overflowed into guard bits
- AccB overflowed into guard bits
  - AccA saturated (bit 31 overflow and saturation)

AccA overflowed into guard bits and saturated (bit 39 overflow and saturation)

AccB saturated (bit 31 overflow and saturation)

AccB overflowed into guard bits and saturated (bit 39 overflow and saturation)

- Logical OR of OA and OB
- SAB: Logical OR of SA and SB

The OA and OB bits are modified each time data passes through the adder/subtracter. When set, they indicate that the most recent operation has overflowed into the accumulator guard bits (bits 32 through 39). The OA and OB bits can also optionally generate an arithmetic warning trap when they and the corresponding Overflow Trap Flag Enable bits (OVATE, OVBTE) in the INTCON1 register (refer to Section 7.0 "Interrupt Controller") are set. This allows the user to take immediate action, for example, to correct system gain.

The SA and SB bits are modified each time data passes through the adder/subtracter, but can only be cleared by the user. When set, they indicate that the accumulator has overflowed its maximum range (bit 31 for 32-bit saturation or bit 39 for 40-bit saturation) and will be saturated (if saturation is enabled). When saturation is not enabled, SA and SB default to bit 39 overflow, and thus, indicate that a catastrophic overflow has occurred. If the COVTE bit in the INTCON1 register is set, SA and SB bits will generate an arithmetic warning trap when saturation is disabled.

The Overflow and Saturation Status bits can optionally be viewed in the STATUS Register (SR) as the logical OR of OA and OB (in bit OAB), and the logical OR of SA and SB (in bit SAB). This allows programmers to check one bit in the STATUS register to determine if either accumulator has overflowed or one bit to determine if either accumulator has saturated. This would be useful for complex number arithmetic, which typically uses both the accumulators.

The device supports three Saturation and Overflow modes:

- 1. Bit 39 Overflow and Saturation:
  - When bit 39 overflow and saturation occurs, the saturation logic loads the maximally positive 9.31 (0x7FFFFFFFFF) or maximally negative 9.31 value (0x8000000000) into the target accumulator. The SA or SB bit is set and remains set until cleared by the user. This is referred to as 'super saturation' and provides protection against erroneous data or unexpected algorithm problems (e.g., gain calculations).
- 2. Bit 31 Overflow and Saturation:
  - When bit 31 overflow and saturation occurs, the saturation logic then loads the maximally positive 1.31 value (0x007FFFFFFF) or maximally negative 1.31 value (0x0080000000) into the target accumulator. The SA or SB bit is set and remains set until cleared by the user. When this Saturation mode is in effect, the guard bits are not used (so the OA, OB or OAB bits are never set).
- 3. Bit 39 Catastrophic Overflow:
  - The bit 39 Overflow Status bit from the adder is used to set the SA or SB bit, which remains set until cleared by the user. No saturation operation is performed and the accumulator is allowed to overflow (destroying its sign). If the COVTE bit in the INTCON1 register is set, a catastrophic overflow can initiate a trap exception.

# 3.6.2.2 Accumulator 'Write Back'

The MAC class of instructions (with the exception of MPY, MPY.N, ED and EDAC) can optionally write a rounded version of the high word (bits 31 through 16) of the accumulator that is not targeted by the instruction into data space memory. The write is performed across the X bus into combined X and Y address space. The following addressing modes are supported:

- 1. W13, Register Direct:
  - The rounded contents of the non-target accumulator are written into W13 as a 1.15 fraction.
- [W13]+ = 2, Register Indirect with Post-Increment: The rounded contents of the non-target accumulator are written into the address pointed to by W13 as a 1.15 fraction. W13 is then incremented by 2 (for a word write).

#### 3.6.2.3 Round Logic

The round logic is a combinational block which performs a conventional (biased) or convergent (unbiased) round function during an accumulator write (store). The Round mode is determined by the state of the RND bit in the CORCON register. It generates a 16-bit, 1.15 data value which is passed to the data space write saturation logic. If rounding is not indicated by the instruction, a truncated 1.15 data value is stored and the least significant word is simply discarded.

Conventional rounding zero-extends bit 15 of the accumulator and adds it to the ACCxH word (bits 16 through 31 of the accumulator). If the ACCxL word (bits 0 through 15 of the accumulator) is between 0x8000 and 0xFFFF (0x8000 included), ACCxH is incremented. If ACCxL is between 0x0000 and 0x7FFF, ACCxH is left unchanged. A consequence of this algorithm is that over a succession of random rounding operations, the value tends to be biased slightly positive.

Convergent (or unbiased) rounding operates in the same manner as conventional rounding, except when ACCxL equals 0x8000. In this case, the Least Significant bit (bit 16 of the accumulator) of ACCxH is examined. If it is '1', ACCxH is incremented. If it is '0', ACCxH is not modified. Assuming that bit 16 is effectively random in nature, this scheme removes any rounding bias that may accumulate.

The SAC and SAC.R instructions store either a truncated (SAC) or rounded (SAC.R) version of the contents of the target accumulator to data memory via the X bus, subject to data saturation (see Section 3.6.2.4 "Data Space Write Saturation"). For the MAC class of instructions, the accumulator write-back operation will function in the same manner, addressing combined MCU (X and Y) data space though the X bus. For this class of instructions, the data is always subject to rounding.

## 3.6.2.4 Data Space Write Saturation

In addition to adder/subtracter saturation, writes to data space can also be saturated – but without affecting the contents of the source accumulator. The data space write saturation logic block accepts a 16-bit, 1.15 fractional value from the round logic block as its input, together with overflow status from the original source (accumulator) and the 16-bit round adder. These inputs are combined and used to select the appropriate 1.15 fractional value as output to write to data space memory.

If the SATDW bit in the CORCON register is set, data (after rounding or truncation) is tested for overflow and adjusted accordingly. For input data greater than 0x007FFF, data written to memory is forced to the maximum positive 1.15 value, 0x7FFF. For input data less than 0xFF8000, data written to memory is forced to the maximum negative 1.15 value, 0x8000. The Most Significant bit of the source (bit 39) is used to determine the sign of the operand being tested.

If the SATDW bit in the CORCON register is not set, the input data is always passed through unmodified under all conditions.

#### 3.6.3 BARREL SHIFTER

The barrel shifter is capable of performing up to 16-bit arithmetic or logic right shifts, or up to 16-bit left shifts in a single cycle. The source can be either of the two DSP accumulators or the X bus (to support multi-bit shifts of register or memory data).

The shifter requires a signed binary value to determine both the magnitude (number of bits) and direction of the shift operation. A positive value shifts the operand right. A negative value shifts the operand left. A value of '0' does not modify the operand.

The barrel shifter is 40 bits wide, thereby obtaining a 40-bit result for DSP shift operations and a 16-bit result for MCU shift operations. Data from the X bus is presented to the barrel shifter between bit positions 16 to 31 for right shifts and between bit positions 0 to 16 for left shifts.

NOTES:	

## 4.0 MEMORY ORGANIZATION

Note 1: This data sheet summarizes the features of the dsPIC33FJXXXMCX06A/ X08A/X10A family of devices. However, it is not intended to be a comprereference hensive source. complement the information in this data sheet, refer to Section 3. "Data Memory" (DS70202) and Section 4. "Program Memory" (DS70203) in the "dsPIC33F/PIC24H Family Reference Manual", which is available from the Microchip web site (www.microchip.com).

The dsPIC33FJXXXMCX06A/X08A/X10A architecture features separate program and data memory spaces, and buses. This architecture also allows the direct access of program memory from the data space during code execution.

#### 4.1 Program Address Space

The program address memory space of the dsPIC33FJXXXMCX06A/X08A/X10A 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 4.6 "Interfacing Program and Data Memory Spaces".

User access to the program memory space is restricted to the lower half of the address range (0x000000 to 0x7FFFFF). The exception is the use of TBLRD/TBLWT operations, which use TBLPAG<7> to permit access to the Configuration bits and Device ID sections of the configuration memory space. Memory usage for the dsPIC33FJXXXMCX06A/X08A/X10A family of devices is shown in Figure 4-1.

FIGURE 4-1: PROGRAM MEMORY MAP FOR dsPIC33FJXXXMCX06A/X08A/X10A DEVICES

	dsPIC33FJ64MCXXXA	dsPIC33FJ128MCXXXA	dsPIC33FJ256MCXXXA
<u> </u>	GOTO Instruction	GOTO Instruction	GOTO Instruction 0x00000
User Memory Space	Reset Address	Reset Address	Reset Address 0x00000
	Interrupt Vector Table	Interrupt Vector Table	Interrupt Vector Table 0x0000
	Reserved	Reserved	Reserved 0x00010
	Alternate Vector Table	Alternate Vector Table	Alternate Vector Table 0x00010
	User Program Flash Memory (22K instructions)	User Program Flash Memory (44K instructions)	User Program Flash Memory (88K instructions)
<u> </u>			0x0157l
2			0x01580
Use	Unimplemented (Read '0's)	Unimplemented (Read '0's)	Unimplemented (Read '0's)
	Reserved	Reserved	Reserved
Configuration Memory Space	Device Configuration	Device Configuration	Device Configuration 0xF800
	Registers	Registers	Registers 0xF800 0xF800
	Reserved	Reserved	Reserved
	DEVID (2)	DEVID (2)	DEVID (2)
V	DEVID (2)		DEVID (2) 0xFFFF

# 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).

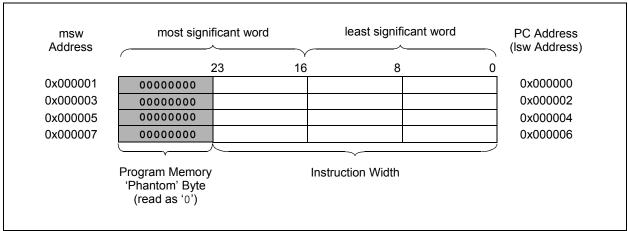
Program memory addresses are always word-aligned on the lower word, and addresses are incremented or decremented by two during code execution. This arrangement also provides compatibility with data memory space addressing and makes it possible to access data in the program memory space.

#### 4.1.2 INTERRUPT AND TRAP VECTORS

All dsPIC33FJXXXMCX06A/X08A/X10A devices reserve the addresses between 0x00000 and 0x000200 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 0x000000, with the actual address for the start of code at 0x000002.

dsPIC33FJXXXMCX06A/X08A/X10A devices also have two interrupt vector tables located from 0x000004 to 0x0000FF and 0x000100 to 0x0001FF. These vector tables allow each of the many device interrupt sources to be handled by separate Interrupt Service Routines (ISRs). A more detailed discussion of the interrupt vector tables is provided in Section 7.1 "Interrupt Vector Table".





### 4.2 Data Address Space

The dsPIC33FJXXXMCX06A/X08A/X10A CPU has a separate 16-bit wide data memory space. The data space is accessed using separate Address Generation Units (AGUs) for read and write operations. Data memory maps of devices with different RAM sizes are shown in Figure 4-3 through Figure 4-5.

All Effective Addresses (EAs) in the data memory space are 16 bits wide and point to bytes within the data space. This arrangement gives a data space address range of 64 Kbytes or 32K words. The lower half of the data memory space (that is, when EA<15> = 0) is used for implemented memory addresses, while the upper half (EA<15> = 1) is reserved for the Program Space Visibility area (see Section 4.6.3 "Reading Data from Program Memory Using Program Space Visibility").

dsPIC33FJXXXMCX06A/X08A/X10A devices implement a total of up to 30 Kbytes of data memory. Should an EA point to a location outside of this area, an all-zero word or byte will be returned.

### 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 EAs resolve to bytes. The Least Significant Bytes of each word have even addresses, while the Most Significant Bytes have odd addresses.

# 4.2.2 DATA MEMORY ORGANIZATION AND ALIGNMENT

To maintain backward compatibility with PIC® microcontrollers and improve data space memory usage efficiency, the dsPIC33FJXXXMCX06A/X08A/X10A instruction set supports both word and byte operations. As a consequence of byte accessibility, all Effective Address 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 that 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 is 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 does 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 Least Significant Byte. The Most Significant Byte is not modified

A sign-extend instruction (SE) 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.

### 4.2.3 SFR SPACE

The first 2 Kbytes of the Near Data Space, from 0x0000 to 0x07FF, is primarily occupied by Special Function Registers (SFRs). These are used by the dsPIC33FJXXXMCX06A/X08A/X10A 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 module. Much of the SFR space contains unused addresses; these are read as '0'.

Note:

The actual set of peripheral features and interrupts varies by the device. Please refer to the corresponding device tables and pinout diagrams for device-specific information.

### 4.2.4 NEAR DATA SPACE

The 8-Kbyte area between 0x0000 and 0x1FFF 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. Additionally, the whole data space is addressable using MoV instructions, which support Memory Direct Addressing mode with a 16-bit address field, or by using Indirect Addressing mode using a working register as an Address Pointer.

FIGURE 4-3: DATA MEMORY MAP FOR dsPIC33FJXXXMCX06A/X08A/X10A DEVICES WITH 8-Kbyte RAM

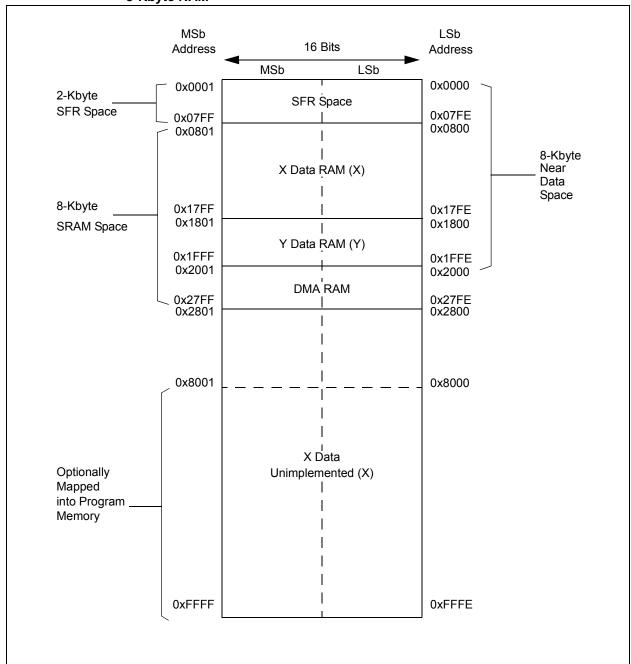


FIGURE 4-4: DATA MEMORY MAP FOR dsPIC33FJXXXMCX06A/X08A/X10A DEVICES WITH 16-Kbyte RAM

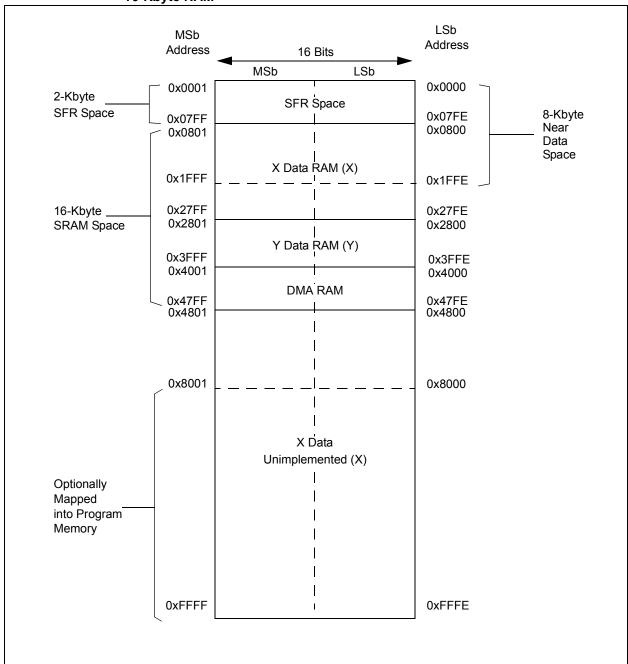
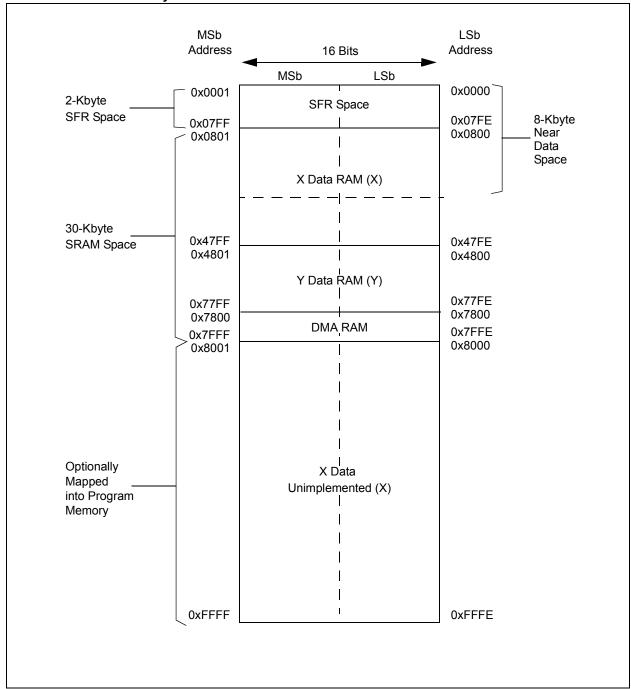


FIGURE 4-5: DATA MEMORY MAP FOR dsPIC33FJXXXMCX06A/X08A/X10A DEVICES WITH 30-Kbyte RAM



### 4.2.5 X AND Y DATA SPACES

The core has two data spaces: X and Y. These data spaces can be considered either separate (for some DSP instructions) or as one unified, linear address range (for MCU instructions). The data spaces are accessed using two Address Generation Units (AGUs) and separate data paths. This feature allows certain instructions to concurrently fetch two words from RAM, thereby enabling efficient execution of DSP algorithms, such as Finite Impulse Response (FIR) filtering and Fast Fourier Transform (FFT).

The X data space is used by all instructions and supports all addressing modes. There are separate read and write data buses for X data space. The X read data bus is the read data path for all instructions that view data space as combined X and Y address space. It is also the X data prefetch path for the dual operand DSP instructions (MAC class).

The Y data space is used in concert with the X data space by the MAC class of instructions (CLR, ED, EDAC, MAC, MOVSAC, MPY, MPY. N and MSC) to provide two concurrent data read paths.

Both the X and Y data spaces support Modulo Addressing mode for all instructions, subject to addressing mode restrictions. Bit-Reversed Addressing mode is only supported for writes to X data space.

All data memory writes, including in DSP instructions, view data space as combined X and Y address space. The boundary between the X and Y data spaces is device-dependent and is not user-programmable.

All Effective Addresses are 16 bits wide and point to bytes within the data space. Therefore, the data space address range is 64 Kbytes, or 32K words, though the implemented memory locations vary by device.

### 4.2.6 DMA RAM

Every dsPIC33FJXXXMCX06A/X08A/X10A device contains 2 Kbytes of dual ported DMA RAM located at the end of Y data space. Memory location is part of Y data RAM and is in the DMA RAM space, and is accessible simultaneously by the CPU and the DMA controller module. DMA RAM is utilized by the DMA controller to store data to be transferred to various peripherals using DMA, as well as data transferred from various peripherals using DMA. The DMA RAM can be accessed by the DMA controller without having to steal cycles from the CPU.

When the CPU and the DMA controller attempt to concurrently write to the same DMA RAM location, the hardware ensures that the CPU is given precedence in accessing the DMA RAM location. Therefore, the DMA RAM provides a reliable means of transferring DMA data without ever having to stall the CPU.

TABLE 4-1:	CPU CORE REGISTERS MAP

SFR Name	SFR 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 R	egister 0								xxxx
WREG1	0002								Working R	egister 1								xxxx
WREG2	0004								Working R	egister 2								xxxx
WREG3	0006								Working R	egister 3								xxxx
WREG4	8000								Working R	egister 4								xxxx
WREG5	000A								Working R	egister 5								xxxx
WREG6	000C								Working R	egister 6								xxxx
WREG7	000E								Working R	egister 7								xxxx
WREG8	0010								Working R	egister 8								xxxx
WREG9	0012								Working R	egister 9								xxxx
WREG10	0014								Working Re	egister 10								xxxx
WREG11	0016								Working Re	egister 11								xxxx
WREG12	0018								Working Re	egister 12								xxxx
WREG13	001A														xxxx			
WREG14	001C		Working Register 14 xxxx												xxxx			
WREG15	001E														0800			
SPLIM	0020							Sta	ck Pointer L	imit Registe	er							xxxx
ACCAL	0022							Accum	ulator A Lov	w Word Reg	jister							0000
ACCAH	0024							Accum	ulator A Hig	h Word Reg	gister							0000
ACCAU	0026							Accum	ılator A Upp	er Word Re	gister							0000
ACCBL	0028							Accum	ulator B Lov	w Word Reg	jister							0000
ACCBH	002A							Accum	ulator B Hig	h Word Reg	gister							0000
ACCBU	002C							Accum	ılator B Upp	er Word Re	gister							0000
PCL	002E							Progran	n Counter Lo	ow Word Re	egister							0000
PCH	0030	_	_	_	_	_	_	_	_			Progra	am Counter I	High Byte R	legister			0000
TBLPAG	0032	-	_	_	_	_	_	_	_			Table	Page Addres	ss Pointer R	Register			0000
PSVPAG	0034	_	_	_	_	_	_	_	_		Progr	am Memor	y Visibility Pa	age Address	s Pointer R	egister		0000
RCOUNT	0036							Repe	eat Loop Co	unter Regis	ter							xxxx
DCOUNT	0038								DCOUNT	<15:0>								xxxx
DOSTARTL	003A							DOS	TARTL<15	:1>							0	xxxx
DOSTARTH	003C	_	_	_	_	_	_	_	_	_	_			DOSTAR	TH<5:0>		•	00xx
DOENDL	003E			•				DO	ENDL<15:1	>							0	xxxx
DOENDH	0040	_	_	_	_	_	_	_	_	_	_			DOE	NDH		•	00xx
SR	0042	OA	ОВ	SA	SB	OAB	SAB	DA	DC	IPL2	IPL1	IPL0	RA	N	OV	Z	С	0000
CORCON	0044	_	_	_	US	EDT		DL<2:0>		SATA	SATB	SATDW	ACCSAT	IPL3	PSV	RND	IF	0020
MODCON	0046	XMODEN	YMODEN	_	_		BWM	1<3:0>			YWN	l<3:0>			XWM	<3:0>	•	0000
XMODSRT	0048								XS<15:1>					•			0	xxxx
XMODEND	004A								XE<15:1>								1	xxxx

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

### TABLE 4-1: CPU CORE REGISTERS MAP (CONTINUED)

	_			_	• -		- ,											
SFR Name	SFR 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
YMODSRT	004C							,	YS<15:1>								0	xxxx
YMODEND	004E				YE<15:1> 1										xxxx			
XBREV	0050	BREN			VP 44.0										xxxx			
DISICNT	0052	_	_						Disab	e Interrupts	Counter F	Register						xxxx
BSRAM	0750	_	_	_	_	_	_	_	_	_	_	_	_	_	IW_BSR	IR_BSR	RL_BSR	0000
SSRAM	0752	_	_	_	_	_	_	_	_	_	_	_	_	_	IW_SSR	IR_SSR	RL_SSR	0000

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

### TABLE 4-2: CHANGE NOTIFICATION REGISTER MAP FOR dsPIC33FJXXXMCX10A DEVICES

SFR Name	SFR 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
CNEN1	0060	CN15IE	CN14IE	CN13IE	CN12IE	CN11IE	CN10IE	CN9IE	CN8IE	CN7IE	CN6IE	CN5IE	CN4IE	CN3IE	CN2IE	CN1IE	CN0IE	0000
CNEN2	0062	_	_	-	_	_	_	_	_	CN23IE	CN22IE	CN21IE	CN20IE	CN19IE	CN18IE	CN17IE	CN16IE	0000
CNPU1	0068	CN15PUE	CN14PUE	CN13PUE	CN12PUE	CN11PUE	CN10PUE	CN9PUE	CN8PUE	CN7PUE	CN6PUE	CN5PUE	CN4PUE	CN3PUE	CN2PUE	CN1PUE	CN0PUE	0000
CNPU2	006A	_	_	_	_	_	_		-	CN23PUE	CN22PUE	CN21PUE	CN20PUE	CN19PUE	CN18PUE	CN17PUE	CN16PUE	0000

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

### TABLE 4-3: CHANGE NOTIFICATION REGISTER MAP FOR dsPIC33FJXXXMCX08A DEVICES

SFR Name	SFR 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	AII Resets
CNEN1	0060	CN15IE	CN14IE	CN13IE	CN12IE	CN11IE	CN10IE	CN9IE	CN8IE	CN7IE	CN6IE	CN5IE	CN4IE	CN3IE	CN2IE	CN1IE	CN0IE	0000
CNEN2	0062	_	_	-	-	-	-	_	_	_	_	CN21IE	CN20IE	CN19IE	CN18IE	CN17IE	CN16IE	0000
CNPU1	0068	CN15PUE	CN14PUE	CN13PUE	CN12PUE	CN11PUE	CN10PUE	CN9PUE	CN8PUE	CN7PUE	CN6PUE	CN5PUE	CN4PUE	CN3PUE	CN2PUE	CN1PUE	CN0PUE	0000
CNPU2	006A	_	_	-	-	-	-	_	_	_	_	CN21PUE	CN20PUE	CN19PUE	CN18PUE	CN17PUE	CN16PUE	0000

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

### TABLE 4-4: CHANGE NOTIFICATION REGISTER MAP FOR dsPIC33FJXXXMCX06A DEVICES

SFR Name	SFR 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
CNEN1	0060	CN15IE	CN14IE	CN13IE	CN12IE	CN11IE	CN10IE	CN9IE	CN8IE	CN7IE	CN6IE	CN5IE	CN4IE	CN3IE	CN2IE	CN1IE	CN0IE	0000
CNEN2	0062	_	_	_	_	-	_	_	-	-	_	CN21IE	CN20IE	_	CN18IE	CN17IE	CN16IE	0000
CNPU1	0068	CN15PUE	CN14PUE	CN13PUE	CN12PUE	CN11PUE	CN10PUE	CN9PUE	CN8PUE	CN7PUE	CN6PUE	CN5PUE	CN4PUE	CN3PUE	CN2PUE	CN1PUE	CN0PUE	0000
CNPU2	006A	_	_	_	_	_	_			_	_	CN21PUE	CN20PUE	_	CN18PUE	CN17PUE	CN16PUE	0000

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

С	
Š	
~	
$\overline{C}$	١
O	
U594D	
<u>+</u>	١
Ç	,
+	
ă	ì
ō	•
)-page	)
1	
5	
_	

**TABLE 4-5**: INTERRUPT CONTROLLER REGISTER MAP

SFR Name	SFR 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	OVAERR	OVBERR	COVAERR	COVBERR	OVATE	OVBTE	COVTE	SFTACERR	DIV0ERR	DMACERR	MATHERR	ADDRERR	STKERR	OSCFAIL	_	0000
INTCON2	0082	ALTIVT	DISI	_	_	_		_	_	_	_	_	INT4EP	INT3EP	INT2EP	INT1EP	INT0EP	0000
IFS0	0084	_	DMA1IF	AD1IF	U1TXIF	U1RXIF	SPI1IF	SPI1EIF	T3IF	T2IF	OC2IF	IC2IF	DMA0IF	T1IF	OC1IF	IC1IF	INT0IF	0000
IFS1	0086	U2TXIF	U2RXIF	INT2IF	T5IF	T4IF	OC4IF	OC3IF	DMA2IF	IC8IF	IC7IF	AD2IF	INT1IF	CNIF	ı	MI2C1IF	SI2C1IF	0000
IFS2	8800	T6IF	DMA4IF	_	OC8IF	OC7IF	OC6IF	OC5IF	IC6IF	IC5IF	IC4IF	IC3IF	DMA3IF	C1IF	C1RXIF	SPI2IF	SPI2EIF	0000
IFS3	008A	FLTAIF	I	DMA5IF	ı	-	QEIIF	PWMIF	C2IF	C2RXIF	INT4IF	INT3IF	T9IF	T8IF	MI2C2IF	SI2C2IF	T7IF	0000
IFS4	008C	_	I		ı	-	_	-	_	C2TXIF	C1TXIF	DMA7IF	DMA6IF	_	U2EIF	U1EIF	FLTBIF	0000
IEC0	0094	_	DMA1IE	AD1IE	U1TXIE	U1RXIE	SPI1IE	SPI1EIE	T3IE	T2IE	OC2IE	IC2IE	DMA0IE	T1IE	OC1IE	IC1IE	INT0IE	0000
IEC1	0096	U2TXIE	U2RXIE	INT2IE	T5IE	T4IE	OC4IE	OC3IE	DMA2IE	IC8IE	IC7IE	AD2IE	INT1IE	CNIE	-	MI2C1IE	SI2C1IE	0000
IEC2	0098	T6IE	DMA4IE		OC8IE	OC7IE	OC6IE	OC5IE	IC6IE	IC5IE	IC4IE	IC3IE	DMA3IE	C1IE	C1RXIE	SPI2IE	SPI2EIE	0000
IEC3	009A	FLTAIE	I	DMA5IE	ı	-	QEIIE	PWMIE	C2IE	C2RXIE	INT4IE	INT3IE	T9IE	T8IE	MI2C2IE	SI2C2IE	T7IE	0000
IEC4	009C	_	I		ı	-	_	-	_	C2TXIE	C1TXIE	DMA7IE	DMA6IE	_	U2EIE	U1EIE	FLTBIE	0000
IPC0	00A4	_		T1IP<2:0>	•	-	(	-   -   - OC1IP<2:0>		-		IC1IP<2:0>		_	11	NT0IP<2:0>	•	4444
IPC1	00A6	_		T2IP<2:0>	•	-	(	OC2IP<2:0	>	-		IC2IP<2:0>		_	D	MA0IP<2:0	>	4444
IPC2	8A00	_	J	J1RXIP<2:0	)>	-	5	SPI1IP<2:0	)>	-	;	SPI1EIP<2:0	>	_		T3IP<2:0>		4444
IPC3	00AA	_	I		ı	-	D	MA1IP<2:	0>	-		AD1IP<2:0>	•	_	U	1TXIP<2:0	>	0444
IPC4	00AC	_		CNIP<2:0>	>	-	_	-	_	-	I	MI2C1IP<2:0	>	_	SI	2C1IP<2:0	>	4044
IPC5	00AE	_		IC8IP<2:0>	>	_		IC7IP<2:0	>	_		AD2IP<2:0>	•	_	11	NT1IP<2:0>	•	4444
IPC6	00B0	_		T4IP<2:0>	•	_	(	OC4IP<2:0	>	_		OC3IP<2:0>	•	_	D	MA2IP<2:0	>	4444
IPC7	00B2	_	l	J2TXIP<2:0	)>	-	U	I2RXIP<2:	0>	-		INT2IP<2:0>	>	_		T5IP<2:0>		4444
IPC8	00B4	_		C1IP<2:0>	•		C	1RXIP<2:	0>	ı		SPI2IP<2:0>	>	_	SI	PI2EIP<2:0	>	4444
IPC9	00B6	_		IC5IP<2:0>	>	_		IC4IP<2:0	>	_		IC3IP<2:0>		_	D	MA3IP<2:0	>	4444
IPC10	00B8	_		OC7IP<2:0	>	-	(	OC6IP<2:0	>	-		OC5IP<2:0>	•	_	_	C6IP<2:0>		4444
IPC11	00BA	_		T6IP<2:0>	•		D	MA4IP<2:	0>	ı	_	-	ı	_	C	)C8IP<2:0>	•	4404
IPC12	00BC	_		T8IP<2:0>	•	-	M	II2C2IP<2:	0>	-		SI2C2IP<2:0	>	_		T7IP<2:0>		4444
IPC13	00BE	_	(	C2RXIP<2:0	)>	-	ı	NT4IP<2:0	)>	-		INT3IP<2:0>	•	_		T9IP<2:0>		4444
IPC14	00C0	_	_	_	_	_		QEIIP<2:0	>	_		PWMIP<2:0	>	_		C2IP<2:0>		0444
IPC15	00C2	_		FLTAIP<2:0	)>	_	_	_	_	_		DMA5IP<2:0	>	_	_	_	_	4040
IPC16	00C4	_	_	_	_	_		J2EIP<2:0	>	_		U1EIP<2:0>	•	_	F	LTBIP<2:0>	>	0444
IPC17	00C6	_	(	C2TXIP<2:0	)>	_	C	1TXIP<2:	)>	_		DMA7IP<2:0	>	_	D	MA6IP<2:0	>	4444
INTTREG	00E0		-	_	_		ILR<	3:0>		_			VE	CNUM<6:0>				0000

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

	<b>TABLE 4-6:</b>	TIMER REC	GISTER MAP
--	-------------------	-----------	------------

	<del>7 U.</del>																	
SFR Name	SFR 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	0100								Timer1	Register								0000
PR1	0102								Period F	Register 1								FFFF
T1CON	0104	TON	_	TSIDL	_	_	_	_	_	_	TGATE	TCKP	S<1:0>	_	TSYNC	TCS	_	0000
TMR2	0106								Timer2	Register					-			0000
TMR3HLD	0108						Tim	er3 Holding	Register (for	32-bit time	r operations	only)						xxxx
TMR3	010A								Timer3	Register								0000
PR2	010C								Period F	Register 2								FFFF
PR3	010E								Period F	Register 3								FFFF
T2CON	0110	TON	_	TSIDL	_	_	_	_	_	_	TGATE	TCKP	S<1:0>	T32	_	TCS	_	0000
T3CON	0112	TON	_	TSIDL	_	_	_	_	_	_	TGATE	TCKP	S<1:0>	_	_	TCS	_	0000
TMR4	0114								Timer4	Register								0000
TMR5HLD	0116		Timer5 Holding Register (for 32-bit operations only) xxxxx Timer5 Register 0000										xxxx					
TMR5	0118		Timer5 Register 0000										0000					
PR4	011A								Period F	Register 4								FFFF
PR5	011C								Period F	Register 5								FFFF
T4CON	011E	TON	_	TSIDL	_	_	_	_	_	_	TGATE	TCKP	S<1:0>	T32	_	TCS	_	0000
T5CON	0120	TON	_	TSIDL	_	_	_	_	_	_	TGATE	TCKP	S<1:0>	_	_	TCS	_	0000
TMR6	0122								Timer6	Register								0000
TMR7HLD	0124						Т	imer7 Holdi	ng Register	(for 32-bit o	perations onl	y)						xxxx
TMR7	0126								Timer7	Register								0000
PR6	0128								Period F	Register 6								FFFF
PR7	012A								Period F	Register 7								FFFF
T6CON	012C	TON	_	TSIDL	_	_	_	_	_	_	TGATE	TCKP	S<1:0>	T32	_	TCS	_	0000
T7CON	012E	TON	_	TSIDL	_	_	_	_	_	_	TGATE	TCKP	S<1:0>	_	_	TCS	_	0000
TMR8	0130								Timer8	Register								0000
TMR9HLD	0132						Т	imer9 Holdi	ng Register	(for 32-bit o	perations onl	y)						xxxx
TMR9	0134								Timer9	Register								0000
PR8	0136								Period F	Register 8								FFFF
PR9	0138									Register 9								FFFF
T8CON	013A	TON	_	TSIDL	_	_	_	_	_	_	TGATE	TCKP	S<1:0>	T32	_	TCS	_	0000
T9CON	013C	TON	_	TSIDL	_	_	_	_	_	_	TGATE	TCKP	S<1:0>	_	_	TCS	_	0000
L																		

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

TABLE 4-7: INPUT CAPTURE REGISTER MAP

SFR Name	SFR 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
IC1BUF	0140								Input 1 Ca	pture Regis	ter							xxxx
IC1CON	0142	_	_	ICSIDL	_	_	_	_	_	ICTMR	ICI<	1:0>	ICOV	ICBNE		ICM<2:0>		0000
IC2BUF	0144								Input 2 Ca	pture Regis	ter							xxxx
IC2CON	0146	-	_	ICSIDL	_	1	_	1	_	ICTMR	ICI<	1:0>	ICOV	ICBNE		ICM<2:0>		0000
IC3BUF	0148								Input 3 Ca	pture Regis								xxxx
IC3CON	014A	_	_	ICSIDL	_	_	_	_	_	ICTMR	ICI<	1:0>	ICOV	ICBNE		ICM<2:0>		0000
IC4BUF	014C								Input 4 Ca	pture Regis	ter							xxxx
IC4CON	014E		ı	ICSIDL	_	1	_		_	ICTMR	ICI<	1:0>	ICOV	ICBNE		ICM<2:0>		0000
IC5BUF	0150								Input 5 Ca	pture Regis	ter							xxxx
IC5CON	0152		ı	ICSIDL	_	1	_		_	ICTMR	ICI<	1:0>	ICOV	ICBNE		ICM<2:0>		0000
IC6BUF	0154								Input 6 Ca	pture Regis	ter							xxxx
IC6CON	0156		ı	ICSIDL	_	1	_		_	ICTMR	ICI<	1:0>	ICOV	ICBNE		ICM<2:0>		0000
IC7BUF	0158								Input 7 Ca	pture Regis	ter							xxxx
IC7CON	015A	_	_	ICSIDL	_	_	_	_	_	ICTMR	ICI<	1:0>	ICOV	ICBNE		ICM<2:0>		0000
IC8BUF	015C								Input 8 Ca	pture Regis	ter							xxxx
IC8CON	015E	_	_	ICSIDL	_	_	_	_	_	ICTMR	ICI<	1:0>	ICOV	ICBNE		ICM<2:0>	·	0000

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

IABLE 4-	·8: C	טפוטכ	I COMI	AKE K	EGIST	EK WA	_

SFR Name	SFR 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
OC1RS	0180							Out	put Compar	e 1 Seconda	ary Register	r						xxxx
OC1R	0182								Output Co	mpare 1 Re	egister							xxxx
OC1CON	0184	_	_	OCSIDL	_	_	_	_	_	_	_	_	OCFLT	OCTSEL		OCM<2:0>		0000
OC2RS	0186							Out	put Compar	e 2 Seconda	ary Register	r						xxxx
OC2R	0188								Output Co	mpare 2 Re	egister							xxxx
OC2CON	018A	_	_	OCSIDL	_	_	_	_	_	-	-	_	OCFLT	OCTSEL		OCM<2:0>		0000
OC3RS	018C							Out	put Compar	e 3 Seconda	ary Register	r						xxxx
OC3R	018E								Output Co	mpare 3 Re	egister							xxxx
OC3CON	0190	_	_	OCSIDL	_	_	_	_	_	-	-	_	OCFLT	OCTSEL		OCM<2:0>		0000
OC4RS	0192							Out	put Compar	e 4 Seconda	ary Register	r						xxxx
OC4R	0194								Output Co	mpare 4 Re	egister							xxxx
OC4CON	0196	_	_	OCSIDL	-	-	-	_		I		_	OCFLT	OCTSEL		OCM<2:0>		0000
OC5RS	0198							Out	put Compar	e 5 Seconda	ary Register	r						xxxx
OC5R	019A								Output Co	mpare 5 Re	egister							xxxx
OC5CON	019C	_	_	OCSIDL	-	-		_				_	OCFLT	OCTSEL		OCM<2:0>		0000
OC6RS	019E							Out	put Compar	e 6 Seconda	ary Register	r						xxxx
OC6R	01A0								Output Co	mpare 6 Re	egister							xxxx
OC6CON	01A2	_	_	OCSIDL	_	_	_	_	_	_	_	_	OCFLT	OCTSEL		OCM<2:0>		0000
OC7RS	01A4							Out	put Compar	e 7 Seconda	ary Register	r						xxxx
OC7R	01A6								Output Co	mpare 7 Re	egister							xxxx
OC7CON	01A8	_	_	OCSIDL	_	_	_	_	_	_	_	_		0000				
OC8RS	01AA		·	·	·		·	Out	put Compar	e 8 Seconda	ary Register	r						xxxx
OC8R	01AC								Output Co	mpare 8 Re	egister							xxxx
OC8CON	01AE	_	_	OCSIDL	_	_	_	_	_	_	_	_	OCFLT	OCTSEL		OCM<2:0>		0000

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

TABLE 4-9: 8-OUTPUT PWM REGISTER MAP

SFR 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	Reset State
P1TCON	01C0	PTEN	_	PTSIDL	_	_	_	_	_		PTOPS	S<3:0>		PTCKP	'S<1:0>	PTMOI	D<1:0>	0000 0000 0000 0000
P1TMR	01C2	PTDIR						F	PWM Time	Count Val	ue Registe	r						0000 0000 0000 0000
P1TPER	01C4	_						F	PWM Time	Base Perio	od Registe	r						0000 0000 0000 0000
P1SECMP	01C6	SEVTDIR						PW	M Special	Event Com	pare Regi	ster						0000 0000 0000 0000
PWM1CON1	01C8	-	_	-	ı	PMOD4	PMOD3	PMOD2	PMOD1	PEN4H	PEN3H	PEN2H	PEN1H	PEN4L	PEN1L	0000 0000 1111 1111		
PWM1CON2	01CA	_	_	_	_		SEVOP	S<3:0>		-	1	_	_	_	IUE	UDIS	0000 0000 0000 0000	
P1DTCON1	01CC	DTBPS	S<1:0>			DTB<	<5:0>			DTAPS	S<1:0>			DTA		0000 0000 0000 0000		
P1DTCON2	01CE	_	_	_	_	_	_	1	_	DTS4A	DTS4I	DTS3A	DTS3I	DTS2A	DTS2I	DTS1A	DTS1I	0000 0000 0000 0000
P1FLTACON	01D0	FAOV4H	FAOV4L	FAOV3H	FAOV3L	FAOV2H	FAOV2L	FAOV1H	FAOV1L	FLTAM	1	_	_	FAEN4	FAEN3	FAEN2	FAEN1	0000 0000 0000 0000
P1FLTBCON	01D2	FBOV4H	FBOV4L	FBOV3H	FBOV3L	FBOV2H	FBOV2L	FBOV1H	FBOV1L	FLTBM	1	_	_	FBEN4	FBEN3	FBEN2	FBEN1	0000 0000 0000 0000
P10VDCON	01D4	POVD4H	POVD4L	POVD3H	POVD3L	POVD2H	POVD2L	POVD1H	POVD1L	POUT4H	POUT4L	POUT3H	POUT3L	POUT2H	POUT2L	POUT1H	POUT1L	1111 1111 0000 0000
P1DC1	01D6							PWN	□ Duty Cyc	le #1 Regi	ster							0000 0000 0000 0000
P1DC2	01D8							PWN	□ Duty Cyc	le #2 Regi	ster			0000 0000 0000 0000				
P1DC3	01DA							PWN	□ Duty Cyc	le #3 Regi	ster			0000 0000 0000 0000				
P1DC4	01DC	PWM Duty Cycle #4 Register										0000 0000 0000 0000						

**Legend:** u = uninitialized bit, — = unimplemented, read as '0'

### **TABLE 4-10: QEI REGISTER MAP**

SFR 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	ı	Reset Sta	ite
QEI1CON	01E0	CNTERR	_	QEISIDL	INDX	UPDN	Q	EIM<2:0	)>	SWPAB	PCDOUT	TQGATE	TQCKP	S<1:0>	POSRES	TQCS	UPDN_SRC	0000	0000 00	00 0000
DFLT1CON	01E2	ı	_	ı	_	_	IMV<	:1:0>	CEID	QEOUT		QECK<2:0>		-	_	_	ı	0000	0000 00	00 0000
POS1CNT	01E4								Po	sition Cou	nter<15:0>							0000	0000 00	00 0000
MAX1CNT	01E6		Maximum Counte(<15:0>														1111	1111 11	11 1111	

**Legend:** u = uninitialized bit, — = unimplemented, read as '0'

### TABLE 4-11: I2C1 REGISTER MAP

SFR Name	SFR 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	AII Resets	
I2C1RCV	0200	_	_	-	_	1	_	_					I2C1 Recei	ve Register				0000	
I2C1TRN	0202	_	_	ı	_		ı	_	ı				I2C1 Trans	mit Register	r			00FF	
I2C1BRG	0204	_	_	_	_	-	_	_		Baud Rate Generator Register								0000	
I2C1CON	0206	I2CEN	_	I2CSIDL	SCLREL	IPMIEN	A10M	DISSLW	SMEN	GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	1000	
I2C1STAT	0208	ACKSTAT	TRSTAT	_	_	-	BCL	GCSTAT	ADD10	IWCOL	I2COV	D_A	Р	S	R_W	RBF	TBF	0000	
I2C1ADD	020A	_	_	_	_	-	_					0000							
I2C1MSK	020C	_	_	_	_	_	_			I2C1 Address Register I2C1 Address Mask Register									

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

### TABLE 4-12: I2C2 REGISTER MAP

SFR Name	SFR 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	
I2C2RCV	0210	_	_	_	_	_	_	_	_				I2C2 Recei	ive Register				0000	
I2C2TRN	0212	_	_	_	_	_	_	_	-				I2C2 Trans	mit Register	•			00FF	
I2C2BRG	0214	_	_	_	_	_	_	_				Baud Rat	e Generato	r Register				0000	
I2C2CON	0216	I2CEN	_	I2CSIDL	SCLREL	IPMIEN	A10M	DISSLW	SMEN	GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	1000	
I2C2STAT	0218	ACKSTAT	TRSTAT	_	_	_	BCL	GCSTAT	ADD10	IWCOL	I2COV	D_A	Р	S	R_W	RBF	TBF	0000	
I2C2ADD	021A	_	_	_	_	_	_					0000							
I2C2MSK	021C	_	_	_		_	_			I2C2 Address Mask Register									

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

### TABLE 4-13: UART1 REGISTER MAP

SFR Name	SFR 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	0220	UARTEN	_	USIDL	IREN	RTSMD	_	UEN1	UEN0	WAKE	LPBACK	ABAUD	URXINV	BRGH	PDSE	L<1:0>	STSEL	0000	
U1STA	0222	UTXISEL1	UTXINV	UTXISEL0	_	UTXBRK	UTXEN	UTXBF	TRMT	URXISE	EL<1:0>	ADDEN	RIDLE	PERR	FERR	OERR	URXDA	0110	
U1TXREG	0224	_	_	_	_	_	_	_				UART1	Transmit Re	egister				xxxx	
U1RXREG	0226	ı	_	_	_	_	-	-	HADTA Despite Desirtes									0000	
U1BRG	0228							Baud Rate Generator Prescaler											

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

### **TABLE 4-14: UART2 REGISTER MAP**

SFR Name	SFR 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
U2MODE	0230	UARTEN	_	USIDL	IREN	RTSMD	_	UEN1	UEN0	WAKE	LPBACK	ABAUD	URXINV	BRGH	PDSE	L<1:0>	STSEL	0000
U2STA	0232	UTXISEL1	UTXINV	UTXISEL0	_	UTXBRK	UTXEN	UTXBF	TRMT	URXISE	EL<1:0>	ADDEN	RIDLE	PERR	FERR	OERR	URXDA	0110
U2TXREG	0234	_	_	_	_	_	_	_				UART2	Transmit R	egister				xxxx
U2RXREG	0236	_	_	_	_	_	_	_	UART2 Receive Register									0000
U2BRG	0238		—   —   —   —   —   —   — UAR12 Receive Register  Baud Rate Generator Prescaler													0000		

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

### TABLE 4-15: SPI1 REGISTER MAP

SFR Name	SFR 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
SPI1STAT	0240	SPIEN	1	SPISIDL	1	-	_	_	_	_	SPIROV	_	-	_	_	SPITBF	SPIRBF	0000
SPI1CON1	0242	_	_	_	DISSCK	DISSDO	MODE16	SMP	CKE	SSEN	CKP	MSTEN		SPRE<2:0>	•	PPRE	<1:0>	0000
SPI1CON2	0244	FRMEN	SPIFSD	FRMPOL	-	_	_	_	_	_	_	_	_	_	_	FRMDLY	_	0000
SPI1BUF	0248		SPI1 Transmit and Receive Buffer Register												0000			

dsPIC33FJXXXMCX06A/X08A/X10A

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

### TABLE 4-16: SPI2 REGISTER MAP

SFR Name	SFR 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
SPI2STAT	0260	SPIEN	_	SPISIDL	1	1	_	_	_	-	SPIROV	_	-	_	_	SPITBF	SPIRBF	0000
SPI2CON1	0262	_	_	_	DISSCK	DISSDO	MODE16	SMP	CKE	SSEN	CKP	MSTEN		SPRE<2:0>	•	PPRE	<1:0>	0000
SPI2CON2	0264	FRMEN	SPIFSD	FRMPOL	-	-	_	_	_	_	_	_	_	_	_	FRMDLY	_	0000
SPI2BUF	0268							SPI2 Trans	smit and Re	ceive Buffer	Register							0000

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

TABLE 4-17: ADC1 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	0300								ADC1 Dat	ta Buffer 0								xxxx
AD1CON1	0320	ADON	_	ADSIDL	ADDMABM	_	AD12B	FORM	//<1:0>	;	SSRC<2:0>	•	_	SIMSAM	ASAM	SAMP	DONE	0000
AD1CON2	0322	,	VCFG<2:0	>	_	_	CSCNA	CHPS	S<1:0>	BUFS	_		SMPI	<3:0>		BUFM	ALTS	0000
AD1CON3	0324	ADRC	_	_		S	AMC<4:0>						ADCS	<7:0>				0000
AD1CHS123	0326	1	_	_	_	_	CH123N	IB<1:0>	CH123SB	_	_	_	_	_	CH1231	NA<1:0>	CH123SA	0000
AD1CHS0	0328	CH0NB	_	_		С	H0SB<4:0>	•		CH0NA	_	_		(	CH0SA<4:0	)>		0000
AD1PCFGH <sup>(1)</sup>	032A	PCFG31	PCFG30	PCFG29	PCFG28	PCFG27	PCFG26	PCFG25	PCFG24	PCFG23	PCFG22	PCFG21	PCFG20	PCFG19	PCFG18	PCFG17	PCFG16	0000
AD1PCFGL	032C	PCFG15	PCFG14	PCFG13	PCFG12	PCFG11	PCFG10	PCFG9	PCFG8	PCFG7	PCFG6	PCFG5	PCFG4	PCFG3	PCFG2	PCFG1	PCFG0	0000
AD1CSSH <sup>(1)</sup>	032E	CSS31	CSS30	CSS29	CSS28	CSS27	CSS26	CSS25	CSS24	CSS23	CSS22	CSS21					CSS16	0000
AD1CSSL	0330	CSS15	CSS14	CSS13	CSS12	CSS11	CSS10	CSS9	CSS8	CSS7	CSS6	CSS5	CSS4	CSS3	CSS2	CSS1	CSS0	0000
AD1CON4	0332	_	_		_	_	_		_	_	_	_	— — DMABL<2:0>					0000

 $\textbf{Legend:} \qquad \textbf{x} \texttt{ = unknown value on Reset, } \textbf{—} \texttt{ = unimplemented, read as `0'. Reset values are shown in hexadecimal.}$ 

Note 1: Not all ANx inputs are available on all devices. Refer to the device pin diagrams for available ANx inputs.

### TABLE 4-18: ADC2 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	AII Resets
ADC2BUF0	0340								ADC2 Data	Buffer 0								xxxx
AD2CON1	0360	ADON	_	ADSIDL	ADDMABM	_	AD12B	FOR	VI<1:0>	;	SSRC<2:0	>	_	SIMSAM	ASAM	SAMP	DONE	0000
AD2CON2	0362	,	VCFG<2:0>	>	_	_	CSCNA	CHP	S<1:0>	BUFS	_		SMPI	<3:0>		BUFM	ALTS	0000
AD2CON3	0364	ADRC	_	_		S	AMC<4:0>						ADC	S<7:0>				0000
AD2CHS123	0366	-	_	_	_	_	CH123N	IB<1:0>	CH123SB	_	_	_	_	_	CH123N	NA<1:0>	CH123SA	0000
AD2CHS0	0368	CH0NB	1	1	_		CH0S	B<3:0>		CH0NA	_		ı		CH0S	A<3:0>		0000
Reserved	036A	1	1	1	_	ı	_	ı	1	I	_		ı	_	I	-	_	0000
AD2PCFGL	036C	PCFG15	PCFG14	PCFG13	PCFG12	PCFG11	PCFG10	PCFG9	PCFG8	PCFG7	PCFG6	PCFG5	PCFG4	PCFG3	PCFG2	PCFG1	PCFG0	0000
Reserved	036E	1	1	1	_	ı	_	ı	1	I	_		ı				0000	
AD2CSSL	0370	CSS15	CSS14	CSS13	CSS12	CSS11	CSS10	CSS9	CSS8	CSS7	CSS6	CSS5	CSS4	CSS3	CSS2	CSS1	CSS0	0000
AD2CON4	0372	1				ı	_			I	_			_	— DMABL<2:0>			

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

TABLE 4-19:	DMA	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
DMA0CON	0380	CHEN	SIZE	DIR	HALF	NULLW	_	_	_	-	_	AMOD	E<1:0>	_	_	MODE	<1:0>	0000
DMA0REQ	0382	FORCE	_	_	_	_	_	_	_	_			I	RQSEL<6:0	>			0000
DMA0STA	0384								S	TA<15:0>								0000
DMA0STB	0386								S	TB<15:0>								0000
DMA0PAD	0388								P.	AD<15:0>								0000
DMA0CNT	038A	_	_	_	_	_	_					CNT	<9:0>					0000
DMA1CON	038C	CHEN	SIZE	DIR	HALF	NULLW	_	_	_	-	_	AMOD	E<1:0>	_	_	MODE	<1:0>	0000
DMA1REQ	038E	FORCE	_	_	_	_	_	_	_	-			I	RQSEL<6:0	>			0000
DMA1STA	0390								S	TA<15:0>								0000
DMA1STB	0392								S	TB<15:0>								0000
DMA1PAD	0394								P.	AD<15:0>								0000
DMA1CNT	0396	_	_	_	_	_	_					CNT	<9:0>					0000
DMA2CON	0398	CHEN	SIZE	DIR	HALF	NULLW	_	_	_	_	_	AMOD	E<1:0>	_	_	MODE	<1:0>	0000
DMA2REQ	039A	FORCE	_	_		_	_	_	_	_			l	RQSEL<6:0	>			0000
DMA2STA	039C																0000	
DMA2STB	039E								S	TB<15:0>								0000
DMA2PAD	03A0								P.	AD<15:0>								0000
DMA2CNT	03A2	_	_	_	_	_	_					CNT	T<9:0>					0000
DMA3CON	03A4	CHEN	SIZE	DIR	HALF	NULLW	_		_		_	AMOD	E<1:0>	_	_	MODE	<1:0>	0000
DMA3REQ	03A6	FORCE	_	_	_	_	_	_	_	_			I	RQSEL<6:0	>			0000
DMA3STA	03A8								S	TA<15:0>								0000
DMA3STB	03AA								S	TB<15:0>								0000
DMA3PAD	03AC								P	AD<15:0>								0000
DMA3CNT	03AE		_	_		_	_					CNT	T<9:0>					0000
DMA4CON	03B0	CHEN	SIZE	DIR	HALF	NULLW	_	_	_	_	_	AMOD	E<1:0>	_	_	MODE	<1:0>	0000
DMA4REQ	03B2	FORCE	_	_	_	_	_	_	_	_			I	RQSEL<6:0	>			0000
DMA4STA	03B4								S	TA<15:0>								0000
DMA4STB	03B6								S	TB<15:0>								0000
DMA4PAD	03B8								P.	AD<15:0>								0000
DMA4CNT	03BA		_	_		_	_					CNT	T<9:0>					0000
DMA5CON	03BC	CHEN	SIZE	DIR	HALF	NULLW	_		_	_	_	AMOD	E<1:0>	_	_	MODE	<1:0>	0000
DMA5REQ	03BE	FORCE	_	_	_	_	_	_	_	_			I	RQSEL<6:0	>			0000
DMA5STA	03C0								S	TA<15:0>								0000
DMA5STB	03C2								S	TB<15:0>								0000

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

### TABLE 4-19: DMA 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	AII Resets		
DMA5PAD	03C4								P/	AD<15:0>								0000		
DMA5CNT	03C6	_	_	_	_	_	_					CNT	<9:0>					0000		
DMA6CON	03C8	CHEN	SIZE	DIR	HALF	NULLW	_	_	_	_	_	AMOD	E<1:0>	_	_	MODE	<1:0>	0000		
DMA6REQ	03CA	FORCE	_	_	_	_	_	_	_	_			I	RQSEL<6:0	>			0000		
DMA6STA	03CC								S	TA<15:0>								0000		
DMA6STB	03CE								S	TB<15:0>								0000		
DMA6PAD	03D0															0000				
DMA6CNT	03D2	-	_	_	_	_	_	CNT<9:0>												
DMA7CON	03D4	CHEN	SIZE	DIR	HALF	NULLW	_	_	_	_	_	AMOD	E<1:0>	_	_	MODE	<1:0>	0000		
DMA7REQ	03D6	FORCE	_	_	_	_	_	_	_	_			- 1	RQSEL<6:0	>			0000		
DMA7STA	03D8								S	TA<15:0>								0000		
DMA7STB	03DA								S	TB<15:0>								0000		
DMA7PAD	03DC								P	AD<15:0>								0000		
DMA7CNT	03DE	_	_	_	_	_	_					CNT	<9:0>					0000		
DMACS0	03E0	PWCOL7	PWCOL6	PWCOL5	PWCOL4	PWCOL3	PWCOL2	PWCOL1	DL1 PWCOL0 XWCOL7 XWCOL6 XWCOL5 XWCOL4 XWCOL3 XWCOL2 XWCOL1 XWCOL0 0											
DMACS1	03E2	_	_	_	_		LSTCH	H<3:0>												
DSADR	03E4								DS/	ADR<15:0>								0000		

dsPIC33FJXXXMCX06A/X08A/X10A

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

_
$\subset$
ŭ
_
0594
œ
94
Т
$\sigma$
7
g
page
age 55

### TABLE 4-20: ECAN1 REGISTER MAP WHEN WIN (C1CTRL<0>) = 0 OR 1

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
C1CTRL1	0400	_	_	CSIDL	ABAT	_	RE	QOP<2:0>	>	OPN	//ODE<2:0	>	_	CANCAP	_	_	WIN	0480
C1CTRL2	0402	_	_	-	_	_	_	_	_	_	_	_		DI	NCNT<4:0	>		0000
C1VEC	0404	_	_	-		FI	LHIT<4:0>			_			I	CODE<6:0>	•			0000
C1FCTRL	0406	D	MABS<2:0	>	_	_	_	_	_	_	_	_		1	FSA<4:0>			0000
C1FIFO	0408	_	_			FBP<	5:0>			_	_			FNRB	<5:0>			0000
C1INTF	040A	_	_	TXBO	TXBP	RXBP	TXWAR	RXWAR	EWARN	IVRIF	WAKIF	ERRIF	_	FIFOIF	RBOVIF	RBIF	TBIF	0000
C1INTE	040C	_	_	_	_	_	_	_	_	IVRIE	WAKIE	ERRIE	_	FIFOIE	RBOVIE	RBIE	TBIE	0000
C1EC	040E				TERRON	T<7:0>							RERRCN	T<7:0>				0000
C1CFG1	0410	_	_	_	_	_	_	_	_	SJW<1	:0>			BRP<	5:0>			0000
C1CFG2	0412	_	WAKFIL	_	_	_	SE	G2PH<2:0	>	SEG2PHTS	SAM	S	EG1PH<2:	:0>	Р	RSEG<2:0	)>	0000
C1FEN1	0414	FLTEN15	FLTEN14	FLTEN13	FLTEN12	FLTEN11	FLTEN10	FLTEN9	FLTEN8	FLTEN7	FLTEN6	FLTEN5	FLTEN4	FLTEN3	FLTEN2	FLTEN1	FLTEN0	FFFF
C1FMSKSEL1	0418	F7MSk	<1:0>	F6MSI	<<1:0>	F5MSI	K<1:0>	F4MSh	<1:0>	F3MSK<	:1:0>	F2MSk	<1:0>	F1MSK	<1:0>	F0MS	K<1:0>	0000
C1FMSKSEL2	041A	F15MS	K<1:0>	F14MS	K<1:0>	F13MS	K<1:0>	F12MS	K<1:0>	F11MSK	<1:0>	F10MS	K<1:0>	F9MSK	<1:0>	F8MS	K<1:0>	0000

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

### TABLE 4-21: ECAN1 REGISTER MAP WHEN WIN (C1CTRL<0>) = 0

							( • . •	_ , , _	•									
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
	0400- 041E							See	definition	when WIN	= x							
C1RXFUL1	0420	RXFUL15	RXFUL14	RXFUL13	RXFUL12	RXFUL11	RXFUL10	RXFUL9	RXFUL8	RXFUL7	RXFUL6	RXFUL5	RXFUL4	RXFUL3	RXFUL2	RXFUL1	RXFUL0	0000
C1RXFUL2	0422	RXFUL31	RXFUL30	RXFUL29	RXFUL28	RXFUL27	RXFUL26	RXFUL25	RXFUL24	RXFUL23	RXFUL22	RXFUL21	RXFUL20	RXFUL19	RXFUL18	RXFUL17	RXFUL16	0000
C1RXOVF1	0428	RXOVF15	RXOVF14	RXOVF13	RXOVF12	RXOVF11	RXOVF10	RXOVF9	RXOVF8	RXOVF7	RXOVF6	RXOVF5	RXOVF4	RXOVF3	RXOVF2	RXOVF1	RXOVF0	0000
C1RXOVF2	042A	RXOVF31	RXOVF30	RXOVF29	RXOVF28	RXOVF27	RXOVF26	RXOVF25	RXOVF24	RXOVF23	RXOVF22	RXOVF21	RXOVF20	RXOVF19	RXOVF18	RXOVF17	RXOVF16	0000
C1TR01CON	0430	TXEN1	TXABT1	TXLARB1	TXERR1	TXREQ1	RTREN1	TX1PF	RI<1:0>	TXEN0	TXABAT0	TXLARB0	TXERR0	TXREQ0	RTREN0	TX0PF	RI<1:0>	0000
C1TR23CON	0432	TXEN3	TXABT3	TXLARB3	TXERR3	TXREQ3	RTREN3	TX3PF	RI<1:0>	TXEN2	TXABAT2	TXLARB2	TXERR2	TXREQ2	RTREN2	TX2PF	RI<1:0>	0000
C1TR45CON	0434	TXEN5	TXABT5	TXLARB5	TXERR5	TXREQ5	RTREN5	TX5PF	RI<1:0>	TXEN4	TXABAT4	TXLARB4	TXERR4	TXREQ4	RTREN4	TX4PF	RI<1:0>	0000
C1TR67CON	0436	TXEN7	TXABT7	TXLARB7	TXERR7	TXREQ7	RTREN7	+		TXEN6	TXABAT6	TXLARB6	TXERR6	TXREQ6	RTREN6	TX6PF	RI<1:0>	xxxx
C1RXD	0440		•	•		•		EC	AN1 Recei	ved Data W	ord	•		•	•			xxxx
C1TXD	0442		•	•		•		EC	AN1 Trans	mit Data Wo	ord	•		•	•			xxxx

dsPIC33FJXXXMCX06A/X08A/X10A

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

**TABLE 4-22:** ECAN1 REGISTER MAP WHEN WIN (C1CTRL<0>) = 1

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	AII Resets
	0400- 041E								See defini	tion when V	VIN = x							
C1BUFPNT1	0420		F3BF	P<3:0>			F2BI	P<3:0>			F1BP	<3:0>			F0BP	<3:0>		0000
C1BUFPNT2	0422		F7BF	P<3:0>			F6BI	P<3:0>			F5BP	<3:0>			F4BP	<3:0>		0000
C1BUFPNT3	0424		F11B	P<3:0>			F10B	P<3:0>			F9BP	<3:0>			F8BP	<3:0>		0000
C1BUFPNT4	0426		F15B	P<3:0>			F14B	P<3:0>			F13BF	P<3:0>			F12BF	P<3:0>		0000
C1RXM0SID	0430				SID<	:10:3>					SID<2:0>		_	MIDE	_	EID<	17:16>	xxxx
C1RXM0EID	0432				EID<	:15:8>							EID<	7:0>				xxxx
C1RXM1SID	0434				SID<	:10:3>					SID<2:0>		_	MIDE	_	EID<	17:16>	xxxx
C1RXM1EID	0436				EID<	:15:8>							EID<	7:0>				xxxx
C1RXM2SID	0438				SID<	:10:3>					SID<2:0>		_	MIDE	_	EID<	17:16>	xxxx
C1RXM2EID	043A				EID<	:15:8>							EID<	7:0>				xxxx
C1RXF0SID	0440				SID<	:10:3>					SID<2:0>		_	EXIDE	_	EID<	17:16>	xxxx
C1RXF0EID	0442				EID<	:15:8>							EID<	7:0>				xxxx
C1RXF1SID	0444				SID<	:10:3>					SID<2:0>		_	EXIDE	_	EID<	17:16>	xxxx
C1RXF1EID	0446				EID<	:15:8>							EID<	7:0>				xxxx
C1RXF2SID	0448				SID<	:10:3>					SID<2:0>		_	EXIDE	_	EID<	17:16>	xxxx
C1RXF2EID	044A				EID<	:15:8>							EID<	7:0>				xxxx
C1RXF3SID	044C				SID<	:10:3>					SID<2:0>		_	EXIDE	_	EID<	17:16>	xxxx
C1RXF3EID	044E				EID<	:15:8>							EID<	7:0>				xxxx
C1RXF4SID	0450				SID<	:10:3>					SID<2:0>		_	EXIDE	_	EID<	17:16>	xxxx
C1RXF4EID	0452				EID<	:15:8>							EID<	7:0>				xxxx
C1RXF5SID	0454				SID<	:10:3>					SID<2:0>		_	EXIDE	_	EID<	17:16>	xxxx
C1RXF5EID	0456				EID<	:15:8>							EID<	7:0>				xxxx
C1RXF6SID	0458				SID<	:10:3>					SID<2:0>		_	EXIDE	_	EID<	17:16>	xxxx
C1RXF6EID	045A				EID<	:15:8>							EID<	7:0>				xxxx
C1RXF7SID	045C				SID<	:10:3>					SID<2:0>		_	EXIDE	_	EID<	17:16>	xxxx
C1RXF7EID	045E				EID<	:15:8>							EID<	7:0>				xxxx
C1RXF8SID	0460				SID<	:10:3>					SID<2:0>		_	EXIDE	_	EID<	17:16>	xxxx
C1RXF8EID	0462				EID<	:15:8>							EID<	7:0>				xxxx
C1RXF9SID	0464				SID<	:10:3>					SID<2:0>		_	EXIDE	_	EID<	17:16>	xxxx
C1RXF9EID	0466				EID<	:15:8>							EID<	7:0>				xxxx
C1RXF10SID	0468				SID<	:10:3>					SID<2:0>		_	EXIDE	_	EID<	17:16>	xxxx
C1RXF10EID	046A				EID<	:15:8>							EID<	7:0>				xxxx

dsPIC33FJXXXMCX06A/X08A/X10A

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

TABLE 4-22: ECAN1 REGISTER MAP WHEN WIN (C1CTRL<0>) = 1 (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
C1RXF11SID	046C				SID<	10:3>					SID<2:0>		_	EXIDE	_	EID<1	17:16>	xxxx
C1RXF11EID	046E				EID<	15:8>							EID<	7:0>				xxxx
C1RXF12SID	0470				SID<	10:3>					SID<2:0>		_	EXIDE	_	EID<1	7:16>	xxxx
C1RXF12EID	0472	EID<15:8> EID<7:0>												xxxx				
C1RXF13SID	0474				SID<	10:3>					SID<2:0>		_	EXIDE	_	EID<1	17:16>	xxxx
C1RXF13EID	0476				EID<	15:8>							EID<	7:0>				xxxx
C1RXF14SID	0478				SID<	10:3>					SID<2:0>		_	EXIDE	_	EID<1	17:16>	xxxx
C1RXF14EID	047A				EID<	15:8>							EID<	7:0>		•		xxxx
C1RXF15SID	047C				SID<	10:3>					SID<2:0>		_	EXIDE	_	EID<1	17:16>	xxxx
C1RXF15EID	047E				EID<	15:8>							EID<	7:0>	•	-		xxxx

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

TABLE 4-23: ECAN2 REGISTER MAP WHEN WIN (C1CTRL<0>) = 0 OR 1 FOR dsPIC33FJXXXMC708A/710A DEVICES

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
C2CTRL1	0500	_	_	CSIDL	ABAT	_	RE	EQOP<2:0	>	OPM	10DE<2:0	>	_	CANCAP	_	_	WIN	0480
C2CTRL2	0502	-	-	_	_	_	_	_	_	_	_	_		D	NCNT<4:0	<b> &gt;</b>		0000
C2VEC	0504	-	-	_		FI	LHIT<4:0>			_				ICODE<6:0	)>			0000
C2FCTRL	0506		MABS<2:0>	>	_	_	_	_	_	_	_	_			FSA<4:0>			0000
C2FIFO	0508	_	_			FBP<5	:0>			_	_		FNRB<5:0>					0000
C2INTF	050A	1	-	TXBO	TXBP	RXBP	TXWAR	RXWAR	EWARN	IVRIF	WAKIF	ERRIF	_	FIFOIF	RBOVIF	RBIF	TBIF	0000
C2INTE	050C	1	-	_	-	_	_	_	_	IVRIE	WAKIE	ERRIE	_	FIFOIE	RBOVIE	RBIE	TBIE	0000
C2EC	050E				TERRON	Γ<7:0>							RERRCN	NT<7:0>				0000
C2CFG1	0510	_	_	_	_	_	_	_	_	SJW<1	:0>			BRP	<5:0>			0000
C2CFG2	0512	_	WAKFIL	_	_	_	SE	G2PH<2:0	)>	SEG2PHTS	SAM	SI	EG1PH<2:	0>	Р	RSEG<2:0	)>	0000
C2FEN1	0514	FLTEN15	FLTEN14	FLTEN13	TEN13 FLTEN12 FLTEN11 FLTEN10 FLTEN9 FLTEN8 FLTEN7 FLTEN6 FLTEN5 FLTEN4 FLTEN3 FLTEN2 FLTEN1 FLTEN						FLTEN0	FFFF						
C2FMSKSEL1	0518	F7MSł	<<1:0>	F6MSI	<<1:0>	F5MSk	<1:0>	F4MSI	<<1:0>	F3MSK<	:1:0>	F2MSh	<<1:0>	F1MSk	<<1:0>	F0MS	K<1:0>	0000
C2FMSKSEL2	051A	F15MS	K<1:0>	F14MS	K<1:0>	F13MS	K<1:0>	F12MS	K<1:0>	F11MSK	<1:0>	F10MS	K<1:0>	F9MSk	<<1:0>	F8MS	K<1:0>	0000

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

### **TABLE 4-24:** ECAN2 REGISTER MAP WHEN WIN (C1CTRL<0>) = 0 FOR dsPIC33FJXXXMC708A/710A DEVICES

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	AII Resets
	0500- 051E							See	e definition	when WIN	= x							
C2RXFUL1	0520	RXFUL15	RXFUL14	RXFUL13	RXFUL12	RXFUL11	RXFUL10	RXFUL9	RXFUL8	RXFUL7	RXFUL6	RXFUL5	RXFUL4	RXFUL3	RXFUL2	RXFUL1	RXFUL0	0000
C2RXFUL2	0522	RXFUL31	RXFUL30	RXFUL29	RXFUL28	RXFUL27	RXFUL26	RXFUL25	RXFUL24	RXFUL23	RXFUL22	RXFUL21	RXFUL20	RXFUL19	RXFUL18	RXFUL17	RXFUL16	0000
C2RXOVF1	0528	RXOVF15	RXOVF14	RXOVF13	RXOVF12	RXOVF11	RXOVF10	RXOVF09	RXOVF08	RXOVF7	RXOVF6	RXOVF5	RXOVF4	RXOVF3	RXOVF2	RXOVF1	RXOVF0	0000
C2RXOVF2	052A	RXOVF31	RXOVF30	RXOVF29	RXOVF28	RXOVF27	RXOVF26	RXOVF25	RXOVF24	RXOVF23	RXOVF22	RXOVF21	RXOVF20	RXOVF19	RXOVF18	RXOVF17	RXOVF16	0000
C2TR01CON	0530	TXEN1	TX ABAT1	TX LARB1	TX ERR1	TX REQ1	RTREN1	TX1PRI<1:0>		TXEN0	TX ABAT0	TX LARB0	TX ERR0	TX REQ0	RTREN0	TX0PF	RI<1:0>	0000
C2TR23CON	0532	TXEN3	TX ABAT3	TX LARB3	TX ERR3	TX REQ3	RTREN3	TX3PF	RI<1:0>	TXEN2	TX ABAT2	TX LARB2	TX ERR2	TX REQ2	RTREN2	TX2PF	RI<1:0>	0000
C2TR45CON	0534	TXEN5	TX ABAT5	TX LARB5	TX ERR5	TX REQ5	RTREN5	TX5PF	RI<1:0>	TXEN4	TX ABAT4	TX LARB4	TX ERR4	TX REQ4	RTREN4	TX4PF	RI<1:0>	0000
C2TR67CON	0536	TXEN7	TX ABAT7	TX LARB7	TX ERR7	TX REQ7	RTREN7	TX7PRI<1:0>		TXEN6	TX ABAT6	TX LARB6	TX ERR6	TX REQ6	RTREN6	TX6PF	RI<1:0>	xxxx
C2RXD	0540							EC	AN2 Recie	ved Data W	ord							xxxx
C2TXD	0542							EC	CAN2 Trans	mit Data Wo	ord							xxxx

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

**TABLE 4-25:** ECAN2 REGISTER MAP WHEN WIN (C1CTRL<0>) = 1 FOR dsPIC33FJXXXMC708A/710A DEVICES

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
	0500- 051E							See	edefinition	when WIN	= x							
C2BUFPNT1	0520		F3BF	P<3:0>			F2BP	2<3:0>			F1BF	2<3:0>			F0BF	P<3:0>		0000
C2BUFPNT2	0522		F7BF	P<3:0>			F6BP	2<3:0>			F5BF	2<3:0>			F4BF	P<3:0>		0000
C2BUFPNT3	0524		F11BI	P<3:0>			F10BF	P<3:0>			F9BF	>3:0>			F8BF	P<3:0>		0000
C2BUFPNT4	0526		F15BI	P<3:0>			F14BF	P<3:0>			F13BI	P<3:0>			F12BI	P<3:0>		0000
C2RXM0SID	0530				SID<	10:3>					SID<2:0>		_	MIDE	_	EID<	17:16>	xxxx
C2RXM0EID	0532				EID<	15:8>							EID<	<7:0>				xxxx
C2RXM1SID	0534				SID<	10:3>					SID<2:0>		_	MIDE	_	EID<	17:16>	xxxx
C2RXM1EID	0536				EID<	15:8>							EID<	<7:0>				xxxx
C2RXM2SID	0538				SID<	10:3>					SID<2:0>		_	MIDE	_	EID<	17:16>	xxxx
C2RXM2EID	053A				EID<	15:8>							EID<	<7:0>				xxxx
C2RXF0SID	0540				SID<	10:3>					SID<2:0>		_	EXIDE	_	EID<	17:16>	xxxx
C2RXF0EID	0542				EID<	15:8>							EID<	<7:0>				xxxx
C2RXF1SID	0544				SID<	10:3>					SID<2:0>		_	EXIDE	_	EID<	17:16>	xxxx
C2RXF1EID	0546				EID<	15:8>							EID<	<7:0>				xxxx
C2RXF2SID	0548				SID<	10:3>					SID<2:0>		_	EXIDE	_	EID<	17:16>	xxxx
C2RXF2EID	054A				EID<	15:8>							EID	<7:0>				xxxx
C2RXF3SID	054C				SID<	10:3>					SID<2:0>		_	EXIDE	_	EID<	17:16>	xxxx
C2RXF3EID	054E				EID<	15:8>							EID	<7:0>				xxxx
C2RXF4SID	0550				SID<	10:3>					SID<2:0>		_	EXIDE	_	EID<	17:16>	xxxx
C2RXF4EID	0552				EID<	15:8>							EID	<7:0>				xxxx
C2RXF5SID	0554				SID<	10:3>					SID<2:0>		_	EXIDE	_	EID<	17:16>	xxxx
C2RXF5EID	0556				EID<	15:8>							EID	<7:0>				xxxx
C2RXF6SID	0558				SID<	10:3>					SID<2:0>		_	EXIDE	_	EID<	17:16>	xxxx
C2RXF6EID	055A				EID<	15:8>							EID	<7:0>				xxxx
C2RXF7SID	055C				SID<	10:3>					SID<2:0>		_	EXIDE	_	EID<	17:16>	xxxx
C2RXF7EID	055E				EID<	15:8>							EID	<7:0>				xxxx
C2RXF8SID	0560				SID<	10:3>					SID<2:0>		_	EXIDE	_	EID<	17:16>	xxxx
C2RXF8EID	0562				EID<	15:8>							EID	<7:0>				xxxx
C2RXF9SID	0564				SID<	10:3>					SID<2:0>		_	EXIDE	_	EID<	17:16>	xxxx
C2RXF9EID	0566				EID<	15:8>							EID<	<7:0>				xxxx
C2RXF10SID	0568				SID<	10:3>		·			SID<2:0>		_	EXIDE	_	EID<	17:16>	xxxx
C2RXF10EID	056A				EID<	15:8>							EID	<7:0>				xxxx

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

### TABLE 4-25: ECAN2 REGISTER MAP WHEN WIN (C1CTRL<0>) = 1 FOR dsPIC33FJXXXMC708A/710A DEVICES (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
C2RXF11SID	056C				SID<	10:3>					SID<2:0>		_	EXIDE	_	EID<1	7:16>	xxxx
C2RXF11EID	056E				EID<	15:8>							EID<	<7:0>				xxxx
C2RXF12SID	0570				SID<	10:3>					SID<2:0>		_	EXIDE	_	EID<1	7:16>	xxxx
C2RXF12EID	0572				EID<	15:8>							EID<	<7:0>				xxxx
C2RXF13SID	0574				SID<	10:3>					SID<2:0>		_	EXIDE	_	EID<1	7:16>	xxxx
C2RXF13EID	0576				EID<	15:8>							EID<	<7:0>				xxxx
C2RXF14SID	0578				SID<	10:3>					SID<2:0>		_	EXIDE	_	EID<1	7:16>	xxxx
C2RXF14EID	057A				EID<	15:8>							EID<	<7:0>				xxxx
C2RXF15SID	057C				SID<	10:3>					SID<2:0>		_	EXIDE	_	EID<1	7:16>	xxxx
C2RXF15EID	057E				EID<	15:8>							EID<	<7:0>				xxxx

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

### TABLE 4-26: PORTA REGISTER MAP<sup>(1)</sup>

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	AII Resets
TRISA	02C0	TRISA15	TRISA14	-	_	-	TRISA10	TRISA9	_	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	C6FF
PORTA	02C2	RA15	RA14	_	_	_	RA10	RA9	_	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0	xxxx
LATA	02C4	LATA15	LATA14	_	_	_	LATA10	LATA9	_	LATA7	LATA6	LATA5	LATA4	LATA3	LATA2	LATA1	LATA0	xxxx
ODCA <sup>(2)</sup>	06C0	ODCA15	ODCA14	_	_	_	_	_	_	_	_	ODCA5	ODCA4	ODCA3	ODCA2	ODCA1	ODCA0	0000

dsPIC33FJXXXMCX06A/X08A/X10A

Legend: x = unknown value on Reset, — = unimplemented, read as '0'. Reset values are shown in hexadecimal for high pin count devices.

Note 1: The actual set of I/O port pins varies from one device to another. Please refer to the corresponding pinout diagrams.

### TABLE 4-27: PORTB REGISTER MAP<sup>(1)</sup>

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	02C6	TRISB15	TRISB14	TRISB13	TRISB12	TRISB11	TRISB10	TRISB9	TRISB8	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	FFFF
PORTB	02C8	RB15	RB14	RB13	RB12	RB11	RB10	RB9	RB8	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx
LATB	02CA	LATB15	LATB14	LATB13	LATB12	LATB11	LATB10	LATB9	LATB8	LATB7	LATB6	LATB5	LATB4	LATB3	LATB2	LATB1	LATB0	xxxx

Legend: x = unknown value on Reset, — = unimplemented, read as '0'. Reset values are shown in hexadecimal for high pin count devices.

Note 1: The actual set of I/O port pins varies from one device to another. Please refer to the corresponding pinout diagrams.

## TABLE 4-28: PORTC REGISTER MAP<sup>(1)</sup>

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
TRISC	02CC	TRISC15	TRISC14	TRISC13	TRISC12	_	_	1	_	_	1	_	TRISC4	TRISC3	TRISC2	TRISC1	1	F01E
PORTC	02CE	RC15	RC14	RC13	RC12	_	_	_	_	_	-	_	RC4	RC3	RC2	RC1	_	xxxx
LATC	02D0	LATC15	LATC14	LATC13	LATC12	_	ı	ı	ı	_	ı	_	LATC4	LATC3	LATC2	LATC1	ı	xxxx

Legend: x = unknown value on Reset, — = unimplemented, read as '0'. Reset values are shown in hexadecimal for high pin count devices.

Note 1: The actual set of I/O port pins varies from one device to another. Please refer to the corresponding pinout diagrams.

### TABLE 4-29: PORTD REGISTER MAP<sup>(1)</sup>

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
TRISD	02D2	TRISD15	TRISD14	TRISD13	TRISD12	TRISD11	TRISD10	TRISD9	TRISD8	TRISD7	TRISD6	TRISD5	TRISD4	TRISD3	TRISD2	TRISD1	TRISD0	FFFF
PORTD	02D4	RD15	RD14	RD13	RD12	RD11	RD10	RD9	RD8	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	xxxx
LATD	02D6	LATD15	LATD14	LATD13	LATD12	LATD11	LATD10	LATD9	LATD8	LATD7	LATD6	LATD5	LATD4	LATD3	LATD2	LATD1	LATD0	xxxx
ODCD	06D2	ODCD15	ODCD14	ODCD13	ODCD12	ODCD11	ODCD10	ODCD9	ODCD8	ODCD7	ODCD6	ODCD5	ODCD4	ODCD3	ODCD2	ODCD1	ODCD0	0000

.egend: x = unknown value on Reset, — = unimplemented, read as '0'. Reset values are shown in hexadecimal for high pin count devices.

Note 1: The actual set of I/O port pins varies from one device to another. Please refer to the corresponding pinout diagrams.

### TABLE 4-30: PORTE REGISTER MAP<sup>(1)</sup>

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	AII Resets
TRISE	02D8	-	_	_	_	_	_	TRISE9	TRISE8	TRISE7	TRISE6	TRISE5	TRISE4	TRISE3	TRISE2	TRISE1	TRISE0	01FF
PORTE	02DA	-	_	_	_	_	_	RE9	RE8	RE7	RE6	RE5	RE4	RE3	RE2	RE1	RE0	xxxx
LATE	02DC	_		-	_	_	_	LATE9	LATE8	LATE7	LATE6	LATE5	LATE4	LATE3	LATE2	LATE1	LATE0	xxxx

dsPIC33FJXXXMCX06A/X08A/X10A

Legend: x = unknown value on Reset, — = unimplemented, read as '0'. Reset values are shown in hexadecimal for high pin count devices.

Note 1: The actual set of I/O port pins varies from one device to another. Please refer to the corresponding pinout diagrams.

### TABLE 4-31: PORTF REGISTER MAP<sup>(1)</sup>

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
TRISF	02DE	_	_	TRISF13	TRISF12	_	_	_	TRISF8	TRISF7	TRISF6	TRISF5	TRISF4	TRISF3	TRISF2	TRISF1	TRISF0	31FF
PORTF	02E0	_	_	RF13	RF12	_	_		RF8	RF7	RF6	RF5	RF4	RF3	RF2	RF1	RF0	xxxx
LATF	02E2	_	_	LATF13	LATF12	_	-	_	LATF8	LATF7	LATF6	LATF5	LATF4	LATF3	LATF2	LATF1	LATF0	xxxx
ODCF	06DE	_	_	ODCF13	ODCF12	_	1	_	ODCF8	ODCF7	ODCF6	ODCF5	ODCF4	ODCF3	ODCF2	ODCF1	ODCF0	0000

Legend: x = unknown value on Reset, — = unimplemented, read as '0'. Reset values are shown in hexadecimal for high pin count devices.

Note 1: The actual set of I/O port pins varies from one device to another. Please refer to the corresponding pinout diagrams.

### TABLE 4-32: PORTG REGISTER MAP<sup>(1)</sup>

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
TRISG	02E4	TRISG15	TRISG14	TRISG13	TRISG12	1		TRISG9	TRISG8	TRISG7	TRISG6	ı	_	TRISG3	TRISG2	TRISG1	TRISG0	F3CF
PORTG	02E6	RG15	RG14	RG13	RG12	-	-	RG9	RG8	RG7	RG6	ı	_	RG3	RG2	RG1	RG0	xxxx
LATG	02E8	LATG15	LATG14	LATG13	LATG12	1	_	LATG9	LATG8	LATG7	LATG6	_	_	LATG3	LATG2	LATG1	LATG0	xxxx
ODCG	06E4	ODCG15	ODCG14	ODCG13	ODCG12	_	_	ODCG9	ODCG8	ODCG7	ODCG6	_	_	ODCG3	ODCG2	ODCG1	ODCG0	0000

Legend: x = unknown value on Reset, — = unimplemented, read as '0'. Reset values are shown in hexadecimal for high pin count devices.

Note 1: The actual set of I/O port pins varies from one device to another. Please refer to the corresponding pinout diagrams.

### TABLE 4-33: SYSTEM CONTROL 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	0740	TRAPR	IOPUWR	_	_	_	_	_	VREGS	EXTR	SWR	SWDTEN	WDTO	SLEEP	IDLE	BOR	POR	xxxx(1)
OSCCON	0742	_	(	COSC<2:0>	>	_	١	NOSC<2:0	>	CLKLOCK	_	LOCK	_	CF	_	LPOSCEN	OSWEN	0300(2)
CLKDIV	0744	ROI	[	DOZE<2:0>	>	DOZEN	F	RCDIV<2:0	)>	PLLPOS	ST<1:0>	_		F	LLPRE<4:	:0>		3040
PLLFBD	0746	_	_	_	_	_	ı	_				F	PLLDIV<8:0	)>				0030
OSCTUN	0748	_	_	_	_	_	_	_	_	_	_			TUN	l<5:0>			0000

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

Note 1: RCON register Reset values dependent on type of Reset.

2: OSCCON register Reset values dependent on the FOSC Configuration bits and type of Reset.

### TABLE 4-34: 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	_	_		NVMO	P<3:0>		0000(1)
NVMKEY	0766	_	_	_	_	_	_	_	_				NVMKE	Y<7:0>				0000

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

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

### TABLE 4-35: 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	0770	T5MD	T4MD	T3MD	T2MD	T1MD	QEI1MD	PWMMD	_	I2C1MD	U2MD	U1MD	SPI2MD	SPI1MD	C2MD	C1MD	AD1MD	0000
PMD2	0772	IC8MD	IC7MD	IC6MD	IC5MD	IC4MD	IC3MD	IC2MD	IC1MD	OC8MD	OC7MD	OC6MD	OC5MD	OC4MD	OC3MD	OC2MD	OC1MD	0000
PMD3	0774	T9MD	T8MD	T7MD	T6MD	_	_	-	_	_	_	_	1	_	-	I2C2MD	AD2MD	0000

Legend: x = unknown value on Reset, — = unimplemented, read as '0'. Reset values are shown in hexadecimal for high pin count devices.

### 4.2.7 SOFTWARE STACK

In addition to its use as a working register, the W15 register in the dsPIC33FJXXXMCX06A/X08A/X10A devices is also used as a software Stack Pointer. The Stack Pointer always points to the first available free word and grows from lower to higher addresses. It pre-decrements for stack pops and post-increments for stack pushes, as shown in Figure 4-6. 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.

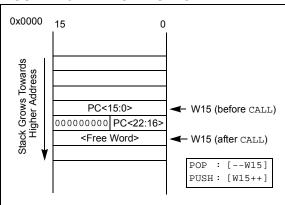
Note: A PC push during exception processing concatenates the SRL register to the MSb of the PC prior to the push.

The Stack Pointer Limit register (SPLIM) 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' because 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 0x2000 in RAM, initialize the SPLIM with the value 0x1FFE.

Similarly, a Stack Pointer underflow (stack error) trap is generated when the Stack Pointer address is found to be less than 0x0800. This prevents the stack from interfering with the Special Function Register (SFR) space.

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

FIGURE 4-6: CALL STACK FRAME



### 4.2.8 DATA RAM PROTECTION FEATURE

The dsPIC33FJXXXMCX06A/X08A/X10A devices support data RAM protection features which enable segments of RAM to be protected when used in conjunction with Boot and Secure Code Segment Security. BSRAM (Secure RAM segment for BS) is accessible only from the Boot Segment Flash code when enabled. SSRAM (Secure RAM segment for RAM) is accessible only from the Secure Segment Flash code when enabled. See Table 4-1 for an overview of the BSRAM and SSRAM SFRs.

### 4.3 Instruction Addressing Modes

The addressing modes in Table 4-36 form the basis of the addressing modes optimized to support the specific features of individual instructions. The addressing modes provided in the MAC class of instructions are somewhat different from those in the other instruction types.

### 4.3.1 FILE REGISTER INSTRUCTIONS

Most file register instructions use a 13-bit address field (f) to directly address data present in the first 8192 bytes of data memory (Near Data Space). Most file register instructions employ a working register, W0, which is denoted as WREG in these instructions. The destination is typically either the same file register or WREG (with the exception of the MUL instruction), which writes the result to a register or register pair. The MOV instruction allows additional flexibility and can access the entire data space.

### 4.3.2 MCU INSTRUCTIONS

The 3-operand MCU instructions are of the following form:

Operand 3 = Operand 1 < function > Operand 2

where Operand 1 is always a working register (i.e., the addressing mode can only be Register Direct) which is referred to as Wb. Operand 2 can be a W register fetched from data memory or a 5-bit literal. The result location can be either a W register or a data memory location. The following addressing modes are supported by MCU instructions:

- · Register Direct
- · Register Indirect
- · Register Indirect Post-Modified
- · Register Indirect Pre-Modified
- · 5-Bit or 10-Bit Literal

Note: Not all instructions support all the addressing modes given above. Individual instructions may support different subsets of these addressing modes.

TABLE 4-36: FUNDAMENTAL ADDRESSING MODES SUPPORTED

Addressing Mode	Description
File Register Direct	The address of the file register is specified explicitly.
Register Direct	The contents of a register are accessed directly.
Register Indirect	The contents of Wn forms the EA.
Register Indirect Post-Modified	The contents of Wn forms the EA. Wn is post-modified (incremented or decremented) by a constant value.
Register Indirect Pre-Modified	Wn is pre-modified (incremented or decremented) by a signed constant value to form the EA.
Register Indirect with Register Offset	The sum of Wn and Wb forms the EA.
Register Indirect with Literal Offset	The sum of Wn and a literal forms the EA.

# 4.3.3 MOVE AND ACCUMULATOR INSTRUCTIONS

Move instructions and the DSP accumulator class of instructions provide a greater degree of addressing flexibility than other instructions. In addition to the Addressing modes supported by most MCU instructions, move and accumulator instructions also support Register Indirect with Register Offset Addressing mode, also referred to as Register Indexed mode.

Note: For the MOV instructions, the addressing mode specified in the instruction can differ for the source and destination EA. However, the 4-bit Wb (register offset) field is shared between both source and destination (but typically only used by

In summary, the following addressing modes are supported by move and accumulator instructions:

- · Register Direct
- · Register Indirect
- Register Indirect Post-modified
- · Register Indirect Pre-modified
- Register Indirect with Register Offset (Indexed)
- · Register Indirect with Literal Offset
- 8-Bit Literal
- 16-Bit Literal

Note:	Not	all	instructions	support	all	the
	addr	essir	ng modes give	n above. I	ndivi	dual
	instr	uctio	ns may suppo	rt differen	t sub	sets
	of th	ese a	addressing mo	odes.		

### 4.3.4 MAC INSTRUCTIONS

The dual source operand DSP instructions (CLR, ED, EDAC, MAC, MPY, MPY. N, MOVSAC and MSC), also referred to as MAC instructions, utilize a simplified set of addressing modes to allow the user to effectively manipulate the Data Pointers through register indirect tables.

The 2-source operand prefetch registers must be members of the set {W8, W9, W10, W11}. For data reads, W8 and W9 are always directed to the X RAGU, and W10 and W11 will always be directed to the Y AGU. The Effective Addresses generated (before and after modification) must, therefore, be valid addresses within X data space for W8 and W9, and Y data space for W10 and W11.

Note: Register Indirect with Register Offset Addressing mode is only available for W9 (in X space) and W11 (in Y space).

In summary, the following addressing modes are supported by the MAC class of instructions:

- · Register Indirect
- · Register Indirect Post-Modified by 2
- · Register Indirect Post-Modified by 4
- · Register Indirect Post-Modified by 6
- Register Indirect with Register Offset (Indexed)

### 4.3.5 OTHER INSTRUCTIONS

Besides the various addressing modes outlined above, some instructions use literal constants of various sizes. For example, BRA (branch) instructions use 16-bit signed literals to specify the branch destination directly, whereas the DISI instruction uses a 14-bit unsigned literal field. In some instructions, such as ADD Acc, the source of an operand or result is implied by the opcode itself. Certain operations, such as NOP, do not have any operands.

### 4.4 Modulo Addressing

Modulo Addressing mode is a method of providing an automated means to support circular data buffers using hardware. The objective is to remove the need for software to perform data address boundary checks when executing tightly looped code, as is typical in many DSP algorithms.

Modulo Addressing can operate in either data or program space (since the Data Pointer mechanism is essentially the same for both). One circular buffer can be supported in each of the X (which also provides the pointers into program space) and Y data spaces. Modulo Addressing can operate on any W register pointer. However, it is not advisable to use W14 or W15 for Modulo Addressing, since these two registers are used as the Stack Frame Pointer and Stack Pointer, respectively.

In general, any particular circular buffer can only be configured to operate in one direction, as there are certain restrictions on the buffer start address (for incrementing buffers) or end address (for decrementing buffers), based upon the direction of the buffer.

The only exception to the usage restrictions is for buffers which have a power-of-2 length. As these buffers satisfy the start and end address criteria, they may operate in a bidirectional mode (i.e., address boundary checks will be performed on both the lower and upper address boundaries).

### 4.4.1 START AND END ADDRESS

The Modulo Addressing scheme requires that a starting and ending address be specified and loaded into the 16-bit Modulo Buffer Address registers: XMODSRT, XMODEND, YMODSRT and YMODEND (see Table 4-1).

Note: Y space Modulo Addressing EA calculations assume word-sized data (LSb of

every EA is always clear).

The length of a circular buffer is not directly specified. It is determined by the difference between the corresponding start and end addresses. The maximum possible length of the circular buffer is 32K words (64 Kbytes).

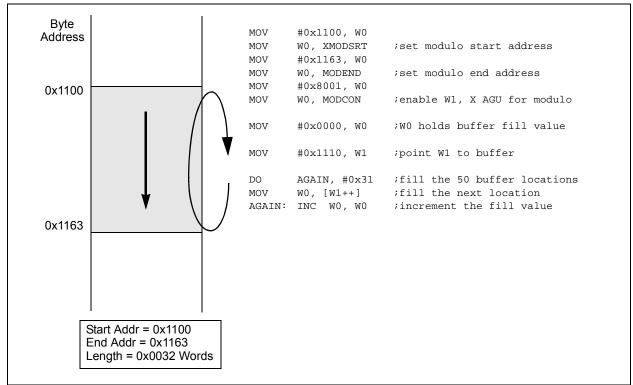
# 4.4.2 W ADDRESS REGISTER SELECTION

The Modulo and Bit-Reversed Addressing Control register, MODCON<15:0>, contains enable flags as well as a W register field to specify the W Address registers. The XWM and YWM fields select which registers will operate with Modulo Addressing. If XWM = 15, X RAGU and X WAGU Modulo Addressing are disabled. Similarly, if YWM = 15, Y AGU Modulo Addressing is disabled.

The X Address Space Pointer W register (XWM) to which Modulo Addressing is to be applied is stored in MODCON<3:0> (see Table 4-1). Modulo Addressing is enabled for X data space when XWM is set to any value other than 15 and the XMODEN bit is set at MODCON<15>.

The Y Address Space Pointer W register (YWM) to which Modulo Addressing is to be applied is stored in MODCON<7:4>. Modulo Addressing is enabled for Y data space when YWM is set to any value other than 15 and the YMODEN bit is set at MODCON<14>.

### FIGURE 4-7: MODULO ADDRESSING OPERATION EXAMPLE



# 4.4.3 MODULO ADDRESSING APPLICABILITY

Modulo Addressing can be applied to the Effective Address (EA) calculation associated with any W register. It is important to realize that the address boundaries check for addresses less than or greater than the upper (for incrementing buffers) and lower (for decrementing buffers) boundary addresses (not just equal to). Address changes may, therefore, jump beyond boundaries and still be adjusted correctly.

Note: The modulo corrected Effective Address is written back to the register only when Pre-Modify or Post-Modify Addressing mode is used to compute the Effective Address. When an address offset (e.g., [W7+W2]) is used, Modulo Address correction is performed but the contents of the register remain unchanged.

### 4.5 Bit-Reversed Addressing

Bit-Reversed Addressing mode is intended to simplify data reordering for radix-2 FFT algorithms. It is supported by the X AGU for data writes only.

The modifier, which may be a constant value or register contents, is regarded as having its bit order reversed. The address source and destination are kept in normal order; thus, the only operand requiring reversal is the modifier.

# 4.5.1 BIT-REVERSED ADDRESSING IMPLEMENTATION

Bit-Reversed Addressing mode is enabled when the following conditions exist:

- The BWM bits (W register selection) in the MODCON register are any value other than 15 (the stack cannot be accessed using Bit-Reversed Addressing).
- 2. The BREN bit is set in the XBREV register.
- 3. The addressing mode used is Register Indirect with Pre-Increment or Post-Increment.

If the length of a bit-reversed buffer is  $M = 2^N$  bytes, the last 'N' bits of the data buffer start address must be zeros.

XB<14:0> is the Bit-Reversed Address modifier, or 'pivot point,' which is typically a constant. In the case of an FFT computation, its value is equal to half of the FFT data buffer size.

Note: All bit-reversed EA calculations assume word-sized data (LSb of every EA is always clear). The XB value is scaled accordingly to generate compatible (byte) addresses.

When enabled, Bit-Reversed Addressing is only executed for Register Indirect with Pre-Increment or Post-Increment Addressing and word-sized data writes. It will not function for any other addressing mode or for byte-sized data; normal addresses are generated instead. When Bit-Reversed Addressing is active, the W Address Pointer is always added to the address modifier (XB) and the offset associated with the Register Indirect Addressing mode is ignored. In addition, as word-sized data is a requirement, the LSb of the EA is ignored (and always clear).

Note: Modulo Addressing and Bit-Reversed Addressing should not be enabled together. In the event that the user attempts to do so, Bit-Reversed Addressing will assume priority for the X WAGU, and X WAGU Modulo Addressing will be disabled. However, Modulo Addressing will continue to function in the X RAGU.

If Bit-Reversed Addressing has already been enabled by setting the BREN bit (XBREV<15>), then a write to the XBREV register should not be immediately followed by an indirect read operation using the W register that has been designated as the Bit-Reversed Pointer.

FIGURE 4-8: BIT-REVERSED ADDRESS EXAMPLE

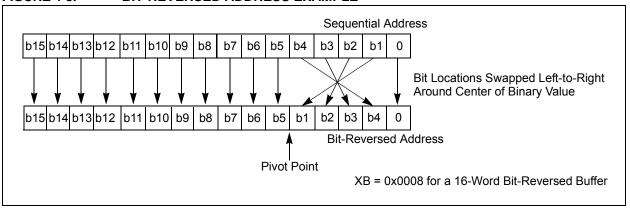


TABLE 4-37: BIT-REVERSED ADDRESS SEQUENCE (16-ENTRY)

Normal Address				Bit-Reversed Address					
А3	A2	<b>A</b> 1	A0	Decimal	А3	A2	<b>A</b> 1	A0	Decimal
0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	0	0	0	8
0	0	1	0	2	0	1	0	0	4
0	0	1	1	3	1	1	0	0	12
0	1	0	0	4	0	0	1	0	2
0	1	0	1	5	1	0	1	0	10
0	1	1	0	6	0	1	1	0	6
0	1	1	1	7	1	1	1	0	14
1	0	0	0	8	0	0	0	1	1
1	0	0	1	9	1	0	0	1	9
1	0	1	0	10	0	1	0	1	5
1	0	1	1	11	1	1	0	1	13
1	1	0	0	12	0	0	1	1	3
1	1	0	1	13	1	0	1	1	11
1	1	1	0	14	0	1	1	1	7
1	1	1	1	15	1	1	1	1	15

# 4.6 Interfacing Program and Data Memory Spaces

The dsPIC33FJXXXMCX06A/X08A/X10A architecture uses a 24-bit wide program space and a 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 dsPIC33FJXXXMCX06A/X08A/X10A 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 capability 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.

### 4.6.1 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 Page 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 Most Significant bit of TBLPAG is used to determine if the operation occurs in the user memory (TBLPAG<7> = 0) or the configuration memory (TBLPAG<7> = 1).

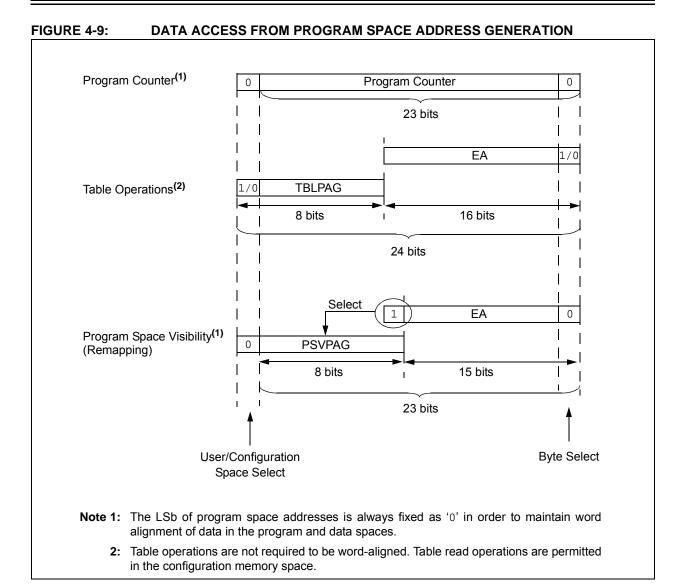
For remapping operations, the 8-bit Program Space Visibility register (PSVPAG) is used to define a 16K word page in the program space. When the Most Significant bit of the EA is '1', PSVPAG is concatenated with the lower 15 bits of the EA to form a 23-bit program space address. Unlike table operations, this limits remapping operations strictly to the user memory area.

Table 4-38 and Figure 4-9 show how the program EA is created for table operations and remapping accesses from the data EA. Here, P<23:0> refers to a program space word, whereas D<15:0> refers to a data space word

TABLE 4-38: PROGRAM SPACE ADDRESS CONSTRUCTION

Access Tyres	Access	Program Space Address						
Access Type	Space	<23>	<22:16>	<15>	<14:1>	<0>		
Instruction Access	User	0	PC<22:1>					
(Code Execution)		0xxx xxxx xxxx xxxx xxxx xxx0						
TBLRD/TBLWT	User	TB	LPAG<7:0>	Data EA<15:0>				
(Byte/Word Read/Write)		0	xxx xxxx	xx xxxx xxxx				
	Configuration	ТВ	LPAG<7:0>	Data EA<15:0>				
		1	xxx xxxx	xxxx xxxx xxxx xxxx				
Program Space Visibility	User	0	PSVPAG<7	7:0>	Data EA<14:0> <sup>(1)</sup>			
(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 PSVPAG<0>.



# 4.6.2 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

is selected when it is '0'.

selected when byte select is '1'; the lower byte

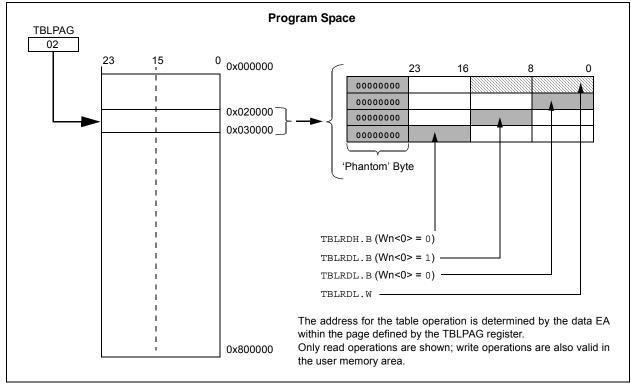
 TBLRDH (Table Read High): In Word mode, it maps the entire upper word of a program address (P<23:16>) to a data address. Note that D<15:8>, the 'phantom' byte, will always be '0'.

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 explained in **Section 5.0** "Flash Program Memory".

For all table operations, the area of program memory space to be accessed is determined by the Table Page 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.

FIGURE 4-10: ACCESSING PROGRAM MEMORY WITH TABLE INSTRUCTIONS



# 4.6.3 READING DATA FROM PROGRAM MEMORY USING PROGRAM SPACE VISIBILITY

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

Program space access through the data space occurs if the Most Significant bit of the data space EA is '1' and program space visibility is enabled by setting the PSV bit in the Core Control register (CORCON<2>). The location of the program memory space to be mapped into the data space is determined by the Program Space Visibility Page register (PSVPAG). This 8-bit register defines any one of 256 possible pages of 16K words in program space. In effect, PSVPAG functions as the upper 8 bits of the program memory address, with the 15 bits of the EA functioning as the lower bits. Note that by incrementing the PC by 2 for each program memory word, the lower 15 bits of data space addresses directly map to the lower 15 bits in the corresponding program space addresses.

Data reads to this area add an additional cycle to the instruction being executed, since two program memory fetches are required.

Although each data space address, 8000h and higher, maps directly into a corresponding program memory address (see Figure 4-11), only the lower 16 bits of the

24-bit program word are used to contain the data. The upper 8 bits of any program space location used as data should be programmed with '1111 1111' or '0000 0000' to force a NOP. This prevents possible issues should the area of code ever be accidentally executed.

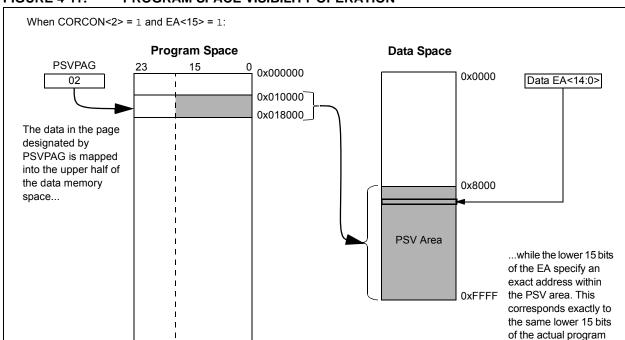
**Note:** PSV access is temporarily disabled during table reads/writes.

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

For operations that use PSV and 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.



0x800000

FIGURE 4-11: PROGRAM SPACE VISIBILITY OPERATION

space address.

N	$\boldsymbol{\cap}$	т	ᆮ	c	
IA	v	ı	ᆮ	J	•

### 5.0 FLASH PROGRAM MEMORY

Note 1: This data sheet summarizes the features of the dsPIC33FJXXXMCX06A/X08A/X10A family of devices. However, it is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to Section 5. "Flash Programming" (DS70191) in the "dsPIC33F/PIC24H Family Reference Manual", which is available from the Microchip web site (www.microchip.com).

2: Some registers and associated bits described in this section may not be available on all devices. Refer to Section 4.0 "Memory Organization" in this data sheet for device-specific register and bit information.

The dsPIC33FJXXXMCX06A/X08A/X10A devices contain internal Flash program memory for storing and executing application code. The memory is readable, writable and erasable during normal operation over the entire VDD range.

Flash memory can be programmed in two ways:

- In-Circuit Serial Programming™ (ICSP™) programming capability
- 2. Run-Time Self-Programming (RTSP)

ICSP allows a dsPIC33FJXXXMCX06A/X08A/X10A device to be serially programmed while in the end application circuit. This is simply done with two lines for programming clock and programming data (one of the alternate programming pin pairs: PGECx/PGEDx), 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 digital signal controller just before shipping the product. This also allows the most recent firmware or a custom firmware to be programmed.

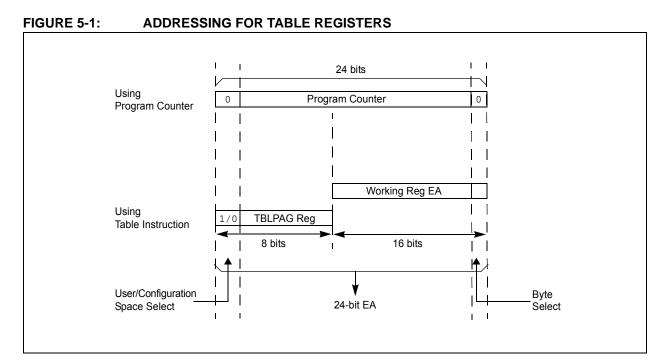
RTSP is accomplished using TBLRD (table read) and TBLWT (table write) instructions. With RTSP, the user can write program memory data by blocks (or 'rows') of 64 instructions (192 bytes) at a time or by single program memory word; the user can erase program memory in blocks or 'pages' of 512 instructions (1536 bytes) at a time.

# 5.1 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 bits<7:0> of the TBLPAG register and the Effective Address (EA) from a W register specified in the table instruction, as shown in Figure 5-1.

The TBLRDL and 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.



#### 5.2 RTSP Operation

The dsPIC33FJXXXMCX06A/X08A/X10A Flash program memory array is organized into rows of 64 instructions or 192 bytes. RTSP allows the user to erase a page of memory at a time, which consists of eight rows (512 instructions), and to program one row or one word at a time. Table 26-12 shows typical erase and programming times. The 8-row erase pages and single row write rows are edge-aligned, from the beginning of program memory, on boundaries of 1536 bytes and 192 bytes, respectively.

The program memory implements holding buffers that can contain 64 instructions of programming data. Prior to the actual programming operation, the write data must be loaded into the buffers in sequential order. The instruction words loaded must always be from a group of 64 boundaries.

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. A total of 64 TBLWTL and TBLWTH instructions are required to load the instructions.

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

#### 5.3 Programming Operations

A complete programming sequence is necessary for programming or erasing the internal Flash in RTSP mode. The processor stalls (waits) until the programming operation is finished.

The programming time depends on the FRC accuracy (see Table 26-19) and the value of the FRC Oscillator Tuning register (see Register 9-4). Use the following formula to calculate the minimum and maximum values for the row write time, page erase time and word write cycle time parameters (see Table 26-12).

#### **EQUATION 5-1: PROGRAMMING TIME**

 $\frac{T}{7.37 \ MHz \times (FRC \ Accuracy)\% \times (FRC \ Tuning)\%}$ 

For example, if the device is operating at +125°C, the FRC accuracy will be ±5%. If the TUN<5:0> bits (see Register 9-4) are set to `b111111, the minimum row write time is equal to Equation 5-2.

## EQUATION 5-2: MINIMUM ROW WRITE

$$T_{RW} = \frac{11064 \ Cycles}{7.37 \ MHz \times (1 + 0.05) \times (1 - 0.00375)} = 1.435 ms$$

The maximum row write time is equal to Equation 5-3.

## EQUATION 5-3: MAXIMUM ROW WRITE TIME

$$T_{RW} = \frac{11064 \ Cycles}{7.37 \ MHz \times (1 - 0.05) \times (1 - 0.00375)} = 1.586ms$$

Setting the WR bit (NVMCON<15>) starts the operation and the WR bit is automatically cleared when the operation is finished.

#### 5.4 Control Registers

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

The NVMCON register (Register 5-1) controls which blocks are to be erased, which memory type is to be programmed and the start of the programming cycle.

NVMKEY is a write-only register that is used for write protection. To start a programming or erase sequence, the user must consecutively write 0x55 and 0xAA to the NVMKEY register. Refer to **Section 5.3 "Programming Operations"** for further details.

#### REGISTER 5-1: NVMCON: FLASH MEMORY CONTROL REGISTER

R/SO-0 <sup>(1)</sup>	R/W-0 <sup>(1)</sup>	R/W-0 <sup>(1)</sup>	U-0	U-0	U-0	U-0	U-0
WR	WREN	WRERR	_	_	_	_	_
bit 15							bit 8

U-0	R/W-0 <sup>(1)</sup>	U-0	U-0	R/W-0 <sup>(1)</sup>	R/W-0 <sup>(1)</sup>	R/W-0 <sup>(1)</sup>	R/W-0 <sup>(1)</sup>
_	ERASE	_	_		NVMOP	<3:0> <sup>(2)</sup>	
bit 7							bit 0

Legend:	SO = Settable Only bit		
R = Readable bit	W = Writable bit	e 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 WR: Write Control bit

1 = Initiates a Flash memory program or erase operation. The operation is self-timed and the bit is cleared by hardware once operation is complete

0 = Program or erase operation is complete and inactive

bit 14 WREN: Write Enable bit

1 = Enable Flash program/erase operations

0 = Inhibit Flash program/erase operations

bit 13 WRERR: Write Sequence Error Flag bit

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 = Perform the erase operation specified by NVMOP<3:0> on the next WR command

0 = Perform the program operation specified by NVMOP<3:0> on the next WR command

bit 5-4 **Unimplemented:** Read as '0'

bit 3-0 **NVMOP<3:0>:** NVM Operation Select bits<sup>(2)</sup>

If ERASE = 1:

1111 = Memory bulk erase operation

1110 = Reserved

1101 = Erase General Segment

1100 = Erase Secure Segment

1011 = Reserved

0011 = No operation

0010 = Memory page erase operation

0001 = No operation

0000 = Erase a single Configuration register byte

If ERASE = 0:

1111 = No operation

1110 = Reserved

1101 = No operation

1100 = No operation

1011 = Reserved

0011 = Memory word program operation

0010 = No operation

0001 = Memory row program operation

0000 = Program a single Configuration register byte

Note 1: These bits can only be reset on POR.

**2:** All other combinations of NVMOP<3:0> are unimplemented.

## 5.4.1 PROGRAMMING ALGORITHM FOR FLASH PROGRAM MEMORY

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

- 1. Read eight rows of program memory (512 instructions) and store it in data RAM.
- 2. Update the program data in RAM with the desired new data.
- 3. Erase the block (see Example 5-1):
  - a) Set the NVMOP 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 page to be erased into the TBLPAG and W registers.
  - c) Write 0x55 to NVMKEY.
  - d) Write 0xAA 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 5-2).
- 5. Write the program block to Flash memory:
  - Set the NVMOP bits to '0001' to configure for row programming. Clear the ERASE bit and set the WREN bit.
  - b) Write 0x55 to NVMKEY.
  - c) Write 0xAA 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 5-3.

#### **EXAMPLE 5-1: ERASING A PROGRAM MEMORY PAGE**

```
; Set up NVMCON for block erase operation
       MOV
              #0x4042, W0
                                             ; Initialize NVMCON
       MOV
              W0, NVMCON
; Init pointer to row to be ERASED
              #tblpage(PROG_ADDR), W0
       MOV
       MOV
              WO, TBLPAG
                                             ; Initialize PM Page Boundary SFR
       MOV
               #tbloffset(PROG_ADDR), W0
                                             ; Initialize in-page EA[15:0] pointer
                                             ; Set base address of erase block
       TBLWTL W0, [W0]
              #5
                                             ; Block all interrupts with priority <7
       DISI
                                             ; for next 5 instructions
       MOV
               #0x55, W0
               WO, NVMKEY
                                             ; Write the 55 key
       MOV
               #0xAA, W1
       MOV
               W1, NVMKEY
                                             ; Write the AA key
       MOV
               NVMCON, #WR
       BSET
                                             ; Start the erase sequence
       NOP
                                             ; Insert two NOPs after the erase
       NOP
                                             ; command is asserted
```

#### **EXAMPLE 5-2: LOADING THE WRITE BUFFERS**

```
; Set up NVMCON for row programming operations
             #0x4001, W0
                                               ; Initialize NVMCON
      MOV
              W0, NVMCON
; Set up a pointer to the first program memory location to be written
; program memory selected, and writes enabled
             #0x0000, W0
      MOV
            W0, TBLPAG
                                              ; Initialize PM Page Boundary SFR
             #0x6000, W0
      MOV
                                              ; An example program memory address
; Perform the TBLWT instructions to write the latches
; Oth_program_word
            #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
      MOV
            #HIGH_BYTE_1, W3
                                              ;
      TBLWTL W2, [W0]
                                              ; Write PM low word into program latch
      TBLWTH W3, [W0++]
                                              ; Write PM high byte into program latch
; 2nd_program_word
             #LOW_WORD_2, W2
      MOV
           #HIGH_BYTE_2, W3
      MOV
                                              ;
      TBLWTL W2, [W0]
                                              ; Write PM low word into program latch
      TBLWTH W3, [W0++]
                                              ; Write PM high byte into program latch
; 63rd_program_word
           #LOW_WORD_31, W2
      MOV
       MOV
              #HIGH_BYTE_31, W3
      TBLWTL W2, [W0]
                                              ; Write PM low word into program latch
      TBLWTH W3, [W0++]
                                              ; Write PM high byte into program latch
```

#### **EXAMPLE 5-3: INITIATING A PROGRAMMING SEQUENCE**

```
DISI
                                 ; Block all interrupts with priority <7
                                 ; for next 5 instructions
MOV
       #0x55, W0
      WO, NVMKEY
                                 ; Write the 55 key
VOM
MOV
       #0xAA, W1
MOV
      W1, NVMKEY
                                ; Write the AA key
BSET NVMCON, #WR
                                 ; Start the erase sequence
NOP
                                 ; Insert two NOPs after the
NOP
                                 ; erase command is asserted
```

dsPIC33FJXXXMCX06A/X08A/X10A						
NOTES:						

#### 6.0 RESET

Note 1: This data sheet summarizes the features of the dsPIC33FJXXXMCX06A/X08A/X10A family of devices. However, it is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to Section 8. "Reset" (DS70192) in the "dsPIC33F/PIC24H Family Reference Manual", which is available from the Microchip web site (www.microchip.com).

2: Some registers and associated bits described in this section may not be available on all devices. Refer to Section 4.0 "Memory Organization" in this data sheet for device-specific register and bit information.

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

· BOR: Brown-out Reset

MCLR: Master Clear Pin Reset

• SWR: RESET Instruction

WDT: Watchdog Timer Reset

TRAPR: Trap Conflict Reset

 IOPUWR: Illegal Opcode and Uninitialized W Register Reset A simplified block diagram of the Reset module is shown in Figure 6-1.

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.

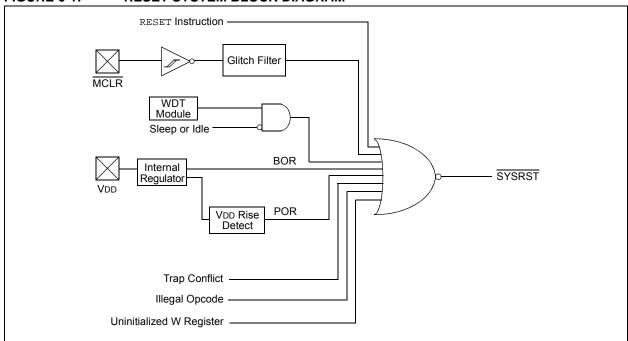
Note: Refer to the specific peripheral or CPU section of this data sheet for register Reset states.

All types of device Reset will set a corresponding status bit in the RCON register to indicate the type of Reset (see Register 6-1). A POR will clear all bits except for the POR bit (RCON<0>), which is set. The user can 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 does 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 manual.

**Note:** The status bits in the RCON register should be cleared after they are read so that the next RCON register value after a device Reset will be meaningful.

FIGURE 6-1: RESET SYSTEM BLOCK DIAGRAM



#### RCON: RESET CONTROL REGISTER<sup>(1)</sup> **REGISTER 6-1:**

R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0	R/W-0
TRAPR	IOPUWR	_	_	_	_	_	VREGS <sup>(3)</sup>
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	SWR	SWDTEN <sup>(2)</sup>	WDTO	SLEEP	IDLE	BOR	POR
bit 7							bit 0

Legend:

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

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

bit 15 TRAPR: Trap Reset Flag bit

> 1 = A Trap Conflict Reset has occurred 0 = A Trap Conflict Reset has not occurred

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

1 = An illegal opcode detection, an illegal address mode or uninitialized W register used as an

Address Pointer caused a Reset

0 = An illegal opcode or uninitialized W Reset has not occurred

bit 13-9 Unimplemented: Read as '0'

**VREGS:** Voltage Regulator Standby During Sleep bit<sup>(3)</sup> bit 8

1 = Voltage regulator is active during Sleep mode

0 = Voltage regulator goes into Standby mode during Sleep

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

> 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 = A RESET instruction has been executed 0 = A RESET instruction has not been executed

SWDTEN: Software Enable/Disable of WDT bit(2) bit 5

1 = WDT is enabled

0 = WDT is disabled

bit 4 WDTO: Watchdog Timer Time-out Flag bit

1 = WDT time-out has occurred

0 = WDT time-out has not occurred

bit 3 SLEEP: Wake-up from Sleep Flag bit

1 = Device has been in Sleep mode

0 = Device has not been in Sleep mode

bit 2 IDLE: Wake-up from Idle Flag bit

1 = Device was in Idle mode

0 = Device was not in Idle mode

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 FWDTEN Configuration bit is '1' (unprogrammed), the WDT is always enabled, regardless of the SWDTEN bit setting.
- 3: For dsPIC33FJ256MCX06A/X08A/X10A devices, this bit is unimplemented and reads back a programmed value.

REGISTER 6-1: RCON: RESET CONTROL REGISTER<sup>(1)</sup> (CONTINUED)

bit 1 BOR: Brown-out Reset Flag bit

1 = A Brown-out Reset has occurred0 = A Brown-out Reset has not occurred

bit 0 POR: Power-on Reset Flag bit

1 = A Power-on Reset has occurred 0 = A Power-on Reset has not occurred

- **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 FWDTEN Configuration bit is '1' (unprogrammed), the WDT is always enabled, regardless of the SWDTEN bit setting.
  - **3:** For dsPIC33FJ256MCX06A/X08A/X10A devices, this bit is unimplemented and reads back a programmed value.

TABLE 6-1: RESET FLAG BIT OPERATION

Flag Bit	Setting Event	Clearing Event	
TRAPR (RCON<15>)	Trap conflict event	POR, BOR	
IOPUWR (RCON<14>)	Illegal opcode or uninitialized W register access	POR, BOR	
EXTR (RCON<7>)	MCLR Reset	POR	
SWR (RCON<6>)	RESET instruction	POR, BOR	
WDTO (RCON<4>)	WDT time-out	PWRSAV instruction, POR, BOR	
SLEEP (RCON<3>)	PWRSAV #SLEEP instruction	POR, BOR	
IDLE (RCON<2>)	PWRSAV #IDLE instruction	POR, BOR	
BOR (RCON<1>)	BOR, POR	_	
POR (RCON<0>)	POR	_	

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

#### 6.1 Clock Source Selection at Reset

If clock switching is enabled, the system clock source at device Reset is chosen, as shown in Table 6-2. If clock switching is disabled, the system clock source is always selected according to the oscillator Configuration bits. Refer to **Section 9.0** "Oscillator Configuration" for further details.

TABLE 6-2: OSCILLATOR SELECTION vs.
TYPE OF RESET (CLOCK
SWITCHING ENABLED)

Reset Type	Clock Source Determinant
POR	Oscillator Configuration bits
BOR	(FNOSC<2:0>)
MCLR	COSC Control bits
WDTR	(OSCCON<14:12>)
SWR	

#### 6.2 Device Reset Times

The Reset times for various types of device Reset are summarized in Table 6-3. The System Reset signal, SYSRST, is released after the POR and PWRT delay times expire.

The time at which the device actually begins to execute code also depends 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 FSCM delay determines the time at which the FSCM begins to monitor the system clock source after the SYSRST signal is released.

TABLE 6-3: RESET DELAY TIMES FOR VARIOUS DEVICE RESETS

Reset Type	Clock Source	Clock Source SYSRST Delay		FSCM Delay	See Notes
POR	EC, FRC, LPRC	TPOR + TSTARTUP + TRST	_	_	1, 2, 3
	ECPLL, FRCPLL	TPOR + TSTARTUP + TRST	TLOCK	TFSCM	1, 2, 3, 5, 6
	XT, HS, SOSC	TPOR + TSTARTUP + TRST	Tost	TFSCM	1, 2, 3, 4, 6
	XTPLL, HSPLL	TPOR + TSTARTUP + TRST	Tost + Tlock	TFSCM	1, 2, 3, 4, 5, 6
BOR	EC, FRC, LPRC	TSTARTUP + TRST	_	_	3
	ECPLL, FRCPLL	TSTARTUP + TRST	TLOCK	TFSCM	3, 5, 6
	XT, HS, SOSC	TSTARTUP + TRST	Tost	TFSCM	3, 4, 6
	XTPLL, HSPLL	TSTARTUP + TRST	Tost + Tlock	TFSCM	3, 4, 5, 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  $\mu$ s nominal).
  - 2: TSTARTUP = Conditional POR delay of 20 μs nominal (if on-chip regulator is enabled) or 64 ms nominal Power-up Timer delay (if regulator is disabled). TSTARTUP is also applied to all returns from powered-down states, including waking from Sleep mode if the regulator is enabled.
  - 3: TRST = Internal state Reset time (20  $\mu$ s nominal).
  - **4:** Tost = Oscillator Start-up Timer. A 10-bit counter counts 1024 oscillator periods before releasing the oscillator clock to the system.
  - 5: TLOCK = PLL lock time (20  $\mu$ s nominal).
  - **6:** TFSCM = Fail-Safe Clock Monitor delay (100 μs nominal).

## 6.2.1 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) have a relatively long start-up time. Therefore, one or more of the following conditions is possible after 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.

## 6.2.2 FAIL-SAFE CLOCK MONITOR (FSCM) AND DEVICE RESETS

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

## 6.2.2.1 FSCM Delay for Crystal and PLL Clock Sources

When the system clock source is provided by a crystal oscillator and/or the PLL, a small delay, TFscM, is automatically inserted after the POR and PWRT delay times. The FSCM does not begin to monitor the system clock source until this delay expires. The FSCM delay time is nominally 500  $\mu s$  and provides additional time for the oscillator and/or PLL to stabilize. In most cases, the FSCM delay prevents an oscillator failure trap at a device Reset when the PWRT is disabled.

## 6.3 Special Function Register Reset States

Most of the Special Function Registers (SFRs) associated with the 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 two registers. The Reset value for the Reset Control register, RCON, depends on the type of device Reset. The Reset value for the Oscillator Control register, OSCCON, depends on the type of Reset and the programmed values of the oscillator Configuration bits in the FOSC Configuration register.

#### 7.0 INTERRUPT CONTROLLER

Note 1: This data sheet summarizes the features of the dsPIC33FJXXXMCX06A/X08A/X10A family of devices. However, it is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to Section 6. "Interrupts" (DS70184) in the "dsPIC33F/PIC24H Family Reference Manual", which is available from the Microchip web site (www.microchip.com).

2: Some registers and associated bits described in this section may not be available on all devices. Refer to Section 4.0 "Memory Organization" in this data sheet for device-specific register and bit information.

The interrupt controller for the dsPIC33FJXXXMCX06A/X08A/X10A family of devices reduces the numerous peripheral interrupt request signals to a single interrupt request signal to the dsPIC33FJXXXMCX06A/X08A/X10A CPU. It has the following features:

- Up to eight processor exceptions and software traps
- Seven user-selectable priority levels
- Interrupt Vector Table (IVT) with up to 118 vectors
- A 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

#### 7.1 Interrupt Vector Table

The Interrupt Vector Table (IVT) is shown in Figure 7-1. The IVT resides in program memory, starting at location 000004h. The IVT contains 126 vectors consisting of eight nonmaskable 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 priority 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.

The dsPIC33FJXXXMCX06A/X08A/X10A family of devices implement up to 67 unique interrupts and five nonmaskable traps. These are summarized in Table 7-1 and Table 7-2.

## 7.1.1 ALTERNATE INTERRUPT VECTOR TABLE

The Alternate Interrupt Vector Table (AIVT) is located after the IVT, as shown in Figure 7-1. Access to the AIVT is provided by the ALTIVT control bit (INTCON2<15>). If the ALTIVT bit is set, all interrupt and exception processes 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 debugging 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.

#### 7.2 Reset Sequence

A device Reset is not a true exception because the interrupt controller is not involved in the Reset process. The dsPIC33FJXXXMCX06A/X08A/X10A device clears its registers in response to a Reset, which forces the PC to zero. The digital signal controller then begins program execution at location 0x000000. 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.

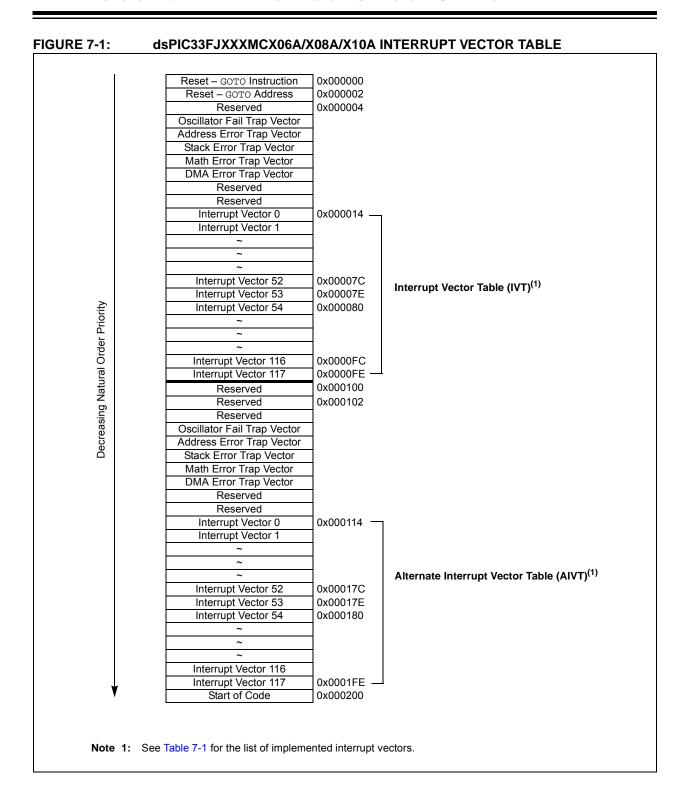


TABLE 7-1: INTERRUPT VECTORS

Vector Number	Interrupt Request (IRQ) Number	IVT Address	AIVT Address	Interrupt Source
8	0	0x000014	0x000114	INT0 – External Interrupt 0
9	1	0x000016	0x000116	IC1 – Input Capture 1
10	2	0x000018	0x000118	OC1 – Output Compare 1
11	3	0x00001A	0x00011A	T1 – Timer1
12	4	0x00001C	0x00011C	DMA0 – DMA Channel 0
13	5	0x00001E	0x00011E	IC2 – Input Capture 2
14	6	0x000020	0x000120	OC2 – Output Compare 2
15	7	0x000022	0x000122	T2 – Timer2
16	8	0x000024	0x000124	T3 – Timer3
17	9	0x000026	0x000126	SPI1E – SPI1 Error
18	10	0x000028	0x000128	SPI1 – SPI1 Transfer Done
19	11	0x00002A	0x00012A	U1RX – UART1 Receiver
20	12	0x00002C	0x00012C	U1TX – UART1 Transmitter
21	13	0x00002E	0x00012E	ADC1 – ADC 1
22	14	0x000030	0x000130	DMA1 – DMA Channel 1
23	15	0x000032	0x000132	Reserved
24	16	0x000034	0x000134	SI2C1 – I2C1 Slave Events
25	17	0x000036	0x000136	MI2C1 – I2C1 Master Events
26	18	0x000038	0x000138	Reserved
27	19	0x00003A	0x00013A	Change Notification Interrupt
28	20	0x00003C	0x00013C	INT1 – External Interrupt 1
29	21	0x00003E	0x00013E	ADC2 – ADC 2
30	22	0x000040	0x000140	IC7 – Input Capture 7
31	23	0x000042	0x000142	IC8 – Input Capture 8
32	24	0x000044	0x000114	DMA2 – DMA Channel 2
33	25	0x000046	0x000146	OC3 – Output Compare 3
34	26	0x000048	0x000148	OC4 – Output Compare 4
35	27	0x00004A	0x00014A	T4 – Timer4
36	28	0x00004C	0x00014C	T5 – Timer5
37	29	0x00004E	0x00014E	INT2 – External Interrupt 2
38	30	0x000050	0x000150	U2RX – UART2 Receiver
39	31	0x000052	0x000152	U2TX – UART2 Transmitter
40	32	0x000054	0x000154	SPI2E – SPI2 Error
41	33	0x000056	0x000156	SPI1 – SPI1 Transfer Done
42	34	0x000058	0x000158	C1RX – ECAN1 Receive Data Ready
43	35	0x00005A	0x00015A	C1 – ECAN1 Event
44	36	0x00005C	0x00015C	DMA3 – DMA Channel 3
45	37	0x00005E	0x00015E	IC3 – Input Capture 3
46	38	0x000060	0x000160	IC4 – Input Capture 4
47	39	0x000062	0x000162	IC5 – Input Capture 5
48	40	0x000064	0x000164	IC6 – Input Capture 6
49	41	0x000066	0x000166	OC5 – Output Compare 5
50	42	0x000068	0x000168	OC6 – Output Compare 6
51	43	0x00006A	0x00016A	OC7 – Output Compare 7
52	44	0x00006C	0x00016C	OC8 – Output Compare 8
53	45	0x00006E	0x00016E	Reserved

TABLE 7-1: INTERRUPT VECTORS (CONTINUED)

Vector Number	Interrupt Request (IRQ) Number	IVT Address	AIVT Address	Interrupt Source
54	46	0x000070	0x000170	DMA4 – DMA Channel 4
55	47	0x000072	0x000172	T6 – Timer6
56	48	0x000074	0x000174	T7 – Timer7
57	49	0x000076	0x000176	SI2C2 – I2C2 Slave Events
58	50	0x000078	0x000178	MI2C2 – I2C2 Master Events
59	51	0x00007A	0x00017A	T8 – Timer8
60	52	0x00007C	0x00017C	T9 – Timer9
61	53	0x00007E	0x00017E	INT3 – External Interrupt 3
62	54	0x000080	0x000180	INT4 – External Interrupt 4
63	55	0x000082	0x000182	C2RX – ECAN2 Receive Data Ready
64	56	0x000084	0x000184	C2 – ECAN2 Event
65	57	0x000086	0x000186	PWM – PWM Period Match
66	58	0x000088	0x000188	QEI – Position Counter Compare
69	61	0x00008E	0x00018E	DMA5 – DMA Channel 5
70	62	0x000090	0x000190	Reserved
71	63	0x000092	0x000192	FLTA – MCPWM Fault A
72	64	0x000094	0x000194	FLTB – MCPWM Fault B
73	65	0x000096	0x000196	U1E – UART1 Error
74	66	0x000098	0x000198	U2E – UART2 Error
75	67	0x00009A	0x00019A	Reserved
76	68	0x00009C	0x00019C	DMA6 – DMA Channel 6
77	69	0x00009E	0x00019E	DMA7 – DMA Channel 7
78	70	0x0000A0	0x0001A0	C1TX – ECAN1 Transmit Data Request
79	71	0x0000A2	0x0001A2	C2TX – ECAN2 Transmit Data Request
80-125	72-117	0x0000A4- 0x0000FE	0x0001A4- 0x0001FE	Reserved

TABLE 7-2: TRAP VECTORS

Vector Number	IVT Address	AIVT Address	Trap Source
0	0x000004	0x000104	Reserved
1	0x000006	0x000106	Oscillator Failure
2	0x000008	0x000108	Address Error
3	0x00000A	0x00010A	Stack Error
4	0x00000C	0x00010C	Math Error
5	0x00000E	0x00010E	DMA Error Trap
6	0x000010	0x000110	Reserved
7	0x000012	0x000112	Reserved

# 7.3 Interrupt Control and Status Registers

dsPIC33FJXXXMCX06A/X08A/X10A devices implement a total of 30 registers for the interrupt controller:

- INTCON1
- INTCON2
- · IFS0 through IFS4
- IEC0 through IEC4
- IPC0 through IPC17
- INTTREG

Global interrupt control functions are controlled from INTCON1 and INTCON2. INTCON1 contains the Interrupt Nesting Disable bit (NSTDIS) 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.

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

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

The IPC registers are used to set the interrupt priority level 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 vector number (VECNUM<6:0>) and Interrupt level bit (ILR<3:0>) 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 same sequence that they are listed in Table 7-1. For example, the INT0 (External Interrupt 0) is shown as having vector number 8 and a natural order priority of 0. Thus, the INT0IF bit is found in IFS0<0>, the INT0IE bit in IEC0<0> and the INT0IP bits in the first position of IPC0 (IPC0<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 CPU STATUS register, SR, contains the IPL<2:0> bits (SR<7:5>). These bits indicate the current CPU interrupt priority level. The user can change the current CPU priority level by writing to the IPL bits.

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

All Interrupt registers are described in Register 7-1 through Register 7-32 in the following pages.

#### REGISTER 7-1: SR: CPU STATUS REGISTER<sup>(1)</sup>

R-0	R-0	R/C-0	R/C-0	R-0	R/C-0	R-0	R/W-0
OA	ОВ	SA	SB	OAB	SAB	DA	DC
bit 15							bit 8

R/W-0 <sup>(3)</sup>	R/W-0 <sup>(3)</sup>	R/W-0 <sup>(3)</sup>	R-0	R/W-0	R/W-0	R/W-0	R/W-0
IPL2 <sup>(2)</sup>	IPL1 <sup>(2)</sup>	IPL0 <sup>(2)</sup>	RA	N	OV	Z	С
bit 7							bit 0

Legend:

C = Clearable bit R = Readable bit U = Unimplemented bit, read as '0'

S = Settable bit W = Writable bit -n = Value at POR '1' = Bit is set '0' = Bit is cleared x = Bit is unknown

bit 7-5 IPL<2:0>: CPU Interrupt Priority Level Status bits<sup>(2)</sup>

111 = CPU interrupt priority level is 7 (15), user interrupts 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)

Note 1: For complete register details, see Register 3-1: "SR: CPU STATUS Register".

2: The IPL<2:0> bits are concatenated with the IPL<3> bit (CORCON<3>) to form the CPU interrupt priority level. The value in parentheses indicates the IPL if IPL<3> = 1. User interrupts are disabled when IPL<3> = 1.

3: The IPL<2:0> status bits are read-only when NSTDIS (INTCON1<15>) = 1.

#### REGISTER 7-2: CORCON: CORE CONTROL REGISTER<sup>(1)</sup>

U-0	U-0	U-0	R/W-0	R/W-0	R-0	R-0	R-0
_	_	_	US	EDT		DL<2:0>	
bit 15							bit 8

R/W-0	R/W-0	R/W-1	R/W-0	R/C-0	R/W-0	R/W-0	R/W-0
SATA	SATB	SATDW	ACCSAT	IPL3 <sup>(2)</sup>	PSV	RND	IF
bit 7							bit 0

**Legend:** C = Clearable bit

R = Readable bit W = Writable bit -n = Value at POR '1' = Bit is set

0' = Bit is cleared 'x = Bit is unknown U = Unimplemented bit, read as '0'

bit 3 IPL3: CPU Interrupt Priority Level Status bit 3<sup>(2)</sup>

1 = CPU interrupt priority level is greater than 7

0 = CPU interrupt priority level is 7 or less

Note 1: For complete register details, see Register 3-2: "CORCON: CORE Control Register".

2: The IPL3 bit is concatenated with the IPL<2:0> bits (SR<7:5>) to form the CPU interrupt priority level.

#### REGISTER 7-3: INTCON1: INTERRUPT 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
NSTDIS	OVAERR	OVBERR	COVAERR	COVBERR	OVATE	OVBTE	COVTE
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
SFTACERR	DIV0ERR	DMACERR	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	OVAERR: Accumulator A Overflow Trap Flag bit
	1 = Trap was caused by overflow of Accumulator A
	0 = Trap was not caused by overflow of Accumulator A
bit 13	OVBERR: Accumulator B Overflow Trap Flag bit
	1 = Trap was caused by overflow of Accumulator B
	0 = Trap was not caused by overflow of Accumulator B
bit 12	COVAERR: Accumulator A Catastrophic Overflow Trap Flag bit
	1 = Trap was caused by catastrophic overflow of Accumulator A
	0 = Trap was not caused by catastrophic overflow of Accumulator A
bit 11	COVBERR: Accumulator B Catastrophic Overflow Trap Flag bit
	1 = Trap was caused by catastrophic overflow of Accumulator B
	0 = Trap was not caused by catastrophic overflow of Accumulator B
bit 10	<b>OVATE:</b> Accumulator A Overflow Trap Enable bit
	1 = Trap overflow of Accumulator A
	0 = Trap disabled
bit 9	<b>OVBTE:</b> Accumulator B Overflow Trap Enable bit
	1 = Trap overflow of Accumulator B
	0 = Trap disabled
bit 8	COVTE: Catastrophic Overflow Trap Enable bit
	1 = Trap on catastrophic overflow of Accumulator A or B enabled
	0 = Trap disabled
bit 7	SFTACERR: Shift Accumulator Error Status bit
	1 = Math error trap was caused by an invalid accumulator shift
	0 = Math error trap was not caused by an invalid accumulator shift
bit 6	DIVOERR: Arithmetic Error Status bit
	1 = Math error trap was caused by a divide by zero
=	0 = Math error trap was not caused by a divide by zero
bit 5	DMACERR: DMA Controller Error Status bit
	1 = DMA controller error trap has occurred
L:1 4	0 = DMA controller error trap has not occurred
bit 4	MATHERR: Arithmetic Error Status bit
	1 = Math error trap has occurred 0 = Math error trap has not occurred
	o - Maur error trap has not occurred

#### REGISTER 7-3: INTCON1: INTERRUPT CONTROL REGISTER 1 (CONTINUED)

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 occurred0 = Oscillator failure trap has not occurred

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

#### REGISTER 7-4: INTCON2: INTERRUPT CONTROL REGISTER 2

R/W-0	R-0	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:

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 = Use Alternate Interrupt Vector Table0 = Use standard (default) 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 7-5: 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	SPI1IF	SPI1EIF	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	DMA01IF	T1IF	OC1IF	IC1IF	INT0IF
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 DMA1IF: DMA Channel 1 Data Transfer Complete Interrupt Flag Status bit

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

bit 13 AD1IF: ADC1 Conversion Complete 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 occurred0 = 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 SPI1IF: SPI1 Event Interrupt Flag Status bit

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

bit 9 SPI1EIF: SPI1 Fault Interrupt Flag Status bit

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

bit 8 T3IF: Timer3 Interrupt Flag Status bit

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

bit 7 T2IF: Timer2 Interrupt Flag Status bit

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

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

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

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

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

bit 4 DMA0IF: DMA Channel 0 Data Transfer Complete Interrupt Flag Status bit

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

bit 3 T1IF: Timer1 Interrupt Flag Status bit

#### REGISTER 7-5: 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 7-6: 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	DMA21IF
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
IC8IF	IC7IF	AD2IF	INT1IF	CNIF	_	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 occurred0 = Interrupt request has not occurred

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

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

bit 13 INT2IF: External Interrupt 2 Flag Status bit

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

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

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

bit 8 DMA2IF: DMA Channel 2 Data Transfer Complete Interrupt Flag Status bit

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

bit 7 IC8IF: Input Capture Channel 8 Interrupt Flag Status bit

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

bit 6 IC7IF: Input Capture Channel 7 Interrupt Flag Status bit

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

bit 5 AD2IF: ADC2 Conversion Complete Interrupt Flag Status bit

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

bit 4 INT1IF: External Interrupt 1 Flag Status bit

### REGISTER 7-6: IFS1: INTERRUPT FLAG STATUS REGISTER 1 (CONTINUED)

bit 3 CNIF: Input Change Notification Interrupt Flag Status bit

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

bit 2 Unimplemented: Read as '0'

bit 1 MI2C1IF: I2C1 Master Events Interrupt Flag Status bit

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

bit 0 SI2C1IF: I2C1 Slave Events Interrupt Flag Status bit

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

R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
T6IF	DMA4IF	_	OC8IF	OC7IF	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	C1IF	C1RXIF	SPI2IF	SPI2EIF
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 **T6IF:** Timer6 Interrupt Flag Status bit

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

bit 14 DMA4IF: DMA Channel 4 Data Transfer Complete Interrupt Flag Status bit

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

bit 13 Unimplemented: Read as '0'

bit 12 OC8IF: Output Compare Channel 8 Interrupt Flag Status bit

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

bit 11 OC7IF: Output Compare Channel 7 Interrupt Flag Status bit

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

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 Data Transfer Complete Interrupt Flag Status bit

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

bit 3 C1IF: ECAN1 Event Interrupt Flag Status bit

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

bit 2 C1RXIF: ECAN1 Receive Data Ready Interrupt Flag Status bit

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

bit 1 SPI2IF: SPI2 Event Interrupt Flag Status bit

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

bit 0 SPI2EIF: SPI2 Error Interrupt Flag Status bit

#### REGISTER 7-8: IFS3: INTERRUPT FLAG STATUS REGISTER 3

R/W-0	U-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0
FLTAIF	_	DMA5IF	_	_	QEIIF	PWMIF	C2IF
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
C2RXIF	INT4IF	INT3IF	T9IF	T8IF	MI2C2IF	SI2C2IF	T7IF
bit 7							bit 0

Legend:

bit 9

bit 8

bit 3

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 FLTAIF: PWM Fault A Interrupt Flag Status bit

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

bit 14 Unimplemented: Read as '0'

bit 13 DMA5IF: DMA Channel 5 Data Transfer Complete Interrupt Flag Status bit

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

bit 12-11 Unimplemented: Read as '0'

bit 10 **QEIIF:** QEI Event Interrupt Flag Status bit

1 = Interrupt request has occurred0 = Interrupt request has not occurredPWMIF: PWM Interrupt Flag Status bit

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

C2IF: ECAN2 Event Interrupt Flag Status bit

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

bit 7 C2RXIF: ECAN2 Receive Data Ready Interrupt Flag Status bit

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

bit 6 INT4IF: External Interrupt 4 Flag Status bit

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

bit 5 INT3IF: External Interrupt 3 Flag Status bit

1 = Interrupt request has occurred

0 = Interrupt request has not occurred

bit 4 **T9IF:** Timer9 Interrupt Flag Status bit

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

T8IF: Timer8 Interrupt Flag Status bit

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

bit 2 MI2C2IF: I2C2 Master Events Interrupt Flag Status bit

### REGISTER 7-8: IFS3: INTERRUPT FLAG STATUS REGISTER 3 (CONTINUED)

bit 1 SI2C2IF: I2C2 Slave Events Interrupt Flag Status bit

1 = Interrupt request has occurred

0 = Interrupt request has not occurred

bit 0 T7IF: Timer7 Interrupt Flag Status bit

#### REGISTER 7-9: IFS4: INTERRUPT FLAG STATUS REGISTER 4

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	U-0	R/W-0	R/W-0	R/W-0
C2TXIF	C1TXIF	DMA7IF	DMA6IF	_	U2EIF	U1EIF	FLTBIF
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 C2TXIF: ECAN2 Transmit Data Request Interrupt Flag Status bit

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

bit 6 C1TXIF: ECAN1 Transmit Data Request Interrupt Flag Status bit

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

bit 5 DMA7IF: DMA Channel 7 Data Transfer Complete Interrupt Flag Status bit

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

bit 4 DMA6IF: DMA Channel 6 Data Transfer Complete Interrupt Flag Status bit

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

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

bit 2 **U2EIF:** UART2 Error Interrupt Flag Status bit

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

bit 1 **U1EIF:** UART1 Error Interrupt Flag Status bit

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

bit 0 FLTBIF: PWM Fault B Interrupt Flag Status bit

#### REGISTER 7-10: 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	SPI1IE	SPI1EIE	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:

bit 8

bit 3

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 Data Transfer Complete Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 13 AD1IE: ADC1 Conversion Complete Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

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

1 = Interrupt request enabled0 = Interrupt request not enabled

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

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 10 SPI1IE: SPI1 Event Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 9 SPI1EIE: SPI1 Error Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabledT3IE: Timer3 Interrupt Enable bit

1 = Interrupt request enabled

0 = Interrupt request not enabled

bit 7 **T2IE:** Timer2 Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 6 OC2IE: Output Compare Channel 2 Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 5 IC2IE: Input Capture Channel 2 Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 4 DMA0IE: DMA Channel 0 Data Transfer Complete Interrupt Enable bit

I = Interrupt request enabled
 Tile: Timer1 Interrupt Enable bit

#### REGISTER 7-10: IEC0: INTERRUPT ENABLE CONTROL REGISTER 0 (CONTINUED)

bit 2 OC1IE: Output Compare Channel 1 Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 1 IC1IE: Input Capture Channel 1 Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 0 **INTOIE:** External Interrupt 0 Enable bit

#### REGISTER 7-11: 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	T5IE	T4IE	OC4IE	OC3IE	DMA2IE
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
IC8IE	IC7IE	AD2IE	INT1IE	CNIE	_	MI2C1IE	SI2C1IE
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 **U2TXIE:** UART2 Transmitter Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 14 **U2RXIE:** UART2 Receiver Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 13 INT2IE: External Interrupt 2 Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 12 **T5IE:** Timer5 Interrupt Enable bit

1 = Interrupt request enabled

0 = Interrupt request not enabled

bit 11 T4IE: Timer4 Interrupt Enable bit

1 = Interrupt request enabled

0 = Interrupt request not enabled

bit 10 OC4IE: Output Compare Channel 4 Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 9 OC3IE: Output Compare Channel 3 Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 8 DMA2IE: DMA Channel 2 Data Transfer Complete Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 7 IC8IE: Input Capture Channel 8 Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 6 IC7IE: Input Capture Channel 7 Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 5 AD2IE: ADC2 Conversion Complete Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 4 **INT1IE:** External Interrupt 1 Enable bit

### REGISTER 7-11: IEC1: INTERRUPT ENABLE CONTROL REGISTER 1 (CONTINUED)

bit 3 CNIE: Input Change Notification Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

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

bit 1 MI2C1IE: I2C1 Master Events Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 0 SI2C1IE: I2C1 Slave Events Interrupt Enable bit

#### **REGISTER 7-12: IEC2: INTERRUPT ENABLE CONTROL REGISTER 2**

R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
T6IE	DMA4IE	_	OC8IE	OC7IE	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	C1IE	C1RXIE	SPI2IE	SPI2EIE
bit 7 bit 0							

R = Readable bit	W = W

Legend:

Vritable bit U = Unimplemented bit, read as '0'

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

bit 15 T6IE: Timer6 Interrupt Enable bit

1 = Interrupt request enabled

0 = Interrupt request not enabled

bit 14 DMA4IE: DMA Channel 4 Data Transfer Complete Interrupt Enable bit

> 1 = Interrupt request enabled 0 = Interrupt request not enabled

bit 13 Unimplemented: Read as '0'

bit 12 OC8IE: Output Compare Channel 8 Interrupt Enable bit

> 1 = Interrupt request enabled 0 = Interrupt request not enabled

bit 11 **OC7IE:** Output Compare Channel 7 Interrupt Enable bit

> 1 = Interrupt request enabled 0 = Interrupt request not enabled

bit 10 OC6IE: Output Compare Channel 6 Interrupt Enable bit

> 1 = Interrupt request enabled 0 = Interrupt request not enabled

bit 9 OC5IE: Output Compare Channel 5 Interrupt Enable bit

> 1 = Interrupt request enabled 0 = Interrupt request not enabled

bit 8 IC6IE: Input Capture Channel 6 Interrupt Enable bit

> 1 = Interrupt request enabled 0 = Interrupt request not enabled

bit 7 IC5IE: Input Capture Channel 5 Interrupt Enable bit

> 1 = Interrupt request enabled 0 = Interrupt request not enabled

bit 6 IC4IE: Input Capture Channel 4 Interrupt Enable bit

> 1 = Interrupt request enabled 0 = Interrupt request not enabled

bit 5 IC3IE: Input Capture Channel 3 Interrupt Enable bit

> 1 = Interrupt request enabled 0 = Interrupt request not enabled

bit 4 **DMA3IE:** DMA Channel 3 Data Transfer Complete Interrupt Enable bit

> 1 = Interrupt request enabled 0 = Interrupt request not enabled

bit 3 C1IE: ECAN1 Event Interrupt Enable bit

### REGISTER 7-12: IEC2: INTERRUPT ENABLE CONTROL REGISTER 2 (CONTINUED)

bit 2 C1RXIE: ECAN1 Receive Data Ready Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 1 SPI2IE: SPI2 Event Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 0 SPI2EIE: SPI2 Error Interrupt Enable bit

#### REGISTER 7-13: IEC3: INTERRUPT ENABLE CONTROL REGISTER 3

R/W-0	U-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0
FLTAIE	_	DMA5IE	_	_	QEIIE	PWMIE	C2IE
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
C2RXIE	INT4IE	INT3IE	T9IE	T8IE	MI2C2IE	SI2C2IE	T7IE
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 FLTAIE: PWM Fault A Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 14 Unimplemented: Read as '0'

bit 13 DMA5IE: DMA Channel 5 Data Transfer Complete Interrupt Enable bit

1 = Interrupt request enabled 0 = Interrupt request not enabled

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

bit 10 **QEIIE:** QEI Event Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 9 **PWMIE:** PWM Error Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 8 C2IE: ECAN2 Event Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 7 C2RXIE: ECAN2 Receive Data Ready Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 6 INT4IE: External Interrupt 4 Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 5 INT3IE: External Interrupt 3 Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 4 T9IE: Timer9 Interrupt Enable bit

1 = Interrupt request enabled 0 = Interrupt request not enabled

bit 3 **T8IE**: Timer8 Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 2 MI2C2IE: I2C2 Master Events Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

#### REGISTER 7-13: IEC3: INTERRUPT ENABLE CONTROL REGISTER 3 (CONTINUED)

bit 1 SI2C2IE: I2C2 Slave Events Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

**T7IE:** Timer7 Interrupt Enable bit

bit 0

1 = Interrupt request enabled

0 = Interrupt request not enabled

#### REGISTER 7-14: IEC4: INTERRUPT ENABLE CONTROL REGISTER 4

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	U-0	R/W-0	R/W-0	R/W-0
C2TXIE	C1TXIE	DMA7IE	DMA6IE	_	U2EIE	U1EIE	FLTBIE
bit 7							bit 0

Legend:

bit 3

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 C2TXIE: ECAN2 Transmit Data Request Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 6 C1TXIE: ECAN1 Transmit Data Request Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 5 DMA7IE: DMA Channel 7 Data Transfer Complete Enable Status bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 4 DMA6IE: DMA Channel 6 Data Transfer Complete Enable Status bit

1 = Interrupt request enabled 0 = Interrupt request not enabled Unimplemented: Read as '0'

bit 2 **U2EIE:** UART2 Error Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 1 **U1EIE:** UART1 Error Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 0 FLTBIE: PWM Fault B Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

#### REGISTER 7-15: 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
_		T1IP<2:0>		_		OC1IP<2:0>	
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
_		IC1IP<2:0>		_		INT0IP<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 14-12 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)

.

٠

•

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 7-16: 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
_		T2IP<2:0>		_		OC2IP<2:0>	
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
_		IC2IP<2:0>		_		DMA0IP<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 14-12 **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 Data Transfer Complete Interrupt Priority bits

111 = Interrupt is priority 7 (highest priority interrupt)

•

•

•

001 = Interrupt is priority 1

#### REGISTER 7-17: 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
_		U1RXIP<2:0>		_		SPI1IP<2:0>	
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
_		SPI1EIP<2:0>		_		T3IP<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 14-12 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 SPI1IP<2:0>: SPI1 Event 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 SPI1EIP<2:0>: SPI1 Error 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 7-18: 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
_	_	_	_	_		DMA1IP<2:0>	
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
_		AD1IP<2:0>		_		U1TXIP<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 15-11 **Unimplemented:** Read as '0'

bit 10-8 DMA1IP<2:0>: DMA Channel 1 Data Transfer Complete 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 Conversion Complete 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 7-19: IPC4: INTERRUPT PRIORITY CONTROL REGISTER 4

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
_		CNIP<2:0>		_	_	_	_
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
_		MI2C1IP<2:0>		_		SI2C1IP<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 14-12 CNIP<2:0>: Change Notification Interrupt Priority bits

111 = Interrupt is priority 7 (highest priority interrupt)

•

.

001 = Interrupt is priority 1

000 = Interrupt source is disabled

bit 11-7 **Unimplemented:** Read as '0'

bit 6-4 MI2C1IP<2:0>: I2C1 Master Events 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>: I2C1 Slave Events Interrupt Priority bits

111 = Interrupt is priority 7 (highest priority interrupt)

•

•

001 = Interrupt is priority 1

#### REGISTER 7-20: IPC5: INTERRUPT PRIORITY CONTROL REGISTER 5

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_		IC8IP<2:0>		_		IC7IP<2:0>	
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
_		AD2IP<2:0>		_		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 **Unimplemented:** Read as '0'

bit 14-12 IC8IP<2:0>: Input Capture Channel 8 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 IC7IP<2:0>: Input Capture Channel 7 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 AD2IP<2:0>: ADC2 Conversion Complete 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 **INT1IP<2:0>:** External Interrupt 1 Priority bits

111 = Interrupt is priority 7 (highest priority interrupt)

•

•

•

001 = Interrupt is priority 1

#### REGISTER 7-21: 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
_		T4IP<2:0>		_		OC4IP<2:0>	
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
_		OC3IP<2:0>		_		DMA2IP<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 14-12 **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 Data Transfer Complete Interrupt Priority bits

111 = Interrupt is priority 7 (highest priority interrupt)

.

•

001 = Interrupt is priority 1

#### REGISTER 7-22: 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
_		U2TXIP<2:0>		_		U2RXIP<2:0>	
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
_		INT2IP<2:0>		_		T5IP<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 14-12 **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 7-23: 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
_		C1IP<2:0>		_		C1RXIP<2:0>	
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
_		SPI2IP<2:0>		_		SPI2EIP<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 14-12 C1IP<2:0>: ECAN1 Event 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 C1RXIP<2:0>: ECAN1 Receive Data Ready 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 SPI2IP<2:0>: SPI2 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 SPI2EIP<2:0>: SPI2 Error Interrupt Priority bits

111 = Interrupt is priority 7 (highest priority interrupt)

.

.

001 = Interrupt is priority 1

#### REGISTER 7-24: 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
_		IC5IP<2:0>		_		IC4IP<2:0>	
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
_		IC3IP<2:0>		_		DMA3IP<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 14-12 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 Data Transfer Complete Interrupt Priority bits

111 = Interrupt is priority 7 (highest priority interrupt)

•

.

001 = Interrupt is priority 1

#### REGISTER 7-25: IPC10: INTERRUPT PRIORITY CONTROL REGISTER 10

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_		OC7IP<2:0>		_		OC6IP<2:0>	
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
_		OC5IP<2:0>		_		IC6IP<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 14-12 OC7IP<2:0>: Output Compare Channel 7 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 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 7-26: IPC11: INTERRUPT PRIORITY CONTROL REGISTER 11

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_		T6IP<2:0>		_		DMA4IP<2:0>	
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
_	_	_	_	_		OC8IP<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 14-12 T6IP<2:0>: Timer6 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 DMA4IP<2:0>: DMA Channel 4 Data Transfer Complete 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 OC8IP<2:0>: Output Compare Channel 8 Interrupt Priority bits

111 = Interrupt is priority 7 (highest priority interrupt)

•

•

001 = Interrupt is priority 1

#### REGISTER 7-27: IPC12: INTERRUPT PRIORITY CONTROL REGISTER 12

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_		T8IP<2:0>		_		MI2C2IP<2:0>	
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
_		SI2C2IP<2:0>		_		T7IP<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 14-12 T8IP<2:0>: Timer8 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 MI2C2IP<2:0>: I2C2 Master Events 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>: I2C2 Slave Events 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 T7IP<2:0>: Timer7 Interrupt Priority bits

111 = Interrupt is priority 7 (highest priority interrupt)

•

.

001 = Interrupt is priority 1

#### REGISTER 7-28: 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
_		C2RXIP<2:0>		_		INT4IP<2:0>	
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
_		INT3IP<2:0>		_		T9IP<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 14-12 C2RXIP<2:0>: ECAN2 Receive Data Ready 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

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

bit 2-0 **T9IP<2:0>:** Timer9 Interrupt Priority bits

111 = Interrupt is priority 7 (highest priority interrupt)

•

.

•

001 = Interrupt is priority 1

#### REGISTER 7-29: IPC14: INTERRUPT PRIORITY CONTROL REGISTER 14

U-0	U-1	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
_	_	_	_			QEIIP<2:0>	
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
_		PWMIP<2:0>		_		C2IP<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 15-11 **Unimplemented:** Read as '0'

bit 10-8 **QEIIP<2:0>:** QEI Interrupt Priority bits

111 = Interrupt is priority 7 (highest priority interrupt)

•

• 001 — Intornum ti

001 = Interrupt is priority 1

000 = Interrupt source is disabled

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

bit 6-4 **PWMIP<2:0>:** PWM 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 C2IP<2:0>: ECAN2 Event Interrupt Priority bits

111 = Interrupt is priority 7 (highest priority interrupt)

•

•

•

001 = Interrupt is priority 1

#### REGISTER 7-30: IPC15: INTERRUPT PRIORITY CONTROL REGISTER 15

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
_		FLTAIP<2:0>		_	_	_	_
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	U-1	U-0	U-0
_		DMA5IP<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 14-12 FLTAIP<2:0>: PWM Fault A Interrupt Priority bits

111 = Interrupt is priority 7 (highest priority interrupt)

•

•

001 = Interrupt is priority 1

000 = Interrupt source is disabled

bit 11-7 Unimplemented: Read as '0'

bit 6-4 DMA5IP<2:0>: DMA Channel 5 Data Transfer Complete Interrupt Priority bits

111 = Interrupt is priority 7 (highest priority interrupt)

.

.

001 = Interrupt is priority 1

000 = Interrupt source is disabled

bit 3-0 Unimplemented: Read as '0'

#### REGISTER 7-31: IPC16: INTERRUPT PRIORITY CONTROL REGISTER 16

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
_	_	_	_			U2EIP<2:0>	
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
_		U1EIP<2:0>		_		FLTBIP<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 15-11 **Unimplemented:** Read as '0'

bit 10-8 **U2EIP<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 **U1EIP<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

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

bit 2-0 FLTBIP<2:0>: PWM Fault B Interrupt Priority bits

111 = Interrupt is priority 7 (highest priority interrupt)

•

•

001 = Interrupt is priority 1

#### REGISTER 7-32: IPC17: INTERRUPT PRIORITY CONTROL REGISTER 17

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
_		C2TXIP<2:0>		_		C1TXIP<2:0>	
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
_		DMA7IP<2:0>		_		DMA6IP<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 14-12 C2TXIP<2:0>: ECAN2 Transmit Data Request 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 C1TXIP<2:0>: ECAN1 Transmit Data Request 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 DMA7IP<2:0>: DMA Channel 7 Data Transfer Complete 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 DMA6IP<2:0>: DMA Channel 6 Data Transfer Complete Interrupt Priority bits

111 = Interrupt is priority 7 (highest priority interrupt)

•

•

•

001 = Interrupt is priority 1

#### REGISTER 7-33: INTTREG: INTERRUPT CONTROL AND STATUS REGISTER

U-0	U-0	U-0	U-0	R-0	R-0	R-0	R-0
_	_	_	_		ILR<	3:0>	
bit 15							bit 8

U-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
_				VECNUM<6:0	>		
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 15-12 **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 **Unimplemented:** Read as '0'

bit 6-0 **VECNUM<6:0>:** Vector Number of Pending Interrupt bits

0111111 = Interrupt vector pending is number 135

•

.

0000001 = Interrupt vector pending is number 9 0000000 = Interrupt vector pending is number 8

#### 7.4 Interrupt Setup Procedures

#### 7.4.1 INITIALIZATION

To configure an interrupt source, do the following:

- Set the NSTDIS bit (INTCON1<15>) 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.

- 3. Clear the interrupt flag status bit associated with the peripheral in the associated IFSx register.
- Enable the interrupt source by setting the interrupt enable control bit associated with the source in the appropriate IECx register.

#### 7.4.2 INTERRUPT SERVICE ROUTINE

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 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.

#### 7.4.3 TRAP SERVICE ROUTINE

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.

#### 7.4.4 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 OEh with SRL.

To enable user interrupts, the  ${\tt POP}$  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 (level 8-level 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.

NOTES:			

# 8.0 DIRECT MEMORY ACCESS (DMA)

Note 1: This data sheet summarizes the features of the dsPIC33FJXXXMCX06A/X08A/X10A family of devices. However, it is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to Section 22. "Direct Memory Access (DMA)" (DS70182) in the "dsPIC33F/PIC24H Family Reference Manual", which is available from the Microchip web site (www.microchip.com).

2: Some registers and associated bits described in this section may not be available on all devices. Refer to Section 4.0 "Memory Organization" in this data sheet for device-specific register and bit information.

Direct Memory Access (DMA) is a very efficient mechanism of copying data between peripheral SFRs (e.g., the UART Receive register and Input Capture 1 buffer) and buffers or variables stored in RAM, with minimal CPU intervention. The DMA controller can automatically copy entire blocks of data without requiring the user software to read or write the peripheral Special Function Registers (SFRs) every time a peripheral interrupt occurs. The DMA controller uses a dedicated bus for data transfers, and therefore, does not steal cycles from the code execution flow of the CPU. To exploit the DMA capability, the corresponding user buffers or variables must be located in DMA RAM.

The dsPIC33FJXXXMCX06A/X08A/X10A peripherals that can utilize DMA are listed in Table 8-1 along with their associated Interrupt Request (IRQ) numbers.

TABLE 8-1: PERIPHERALS WITH DMA SUPPORT

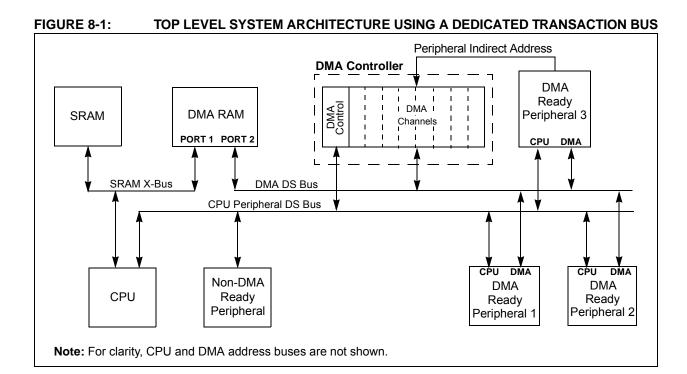
Peripheral	IRQ Number
INT0	0
Input Capture 1	1
Input Capture 2	5
Output Compare 1	2
Output Compare 2	6
Timer2	7
Timer3	8
SPI1	10
SPI2	33
UART1 Reception	11
UART1 Transmission	12
UART2 Reception	30
UART2 Transmission	31
ADC1	13
ADC2	21
ECAN1 Reception	34
ECAN1 Transmission	70
ECAN2 Reception	55
ECAN2 Transmission	71

The DMA controller features eight identical data transfer channels. Each channel has its own set of control and status registers. Each DMA channel can be configured to copy data, either from buffers stored in dual port DMA RAM to peripheral SFRs, or from peripheral SFRs to buffers in DMA RAM.

The DMA controller supports the following features:

- · Word or byte-sized data transfers.
- Transfers from peripheral to DMA RAM or DMA RAM to peripheral.
- Indirect Addressing of DMA RAM locations with or without automatic post-increment.
- Peripheral Indirect Addressing In some peripherals, the DMA RAM read/write addresses may be partially derived from the peripheral.
- One-Shot Block Transfers Terminating DMA transfer after one block transfer.
- Continuous Block Transfers Reloading DMA RAM buffer start address after every block transfer is complete.
- Ping-Pong Mode Switching between two DMA RAM start addresses between successive block transfers, thereby filling two buffers alternately.
- · Automatic or manual initiation of block transfers.
- Each channel can select from 20 possible sources of data sources or destinations.

For each DMA channel, a DMA interrupt request is generated when a block transfer is complete. Alternatively, an interrupt can be generated when half of the block has been filled.



#### 8.1 DMAC Registers

Each DMAC Channel x (x = 0, 1, 2, 3, 4, 5, 6 or 7) contains the following registers:

- A 16-Bit DMA Channel Control register (DMAxCON)
- A 16-Bit DMA Channel IRQ Select register (DMAxREQ)
- A 16-Bit DMA RAM Primary Start Address Offset register (DMAxSTA)

- A 16-Bit DMA RAM Secondary Start Address Offset register (DMAxSTB)
- A 16-Bit DMA Peripheral Address register (DMAxPAD)
- A 10-Bit DMA Transfer Count register (DMAxCNT)

An additional pair of status registers, DMACS0 and DMACS1, are common to all DMAC channels.

#### REGISTER 8-1: DMAxCON: DMA CHANNEL x CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
CHEN	SIZE	DIR	HALF	NULLW	_	_	_
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
_	_	AMODI	E<1:0>	_	_	MODE	<1:0>
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	CHEN: Channel Enable bit
	1 = Channel enabled
	0 = Channel disabled
bit 14	SIZE: Data Transfer Size bit
	1 = Byte
	0 = Word
bit 13	DIR: Transfer Direction bit (source/destination bus select)
	1 = Read from DMA RAM address; write to peripheral address
	0 = Read from peripheral address; write to DMA RAM address
bit 12	HALF: Early Block Transfer Complete Interrupt Select bit
	1 = Initiate block transfer complete interrupt when half of the data has been moved
	0 = Initiate block transfer complete interrupt when all of the data has been moved
bit 11	NULLW: Null Data Peripheral Write Mode Select bit
	1 = Null data write to peripheral in addition to DMA RAM write (DIR bit must also be clear)
	0 = Normal operation
bit 10-6	Unimplemented: Read as '0'
bit 5-4	AMODE<1:0>: DMA Channel Operating Mode Select bits
	11 = Reserved
	10 = Peripheral Indirect Addressing mode
	01 = Register Indirect without Post-Increment mode
	00 = Register Indirect with Post-Increment mode
bit 3-2	00 = Register Indirect with Post-Increment mode  Unimplemented: Read as '0'
bit 3-2 bit 1-0	00 = Register Indirect with Post-Increment mode

10 = Continuous, Ping-Pong modes enabled 01 = One-Shot, Ping-Pong modes disabled 00 = Continuous, Ping-Pong modes disabled

#### REGISTER 8-2: DMAXREQ: DMA CHANNEL x IRQ SELECT REGISTER

R/W-0	U-0						
FORCE <sup>(1)</sup>	_	_	_	_	_	_	_
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
_	IRQSEL6 <sup>(2)</sup>	IRQSEL5 <sup>(2)</sup>	IRQSEL4 <sup>(2)</sup>	IRQSEL3 <sup>(2)</sup>	IRQSEL2 <sup>(2)</sup>	IRQSEL1 <sup>(2)</sup>	IRQSEL0 <sup>(2)</sup>
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 **FORCE:** Force DMA Transfer bit<sup>(1)</sup>

1 = Force a single DMA transfer (Manual mode)

0 = Automatic DMA transfer initiation by DMA request

bit 14-7 **Unimplemented:** Read as '0'

bit 6-0 IRQSEL<6:0>: DMA Peripheral IRQ Number Select bits<sup>(2)</sup>

000000-1111111 = DMAIRQ0-DMAIRQ127 selected to be Channel DMAREQ

**Note 1:** The FORCE bit cannot be cleared by the user. The FORCE bit is cleared by hardware when the forced DMA transfer is complete.

2: See Table 8-1 for a complete listing of IRQ numbers for all interrupt sources.

#### REGISTER 8-3: DMAXSTA: DMA CHANNEL x RAM START ADDRESS OFFSET REGISTER A

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0				
	STA<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	R/W-0			
STA<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 15-0 **STA<15:0>:** Primary DMA RAM Start Address bits (source or destination)

#### REGISTER 8-4: DMAxSTB: DMA CHANNEL x RAM START ADDRESS OFFSET REGISTER B

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0		
STB<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	R/W-0			
STB<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 15-0 STB<15:0>: Secondary DMA RAM Start Address bits (source or destination)

#### REGISTER 8-5: DMAXPAD: DMA CHANNEL x PERIPHERAL ADDRESS REGISTER<sup>(1)</sup>

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0			
	PAD<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	R/W-0			
PAD<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 15-0 PAD<15:0>: Peripheral Address Register bits

**Note 1:** If the channel is enabled (i.e., active), writes to this register may result in unpredictable behavior of the DMA channel and should be avoided.

#### REGISTER 8-6: DMAxCNT: DMA CHANNEL x TRANSFER COUNT REGISTER<sup>(1)</sup>

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
_	_	_	_	_	_	CNT<9:8>(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				
	CNT<7:0> <sup>(2)</sup>										
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 15-10 **Unimplemented:** Read as '0'

bit 9-0 CNT<9:0>: DMA Transfer Count Register bits<sup>(2)</sup>

**Note 1:** If the channel is enabled (i.e., active), writes to this register may result in unpredictable behavior of the DMA channel and should be avoided.

2: Number of DMA transfers = CNT<9:0> + 1.

#### REGISTER 8-7: DMACS0: DMA CONTROLLER STATUS REGISTER 0

| R/C-0  |
|--------|--------|--------|--------|--------|--------|--------|--------|
| PWCOL7 | PWCOL6 | PWCOL5 | PWCOL4 | PWCOL3 | PWCOL2 | PWCOL1 | PWCOL0 |
| bit 15 |        |        |        |        |        |        | bit 8  |

| R/C-0  |
|--------|--------|--------|--------|--------|--------|--------|--------|
| XWCOL7 | XWCOL6 | XWCOL5 | XWCOL4 | XWCOL3 | XWCOL2 | XWCOL1 | XWCOL0 |
| bit 7  |        |        |        |        |        |        | bit 0  |

Legend:		C = Clearable bit				
R = Readable bit	W = Writable 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	PWCOL7: Channel 7 Peripheral Write Collision Flag bit
	1 = Write collision detected
	0 = No write collision detected
bit 14	PWCOL6: Channel 6 Peripheral Write Collision Flag bit
	1 = Write collision detected
	0 = No write collision detected
bit 13	PWCOL5: Channel 5 Peripheral Write Collision Flag bit
	1 = Write collision detected
	0 = No write collision detected
bit 12	PWCOL4: Channel 4 Peripheral Write Collision Flag bit
	1 = Write collision detected
	0 = No write collision detected
bit 11	PWCOL3: Channel 3 Peripheral Write Collision Flag bit
	1 = Write collision detected
1 11 40	0 = No write collision detected
bit 10	PWCOL2: Channel 2 Peripheral Write Collision Flag bit
	1 = Write collision detected 0 = No write collision detected
h:4 O	
bit 9	PWCOL1: Channel 1 Peripheral Write Collision Flag bit  1 = Write collision detected
	0 = No write collision detected
bit 8	PWCOL0: Channel 0 Peripheral Write Collision Flag bit
Dit 0	1 = Write collision detected
	0 = No write collision detected
bit 7	XWCOL7: Channel 7 DMA RAM Write Collision Flag bit
	1 = Write collision detected
	0 = No write collision detected
bit 6	XWCOL6: Channel 6 DMA RAM Write Collision Flag bit
	1 = Write collision detected
	0 = No write collision detected
bit 5	XWCOL5: Channel 5 DMA RAM Write Collision Flag bit
	1 = Write collision detected
	0 = No write collision detected
bit 4	XWCOL4: Channel 4 DMA RAM Write Collision Flag bit
	1 = Write collision detected
	0 = No write collision detected

#### REGISTER 8-7: DMACS0: DMA CONTROLLER STATUS REGISTER 0 (CONTINUED)

bit 3 XWCOL3: Channel 3 DMA RAM Write Collision Flag bit

1 = Write collision detected0 = No write collision detected

bit 2 XWCOL2: Channel 2 DMA RAM Write Collision Flag bit

1 = Write collision detected0 = No write collision detected

bit 1 XWCOL1: Channel 1 DMA RAM Write Collision Flag bit

1 = Write collision detected0 = No write collision detected

bit 0 XWCOL0: Channel 0 DMA RAM Write Collision Flag bit

1 = Write collision detected0 = No write collision detected

#### REGISTER 8-8: DMACS1: DMA CONTROLLER STATUS REGISTER 1

U-0	U-0	U-0	U-0	R-1	R-1	R-1	R-1	
_	_	_	_	LSTCH<3:0>				
bit 15							bit 8	

| R-0   |
|-------|-------|-------|-------|-------|-------|-------|-------|
| PPST7 | PPST6 | PPST5 | PPST4 | PPST3 | PPST2 | PPST1 | PPST0 |
| 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 11-8 LSTCH<3:0>: Last DMA Channel Active bits

1111 = No DMA transfer has occurred since system Reset

1110-1000 = Reserved

0111 = Last data transfer was by DMA Channel 7 0110 = Last data transfer was by DMA Channel 6

0101 = Last data transfer was by DMA Channel 5 0100 = Last data transfer was by DMA Channel 4

0011 = Last data transfer was by DMA Channel 3 0010 = Last data transfer was by DMA Channel 2

0001 = Last data transfer was by DMA Channel 1 0000 = Last data transfer was by DMA Channel 0

bit 7 PPST7: Channel 7 Ping-Pong Mode Status Flag bit

1 = DMA7STB register selected 0 = DMA7STA register selected

bit 6 PPST6: Channel 6 Ping-Pong Mode Status Flag bit

1 = DMA6STB register selected 0 = DMA6STA register selected

bit 5 **PPST5:** Channel 5 Ping-Pong Mode Status Flag bit

1 = DMA5STB register selected0 = DMA5STA register selected

bit 4 PPST4: Channel 4 Ping-Pong Mode Status Flag bit

1 = DMA4STB register selected 0 = DMA4STA register selected

bit 3 PPST3: Channel 3 Ping-Pong Mode Status Flag bit

1 = DMA3STB register selected 0 = DMA3STA register selected

bit 2 PPST2: Channel 2 Ping-Pong Mode Status Flag bit

1 = DMA2STB register selected 0 = DMA2STA register selected

bit 1 PPST1: Channel 1 Ping-Pong Mode Status Flag bit

1 = DMA1STB register selected 0 = DMA1STA register selected

bit 0 PPST0: Channel 0 Ping-Pong Mode Status Flag bit

1 = DMA0STB register selected 0 = DMA0STA register selected

#### REGISTER 8-9: DSADR: MOST RECENT DMA RAM ADDRESS

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	
DSADR<15:8>								
bit 15 bit 8								

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	
DSADR<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 15-0 DSADR<15:0>: Most Recent DMA RAM Address Accessed by DMA Controller bits

# 9.0 OSCILLATOR CONFIGURATION

Note 1: This data sheet summarizes the features of the dsPIC33FJXXXMCX06A/X08A/X10A family of devices. However, it is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to Section 7. "Oscillator" (DS70186) in the "dsPIC33F/PIC24H Family Reference Manual", which is available from the Microchip web site (www.microchip.com).

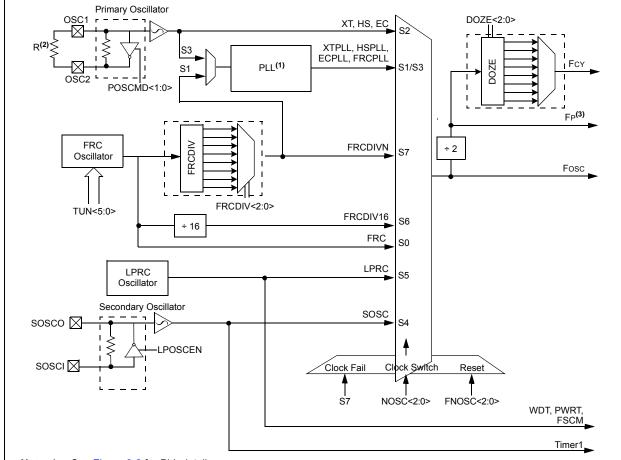
2: Some registers and associated bits described in this section may not be available on all devices. Refer to Section 4.0 "Memory Organization" in this data sheet for device-specific register and bit information.

The dsPIC33FJXXXMCX06A/X08A/X10A oscillator system provides the following:

- Various external and internal oscillator options as clock sources
- An on-chip PLL to scale the internal operating frequency to the required system clock frequency
- The internal FRC oscillator can also be used with the PLL, thereby allowing full-speed operation without any external clock generation hardware
- Clock switching between various clock sources
- Programmable clock postscaler for system power savings
- A Fail-Safe Clock Monitor (FSCM) that detects clock failure and takes fail-safe measures
- A Clock Control register (OSCCON)
- Nonvolatile Configuration bits for main oscillator selection

A simplified diagram of the oscillator system is shown in Figure 9-1.

#### FIGURE 9-1: dsPIC33FJXXXMCX06A/X08A/X10A OSCILLATOR SYSTEM DIAGRAM



- Note 1: See Figure 9-2 for PLL details.
  - 2: If the Oscillator is used with XT or HS modes, an extended parallel resistor with the value of 1  $M\Omega$  must be connected.
  - 3: The term, FP refers to the clock source for all the peripherals, while FcY refers to the clock source for the CPU. Throughout this document FP and FcY are used interchangeably, except in the case of Doze mode. FP and FcY will be different when Doze mode is used in any ratio other than 1:1, which is the default.

#### 9.1 CPU Clocking System

There are seven system clock options provided by the dsPIC33FJXXXMCX06A/X08A/X10A:

- FRC Oscillator
- · FRC Oscillator with PLL
- · Primary (XT, HS or EC) Oscillator
- · Primary Oscillator with PLL
- · Secondary (LP) Oscillator
- · LPRC Oscillator
- · FRC Oscillator with Postscaler

#### 9.1.1 SYSTEM CLOCK SOURCES

The FRC (Fast RC) internal oscillator runs at a nominal frequency of 7.37 MHz. The user software can tune the FRC frequency. User software can optionally specify a factor (ranging from 1:2 to 1:256) by which the FRC clock frequency is divided. This factor is selected using the FRCDIV<2:0> bits (CLKDIV<10:8>).

The primary oscillator can use one of the following as its clock source:

- XT (Crystal): Crystals and ceramic resonators in the range of 3 MHz to 10 MHz. The crystal is connected to the OSC1 and OSC2 pins.
- HS (High-Speed Crystal): Crystals in the range of 10 MHz to 40 MHz. The crystal is connected to the OSC1 and OSC2 pins.
- 3. EC (External Clock): External clock signal is directly applied to the OSC1 pin.

The secondary (LP) oscillator is designed for low power and uses a 32.768 kHz crystal or ceramic resonator. The LP oscillator uses the SOSCI and SOSCO pins.

The LPRC (Low-Power RC) internal oscillator runs at a nominal frequency of 32.768 kHz. It is also used as a reference clock by the Watchdog Timer (WDT) and Fail-Safe Clock Monitor (FSCM).

The clock signals generated by the FRC and primary oscillators can be optionally applied to an on-chip Phase-Locked Loop (PLL) to provide a wide range of output frequencies for device operation. PLL configuration is described in **Section 9.1.3 "PLL Configuration"**.

The FRC frequency depends on the FRC accuracy (see Table 26-19) and the value of the FRC Oscillator Tuning register (see Register 9-4).

#### 9.1.2 SYSTEM CLOCK SELECTION

The oscillator source 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 the program memory. (Refer to **Section 23.1 "Configuration Bits"** for further details.) The Initial Oscillator Selection Configuration bits, FNOSC<2:0> (FOSCSEL<2:0>), and the Primary Oscillator Mode Select Configuration bits,

POSCMD<1:0> (FOSC<1:0>), select the oscillator source that is used at a Power-on Reset. The FRC primary oscillator is the default (unprogrammed) selection.

The Configuration bits allow users to choose between twelve different clock modes, shown in Table 9-1.

The output of the oscillator (or the output of the PLL if a PLL mode has been selected), Fosc, is divided by 2 to generate the device instruction clock (FcY) and the peripheral clock time base (FP). FcY defines the operating speed of the device and speeds up to 40 MHz are supported by the dsPIC33FJXXXMCX06A/X08A/X10A architecture.

Instruction execution speed or device operating frequency, Fcy, is given by the following equation:

## EQUATION 9-1: DEVICE OPERATING FREQUENCY

$$FCY = \frac{FOSC}{2}$$

#### 9.1.3 PLL CONFIGURATION

The primary oscillator and internal FRC oscillator can optionally use an on-chip PLL to obtain higher speeds of operation. The PLL provides a significant amount of flexibility in selecting the device operating speed. A block diagram of the PLL is shown in Figure 9-2.

The output of the primary oscillator or FRC, denoted as 'FIN', is divided down by a prescale factor (N1) of 2, 3, ... or 33 before being provided to the PLL's Voltage Controlled Oscillator (VCO). The input to the VCO must be selected to be in the range of 0.8 MHz to 8 MHz. Since the minimum prescale factor is 2, this implies that FIN must be chosen to be in the range of 1.6 MHz to 16 MHz. The prescale factor, 'N1', is selected using the PLLPRE<4:0> bits (CLKDIV<4:0>).

The PLL feedback divisor, selected using the PLLDIV<8:0> bits (PLLFBD<8:0>), provides a factor, 'M', by which the input to the VCO is multiplied. This factor must be selected such that the resulting VCO output frequency is in the range of 100 MHz to 200 MHz.

The VCO output is further divided by a postscale factor, 'N2'. This factor is selected using the PLLPOST<1:0> bits (CLKDIV<7:6>). 'N2' can be either 2, 4 or 8, and must be selected such that the PLL output frequency (Fosc) is in the range of 12.5 MHz to 80 MHz, which generates device operating speeds of 6.25-40 MIPS.

For a primary oscillator or FRC oscillator output, 'FIN', the PLL output, 'FOSC', is given by the following equation:

#### **EQUATION 9-2:** Fosc CALCULATION

$$Fosc = Fin \cdot \left(\frac{M}{N1 \cdot N2}\right)$$

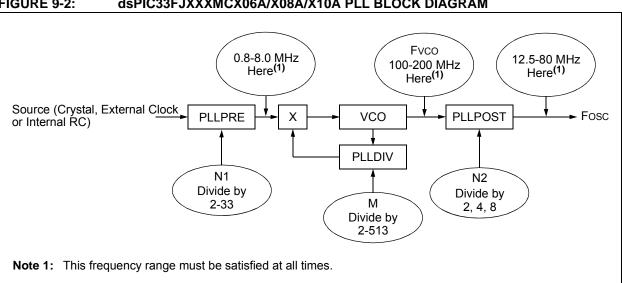
For example, suppose a 10 MHz crystal is being used with "XT with PLL" as the selected oscillator mode. If PLLPRE<4:0> = 0, then N1 = 2. This yields a VCO input of 10/2 = 5 MHz, which is within the acceptable range of 0.8-8 MHz. If PLLDIV<8:0> = 0x1E, then M = 32. This yields a VCO output of 5 \* 32 = 160 MHz, which is within the 100-200 MHz ranged needed.

If PLLPOST<1:0> = 0, then N2 = 2. This provides a Fosc of 160/2 = 80 MHz. The resultant device operating speed is 80/2 = 40 MIPS.

#### **XT WITH PLL MODE EQUATION 9-3: EXAMPLE**

$$FCY = \frac{FOSC}{2} = \frac{1}{2} \left( \frac{10000000 \cdot 32}{2 \cdot 2} \right) = 40 \text{ MIPS}$$

#### FIGURE 9-2: dsPIC33FJXXXMCX06A/X08A/X10A PLL BLOCK DIAGRAM



**TABLE 9-1:** CONFIGURATION BIT VALUES FOR CLOCK SELECTION

Oscillator Mode	Oscillator Source	POSCMD<1:0>	FNOSC<2:0>	See Note
Fast RC Oscillator with Divide-by-N (FRCDIVN)	Internal	xx	111	1, 2
Fast RC Oscillator with Divide-by-16 (FRCDIV16)	Internal	xx	110	1
Low-Power RC Oscillator (LPRC)	Internal	xx	101	1
Secondary (Timer1) Oscillator (Sosc)	Secondary	xx	100	1
Primary Oscillator (HS) with PLL (HSPLL)	Primary	10	011	_
Primary Oscillator (XT) with PLL (XTPLL)	Primary	01	011	_
Primary Oscillator (EC) with PLL (ECPLL)	Primary	0.0	011	1
Primary Oscillator (HS)	Primary	10	010	_
Primary Oscillator (XT)	Primary	01	010	_
Primary Oscillator (EC)	Primary	00	010	1
Fast RC Oscillator with PLL (FRCPLL)	Internal	xx	001	1
Fast RC Oscillator (FRC)	Internal	xx	000	1

Note 1: OSC2 pin function is determined by the OSCIOFNC Configuration bit.

This is the default oscillator mode for an unprogrammed (erased) device.

# REGISTER 9-1: OSCCON: OSCILLATOR CONTROL REGISTER (1,3)

U-0	R-0	R-0	R-0	U-0	R/W-y	R/W-y	R/W-y	
_	COSC<2:0>			_	NOSC<2:0> <sup>(2)</sup>			
bit 15							bit 8	

R/W-0	U-0	R-0	U-0	R/C-0	U-0	R/W-0	R/W-0
CLKLOCK	_	LOCK	_	CF	_	LPOSCEN	OSWEN
bit 7							bit 0

Legend:	y = Value set from Configuration bits on POR					
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 **Unimplemented:** Read as '0'

bit 14-12 COSC<2:0>: Current Oscillator Selection bits (read-only)

111 = Fast RC oscillator (FRC) with Divide-by-N

110 = Fast RC oscillator (FRC) with Divide-by-16

101 = Low-Power RC oscillator (LPRC)

100 = Secondary oscillator (Sosc)

011 = Primary oscillator (XT, HS, EC) with PLL

010 = Primary oscillator (XT, HS, EC)

001 = Fast RC Oscillator (FRC) with Divide-by-N and PLL (FRCDIVN + PLL)

000 = Fast RC oscillator (FRC)

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

bit 10-8 NOSC<2:0>: New Oscillator Selection bits<sup>(2)</sup>

111 = Fast RC oscillator (FRC) with Divide-by-N

110 = Fast RC oscillator (FRC) with Divide-by-16

101 = Low-Power RC oscillator (LPRC)

100 = Secondary oscillator (Sosc)

011 = Primary oscillator (XT, HS, EC) with PLL

010 = Primary oscillator (XT, HS, EC)

001 = Fast RC Oscillator (FRC) with Divide-by-N and PLL (FRCDIVN + PLL)

000 = Fast RC oscillator (FRC)

bit 7 CLKLOCK: Clock Lock Enable bit

1 = If (FCKSM0 = 1), then clock and PLL configurations are locked. If (FCKSM0 = 0), then clock and PLL configurations may be modified.

0 = Clock and PLL selections are not locked; configurations may be modified

bit 6 Unimplemented: Read as '0'

bit 5 LOCK: PLL Lock Status bit (read-only)

1 = Indicates that PLL is in lock or PLL start-up timer is satisfied

0 = Indicates that PLL is out of lock, start-up timer is in progress or PLL is disabled

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

bit 3 **CF:** Clock Fail Detect bit (read/clear by application)

1 = FSCM has detected clock failure

0 = FSCM has not detected clock failure

Note 1: Writes to this register require an unlock sequence. Refer to **Section 7. "Oscillator"** (DS70186) in the "dsPIC33F/PIC24H Family Reference Manual" for details.

- 2: Direct clock switches between any primary oscillator mode with PLL and FRCPLL modes are not permitted. This applies to clock switches in either direction. In these instances, the application must switch to FRC mode as a transition clock source between the two PLL modes.
- 3: This register is reset only on a Power-on Reset (POR).

# REGISTER 9-1: OSCCON: OSCILLATOR CONTROL REGISTER<sup>(1,3)</sup> (CONTINUED)

bit 2 Unimplemented: Read as '0'

bit 1 LPOSCEN: Secondary (LP) Oscillator Enable bit

1 = Enable secondary oscillator0 = Disable secondary oscillator

bit 0 **OSWEN:** Oscillator Switch Enable bit

1 = Request oscillator switch to selection specified by NOSC<2:0> bits

0 = Oscillator switch is complete

**Note 1:** Writes to this register require an unlock sequence. Refer to **Section 7. "Oscillator"** (DS70186) in the "dsPIC33F/PIC24H Family Reference Manual" for details.

- 2: Direct clock switches between any primary oscillator mode with PLL and FRCPLL modes are not permitted. This applies to clock switches in either direction. In these instances, the application must switch to FRC mode as a transition clock source between the two PLL modes.
- 3: This register is reset only on a Power-on Reset (POR).

# REGISTER 9-2: CLKDIV: CLOCK DIVISOR REGISTER(2)

R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0
ROI	DOZE<2:0>		DOZEN <sup>(1)</sup>		FRCDIV<2:0>		
bit 15							bit 8

R/W-0	R/W-1	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	
PLLPOS	ST<1:0>	_	PLLPRE<4:0>					
bit 7			bit					

Legend:	y = Value set from Configuration bits on POR				
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 ROI: Recover on Interrupt bit

1 = Interrupts will clear the DOZEN bit and the processor clock/peripheral clock ratio is set to 1:1

0 = Interrupts have no effect on the DOZEN bit

bit 14-12 **DOZE<2:0>:** Processor Clock Reduction Select bits

000 = Fcy/1

001 = Fcy/2

010 = Fcy/4

011 = Fcy/8 (default)

100 = Fcy/16

101 = Fcy/32

110 = Fcy/64

111 = Fcy/128

bit 11 **DOZEN:** DOZE Mode Enable bit<sup>(1)</sup>

1 = DOZE<2:0> field specifies the ratio between the peripheral clocks and the processor clocks

0 = Processor clock/peripheral clock ratio forced to 1:1

bit 10-8 FRCDIV<2:0>: Internal Fast RC Oscillator Postscaler bits

000 = FRC divide by 1 (default)

001 = FRC divide by 2

010 = FRC divide by 4

011 = FRC divide by 8

100 = FRC divide by 16

101 = FRC divide by 32

110 = FRC divide by 64

111 = FRC divide by 256

bit 7-6 PLLPOST<1:0>: PLL VCO Output Divider Select bits (also denoted as 'N2', PLL postscaler)

00 = Output/2

01 = Output/4 (default)

10 = Reserved

11 = Output/8

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

bit 4-0 PLLPRE<4:0>: PLL Phase Detector Input Divider bits (also denoted as 'N1', PLL prescaler)

00000 = Input/2 (default)

00001 = Input/3

•

•

11111 = Input/33

Note 1: This bit is cleared when the ROI bit is set and an interrupt occurs.

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

# REGISTER 9-3: PLLFBD: PLL FEEDBACK DIVISOR REGISTER<sup>(1)</sup>

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0		
_	_	_	_	_	_	_	PLLDIV<8>		
bit 15									

R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0	
PLLDIV<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 15-9 **Unimplemented:** Read as '0'

bit 8-0 PLLDIV<8:0>: PLL Feedback Divisor bits (also denoted as 'M', PLL multiplier)

0000000000 = 2 000000001 = 3000000010 = 4

•

•

•

000110000 = **50** (default)

•

•

•

111111111 = 513

Note 1: This register is reset only on a Power-on Reset (POR).

# REGISTER 9-4: OSCTUN: FRC OSCILLATOR TUNING REGISTER<sup>(2)</sup>

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		
_	_		TUN<5:0> <sup>(1)</sup>						
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 **TUN<5:0>:** FRC Oscillator Tuning bits<sup>(1)</sup>

011111 = Center frequency + 11.625% (8.23 MHz)

011110 = Center frequency + 11.25% (8.20 MHz)

•

•

000001 = Center frequency + 0.375% (7.40 MHz)

000000 = Center frequency (7.37 MHz nominal)

111111 = Center frequency – 0.375% (7.345 MHz)

•

•

.

100001 = Center frequency - 11.625% (6.52 MHz)

100000 = Center frequency – 12% (6.49 MHz)

**Note 1:** OSCTUN functionality has been provided to help customers compensate for temperature effects on the FRC frequency over a wide range of temperatures. The tuning step size is an approximation and is neither characterized nor tested.

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

# 9.2 Clock Switching Operation

Applications are free to switch between any of the four clock sources (Primary, LP, FRC and LPRC) under software control at any time. To limit the possible side effects that could result from this flexibility, dsPIC33FJXXXMCX06A/X08A/X10A devices have a safeguard lock built into the switch process.

Note: Primary Oscillator mode has three different submodes (XT, HS and EC) which are determined by the POSCMD<1:0> 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.

#### 9.2.1 ENABLING CLOCK SWITCHING

To enable clock switching, the FCKSM1 Configuration bit in the Configuration register must be programmed to '0'. (Refer to **Section 23.1 "Configuration Bits"** for further details.) 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 NOSC control bits (OSCCON<10:8>) do not control the clock selection when clock switching is disabled. However, the COSC bits (OSCCON<14:12>) reflect the clock source selected by the FNOSC Configuration bits.

The OSWEN control bit (OSCCON<0>) has no effect when clock switching is disabled; it is held at '0' at all times.

#### 9.2.2 OSCILLATOR SWITCHING SEQUENCE

At a minimum, performing a clock switch requires the following basic sequence:

- If desired, read the COSC 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 NOSC control 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 COSC status bits with the new value of the NOSC control 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 the CF (OSCCON<3>) status 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 waits until the Oscillator Start-up Timer (OST) expires. If the new source is using the PLL, 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 NOSC bit values are transferred to the COSC status bits.
- The old clock source is turned off at this time, with the exception of LPRC (if WDT or FSCM is enabled) or LP (if LPOSCEN remains set).
  - Note 1: The processor continues 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 transition clock source between the two PLL modes.
    - 3: Refer to **Section 7.** "Oscillator" (DS70186) in the "dsPIC33F/PIC24H Family Reference Manual" for details.

### 9.3 Fail-Safe Clock Monitor (FSCM)

The Fail-Safe Clock Monitor (FSCM) allows the device to continue to operate even in the event of an oscillator failure. The FSCM function is enabled by programming. If the FSCM function is enabled, the LPRC internal oscillator runs at all times (except during Sleep mode) and is not subject to control by the Watchdog Timer.

In the event of an oscillator failure, the FSCM generates a clock failure trap event and switches the system clock over to the FRC oscillator. Then, the application program can either attempt to restart the oscillator or execute a controlled shutdown. The trap can be treated as a warm Reset by simply loading the Reset address into the oscillator fail trap vector.

If the PLL multiplier is used to scale the system clock, the internal FRC is also multiplied by the same factor on clock failure. Essentially, the device switches to FRC with PLL on a clock failure.

dsPIC3	dsPIC33FJXXXMCX06A/X08A/X10A							
NOTES:								

#### 10.0 POWER-SAVING FEATURES

# Note 1: This data sheet summarizes the features of the dsPIC33FJXXXMCX06A/X08A/X10A family of devices. However, it is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to Section 9. "Watchdog Timer and Power-Saving Modes" (DS70196) in the "dsPIC33F/PIC24H Family Reference Manual", which is available from the Microchip web site (www.microchip.com).

2: Some registers and associated bits described in this section may not be available on all devices. Refer to Section 4.0 "Memory Organization" in this data sheet for device-specific register and bit information.

The dsPIC33FJXXXMCX06A/X08A/X10A devices provide 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 constitutes lower consumed power. dsPIC33FJXXXMCX06A/X08A/X10A devices can manage power consumption in four different ways:

- · Clock frequency
- · Instruction-based Sleep and Idle modes
- · 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 Clock Frequency and Clock Switching

dsPIC33FJXXXMCX06A/X08A/X10A devices allow 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 NOSC bits (OSCCON<10:8>). 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".

#### 10.2 Instruction-Based Power-Saving Modes

dsPIC33FJXXXMCX06A/X08A/X10A devices have two special power-saving modes that are entered through the execution of a special 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. The assembly syntax of the PWRSAV instruction is shown in Example 10-1.

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".

#### 10.2.1 SLEEP MODE

Sleep mode has the following features:

- The system clock source is shut down. If an on-chip oscillator is used, it is turned off.
- The device current consumption is reduced to a minimum, provided that no I/O pin is sourcing current
- The Fail-Safe Clock Monitor does not operate during Sleep mode since the system clock source is disabled.
- The LPRC clock continues to run in Sleep mode if the WDT 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 and peripherals that use an external clock input. Any peripheral that requires the system clock source for its operation is disabled in Sleep mode.

The device will wake-up from Sleep mode on any of the following events:

- · Any interrupt source that is individually enabled
- · Any form of device Reset
- A WDT time-out

On wake-up from Sleep, the processor restarts with the same clock source that was active when Sleep mode was entered.

#### EXAMPLE 10-1: PWRSAV INSTRUCTION SYNTAX

PWRSAV #SLEEP\_MODE ; Put the device into SLEEP mode PWRSAV #IDLE\_MODE ; Put the device into IDLE mode

#### 10.2.2 IDLE MODE

Idle mode has the following features:

- · The CPU stops 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 10.4 "Peripheral Module Disable").
- If the WDT or FSCM is enabled, the LPRC also remains active.

The device will wake from Idle mode on any of the following 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 will begin (2-4 clock cycles later), starting with the instruction following the PWRSAV instruction or the first instruction in the ISR.

# 10.2.3 INTERRUPTS COINCIDENT WITH POWER SAVE INSTRUCTIONS

Any interrupt that coincides with the execution of a PWRSAV instruction is held off until entry into Sleep or Idle mode has completed. The device then wakes up from Sleep or Idle mode.

#### 10.3 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 setting.

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.

For example, suppose the device is operating at 20 MIPS and the CAN module has been configured for 500 kbps based on this device operating speed. If the device is now placed in Doze mode with a clock frequency ratio of 1:4, the CAN module continues to communicate at the required bit rate of 500 kbps, but the CPU now starts executing instructions at a frequency of 5 MIPS.

#### 10.4 Peripheral Module Disable

The Peripheral Module Disable registers (PMD) provide a method to disable a peripheral module by stopping all clock sources supplied to that module. When a peripheral is disabled via the appropriate PMD control bit, the peripheral is in a minimum power consumption state. The control and status registers associated with the peripheral are also disabled, so writes to those registers will have no effect and read values will be invalid.

A peripheral module is only enabled if both the associated bit in the PMD register is cleared and the peripheral is supported by the specific dsPIC<sup>®</sup> DSC variant. If the peripheral is present in the device, it is enabled in the PMD register by default.

Note:

If a PMD bit is set, the corresponding module is disabled after a delay of 1 instruction cycle. Similarly, if a PMD bit is cleared, the corresponding module is enabled after a delay of 1 instruction cycle (assuming the module control registers are already configured to enable module operation).

#### REGISTER 10-1: PMD1: PERIPHERAL MODULE DISABLE CONTROL REGISTER 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0
T5MD	T4MD	T3MD	T2MD	T1MD	QEI1MD	PWMMD	_
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
I2C1MD	U2MD	U1MD	SPI2MD	SPI1MD	C2MD	C1MD	AD1MD <sup>(1)</sup>
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	<b>T5MD:</b> Timer5 Module Disable bit
	1 = Timer5 module is disabled
	0 = Timer5 module is enabled
bit 14	<b>T4MD:</b> Timer4 Module Disable bit
	1 = Timer4 module is disabled
	0 = Timer4 module is enabled
bit 13	<b>T3MD:</b> Timer3 Module Disable bit
	1 = Timer3 module is disabled 0 = Timer3 module is enabled
bit 12	<b>T2MD:</b> Timer2 Module Disable bit
DIL 12	1 = Timer2 module is disabled
	0 = Timer2 module is enabled
bit 11	T1MD: Timer1 Module Disable bit
	1 = Timer1 module is disabled
	0 = Timer1 module is enabled
bit 10	<b>QEI1MD:</b> QEI1 Module Disable bit
	1 = QEI1 module is disabled
	0 = QEI1 module is enabled
bit 9	<b>PWMMD:</b> PWM Module Disable bit
	1 = PWM module is disabled 0 = PWM module is enabled
bit 8	Unimplemented: Read as '0'
bit 7	<b>I2C1MD:</b> I2C1 Module Disable bit
DIL 7	1 = I2C1 module is disabled
	0 = I2C1 module is disabled
bit 6	U2MD: UART2 Module Disable bit
	1 = UART2 module is disabled
	0 = UART2 module is enabled
bit 5	U1MD: UART1 Module Disable bit
	1 = UART1 module is disabled
	0 = UART1 module is enabled
bit 4	SPI2MD: SPI2 Module Disable bit
	1 = SPI2 module is disabled 0 = SPI2 module is enabled

**Note 1:** The PCFGx bits have no effect if the ADC module is disabled by setting this bit. In this case, all port pins multiplexed with ANx will be in Digital mode.

#### REGISTER 10-1: PMD1: PERIPHERAL MODULE DISABLE CONTROL REGISTER 1 (CONTINUED)

bit 3

SPI1MD: SPI1 Module Disable bit

1 = SPI1 module is disabled
0 = SPI1 module is enabled

bit 2

C2MD: ECAN2 Module Disable bit

1 = ECAN2 module is disabled
0 = ECAN2 module is enabled

bit 1

C1MD: ECAN1 Module Disable bit
1 = ECAN1 module is disabled

bit 0 AD1MD: ADC1 Module Disable bit<sup>(1)</sup>

1 = ADC1 module is disabled 0 = ADC1 module is enabled

0 = ECAN1 module is enabled

**Note 1:** The PCFGx bits have no effect if the ADC module is disabled by setting this bit. In this case, all port pins multiplexed with ANx will be in Digital mode.

# REGISTER 10-2: PMD2: PERIPHERAL MODULE DISABLE 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
IC8MD	IC7MD	IC6MD	IC5MD	IC4MD	IC3MD	IC2MD	IC1MD
bit 15							bit 8

| R/W-0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| OC8MD | OC7MD | OC6MD | OC5MD | OC4MD | OC3MD | OC2MD | OC1MD |
| 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	IC8MD: Input Capture 8 Module Disable bit
	<ul><li>1 = Input Capture 8 module is disabled</li><li>0 = Input Capture 8 module is enabled</li></ul>
bit 14	IC7MD: Input Capture 7 Module Disable bit
	1 = Input Capture 7 module is disabled
	0 = Input Capture 7 module is enabled
bit 13	IC6MD: Input Capture 6 Module Disable bit
	<ul><li>1 = Input Capture 6 module is disabled</li><li>0 = Input Capture 6 module is enabled</li></ul>
bit 12	IC5MD: Input Capture 5 Module Disable bit
DIL 12	1 = Input Capture 5 module is disabled
	0 = Input Capture 5 module is enabled
bit 11	IC4MD: Input Capture 4 Module Disable bit
	1 = Input Capture 4 module is disabled
	0 = Input Capture 4 module is enabled
bit 10	IC3MD: Input Capture 3 Module Disable bit
	1 = Input Capture 3 module is disabled
	0 = Input Capture 3 module is enabled
bit 9	IC2MD: Input Capture 2 Module Disable bit
	1 = Input Capture 2 module is disabled
	0 = Input Capture 2 module is enabled
bit 8	IC1MD: Input Capture 1 Module Disable bit
	1 = Input Capture 1 module is disabled
	0 = Input Capture 1 module is enabled
bit 7	OC8MD: Output Compare 8 Module Disable bit
	1 = Output Compare 8 module is disabled
	0 = Output Compare 8 module is enabled
bit 6	OC7MD: Output Compare 4 Module Disable bit
	1 = Output Compare 7 module is disabled
	0 = Output Compare 7 module is enabled
bit 5	OC6MD: Output Compare 6 Module Disable bit
	1 = Output Compare 6 module is disabled
	0 = Output Compare 6 module is enabled
bit 4	OC5MD: Output Compare 5 Module Disable bit
	<ul><li>1 = Output Compare 5 module is disabled</li><li>0 = Output Compare 5 module is enabled</li></ul>

# REGISTER 10-2: PMD2: PERIPHERAL MODULE DISABLE CONTROL REGISTER 2 (CONTINUED)

bit 3 OC4MD: Output Compare 4 Module Disable bit 1 = Output Compare 4 module is disabled 0 = Output Compare 4 module is enabled bit 2 OC3MD: Output Compare 3 Module Disable bit 1 = Output Compare 3 module is disabled 0 = Output Compare 3 module is enabled bit 1 OC2MD: Output Compare 2 Module Disable bit 1 = Output Compare 2 module is disabled 0 = Output Compare 2 module is enabled bit 0 OC1MD: Output Compare 1 Module Disable bit 1 = Output Compare 1 module is disabled 0 = Output Compare 1 module is enabled

#### REGISTER 10-3: PMD3: PERIPHERAL MODULE DISABLE CONTROL REGISTER 3

R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
T9MD	T8MD	T7MD	T6MD	_	_	_	_
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
_	_	_	_	_	_	I2C2MD	AD2MD <sup>(1)</sup>
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 T9MD: Timer9 Module Disable bit 1 = Timer9 module is disabled 0 = Timer9 module is enabled bit 14 T8MD: Timer8 Module Disable bit 1 = Timer8 module is disabled 0 = Timer8 module is enabled bit 13 T7MD: Timer7 Module Disable bit 1 = Timer7 module is disabled 0 = Timer7 module is enabled bit 12 T6MD: Timer6 Module Disable bit 1 = Timer6 module is disabled 0 = Timer6 module is enabled bit 11-2 Unimplemented: Read as '0' bit 1 I2C2MD: I2C2 Module Disable bit 1 = I2C2 module is disabled 0 = I2C2 module is enabled AD2MD: AD2 Module Disable bit(1) bit 0 1 = AD2 module is disabled 0 = AD2 module is enabled

**Note 1:** The PCFGx bits have no effect if the ADC module is disabled by setting this bit. In this case, all port pins multiplexed with ANx will be in Digital mode.

dsPIC33	3FJXX/	KIVICX	U6A/X	U8A/X	10A	
NOTES:						

#### 11.0 I/O PORTS

Note 1: This data sheet summarizes the features of the dsPIC33FJXXXMCX06A/X08A/X10A family of devices. However, it is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to Section 10. "I/O Ports" (DS70193) in the "dsPIC33F/PIC24H Family Reference Manual", which is available from the Microchip web site (www.microchip.com).

2: Some registers and associated bits described in this section may not be available on all devices. Refer to Section 4.0 "Memory Organization" in this data sheet for device-specific register and bit information.

All of the device pins (except VDD, VSS, MCLR and OSC1/CLKIN) are shared between the peripherals and the parallel I/O ports. All I/O input ports feature Schmitt Trigger 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 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/O. The Data Direction register (TRISx) 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 latch (LATx), read the latch. Writes to the latch, write the latch. Reads from the port (PORTx), read the port pins, while 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 pins will read as zeros.

When a pin is shared with another peripheral or function that is defined as an input only, it is nevertheless regarded as a dedicated port because there is no other competing source of outputs. An example is the INT4 pin.

**Output Multiplexers Peripheral Module** Peripheral Input Data I/O Peripheral Module Enable Output Enable Peripheral Output Enable Peripheral Output Data **PIO Module** Output Data Read TRIS Data Bus D I/O Pin **WR TRIS** CK TRIS Latch D WR LAT + **WR PORT** ·CK Data Latch Read LAT Input Data Read Port

FIGURE 11-1: BLOCK DIAGRAM OF A TYPICAL SHARED PORT STRUCTURE

#### 11.2 Open-Drain Configuration

In addition to the PORT, LAT and TRIS registers for data control, some port pins can also be individually configured for either 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 5V tolerant pins by using external pull-up resistors. The maximum open-drain voltage allowed is the same as the maximum VIH specification.

See the "Pin Diagrams" section for the available pins and their functionality.

# 11.3 Configuring Analog Port Pins

The ADxPCFGH, ADxPCFGL and TRIS registers control the operation of the ADC port pins. The port pins that are desired as analog inputs must have their corresponding TRIS bit set (input). If the TRIS bit is cleared (output), the digital output level (VOH or VOL) is converted.

Clearing any bit in the ADxPCFGH or ADxPCFGL register configures the corresponding bit to be an analog pin. This is also the Reset state of any I/O pin that has an analog (ANx) function associated with it.

Note: In devices with two ADC modules, if the corresponding PCFG bit in either AD1PCFGH(L) and AD2PCFGH(L) is cleared, the pin is configured as an analog input.

When reading the PORT register, all pins configured as analog input channels will read as cleared (a low level).

Pins configured as digital inputs will not convert an analog input. Analog levels on any pin that is defined as a digital input (including the ANx pins) can cause the input buffer to consume current that exceeds the device specifications.

Note: The voltage on an analog input pin can be between -0.3V to (VDD + 0.3 V).

#### 11.4 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.

### 11.5 Input Change Notification

The input change notification function of the I/O ports allows the dsPIC33FJXXXMCX06A/X08A/X10A devices to generate interrupt requests to the processor in response to a change-of-state 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 24 external signals (CN0 through CN23) that can be selected (enabled) for generating an interrupt request on a change-of-state.

There are four control registers associated with the CN module. The CNEN1 and CNEN2 registers contain the CN Interrupt Enable (CNxIE) 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 also has a weak pull-up connected to it. The pull-ups act as a current source that is connected to the pin and eliminate the need for external resistors when push button or keypad devices are connected. The pull-ups are enabled separately using the CNPU1 and CNPU2 registers, which contain the Weak Pull-up Enable bits (CNxPUE) for each of the CN pins. Setting any of the control bits enables the weak pull-ups for the corresponding pins.

**Note:** Pull-ups on change notification pins should always be disabled whenever the port pin is configured as a digital output.

#### **EXAMPLE 11-1: PORT WRITE/READ EXAMPLE**

MOV 0xFF00, W0 ; Configure PORTB<15:8> as inputs
MOV W0, TRISBB ; and PORTB<7:0> as outputs
NOP ; Delay 1 cycle
btss PORTB, #13 ; Next Instruction

### 11.6 I/O Helpful Tips

- In some cases, certain pins as defined in TABLE 26-9: "DC Characteristics: I/O Pin Input Specifications" under "Injection Current", have internal protection diodes to VDD and Vss. The term "Injection Current" is also referred to as "Clamp Current". On designated pins, with sufficient external current limiting precautions by the user, I/O pin input voltages are allowed to be greater or less than the data sheet absolute maximum ratings with nominal VDD with respect to the Vss and VDD supplies. Note that when the user application forward biases either of the high or low side internal input clamp diodes, that the resulting current being injected into the device that is clamped internally by the VDD and VSS power rails, may affect the ADC accuracy by four to six counts.
- I/O pins that are shared with any analog input pin, (i.e., ANx), are always analog pins by default after any reset. Consequently, any pin(s) configured as an analog input pin, automatically disables the digital input pin buffer. As such, any attempt to read a digital input pin will always return a '0' regardless of the digital logic level on the pin if the analog pin is configured. To use a pin as a digital I/O pin on a shared ANx pin, the user application needs to configure the analog pin configuration registers in the ADC module, (i.e., ADxPCFGL, AD1PCFGH), by setting the appropriate bit that corresponds to that I/O port pin to a '1'. On devices with more than one ADC, both analog pin configurations for both ADC modules must be configured as a digital I/O pin for that pin to function as a digital I/O pin.

Note: Although it is not possible to use a digital input pin when its analog function is enabled, it is possible to use the digital I/O output function, TRISx = 0x0, while the analog function is also enabled. However, this is not recommended, particularly if the analog input is connected to an external analog voltage source, which would create signal contention between the analog signal and the output pin driver.

3. Most I/O pins have multiple functions. Referring to the device pin diagrams in the data sheet, the priorities of the functions allocated to any pins are indicated by reading the pin name from left-to-right. The left most function name takes precedence over any function to its right in the naming convention. For example: AN16/T2CK/T7CK/RC1. This indicates that AN16 is the highest priority in this example and will supersede all other functions to its right in the list. Those other functions to its right, even if enabled, would not work as long as any other function to its left was enabled. This rule applies to all of the functions listed for a given pin.

- 4. Each CN pin has a configurable internal weak pull-up resistor. The pull-ups act as a current source connected to the pin, and eliminates the need for external resistors in certain applications. The internal pull-up is to ~(VDD-0.8) not VDD. This is still above the minimum VIH of CMOS and TTL devices.
- 5. When driving LEDs directly, the I/O pin can source or sink more current than what is specified in the VOH/IOH and VOL/IOL DC characteristic specification. The respective IOH and IOL current rating only applies to maintaining the corresponding output at or above the VOH and at or below the VOL levels. However, for LEDs unlike digital inputs of an externally connected device, they are not governed by the same minimum VIH/VIL levels. An I/O pin output can safely sink or source any current less than that listed in the absolute maximum rating section of the data sheet. For example:

VOH = 2.4v @ IOH = -8 mA and VDD = 3.3V

The maximum output current sourced by any 8 mA I/O pin = 12 mA.

LED source current < 12 mA is technically permitted. Refer to the VOH/IOH graphs in **Section 26.0** "Electrical Characteristics" for additional information.

#### 11.7 I/O Resources

Many useful resources related to I/O are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page, which can be accessed using this link, contains the latest updates and additional information.

Note: In the event you are not able to access the product page using the link above, enter this URL in your browser:

http://www.microchip.com/wwwproducts/
Devices.aspx?dDocName=en546066

#### 11.7.1 KEY RESOURCES

- Section 10. "I/O Ports" (DS70193)
- Code Samples
- Application Notes
- Software Libraries
- · Webinars
- All related dsPIC33F/PIC24H Family Reference Manuals Sections
- · Development Tools

dsPIC33	SFJXX/	XIVICX	UbA/X	U8A/X	10A	
NOTES:						

#### 12.0 TIMER1

Note 1: This data sheet summarizes the features of the dsPIC33FJXXXMCX06A/X08A/X10A family of devices. However, it is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to Section 11. "Timers" (DS70205) in the "dsPIC33F/PIC24H Family Reference Manual", which is available from the Microchip web site (www.microchip.com).

2: Some registers and associated bits described in this section may not be available on all devices. Refer to Section 4.0 "Memory Organization" in this data sheet for device-specific register and bit information.

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 the following features:

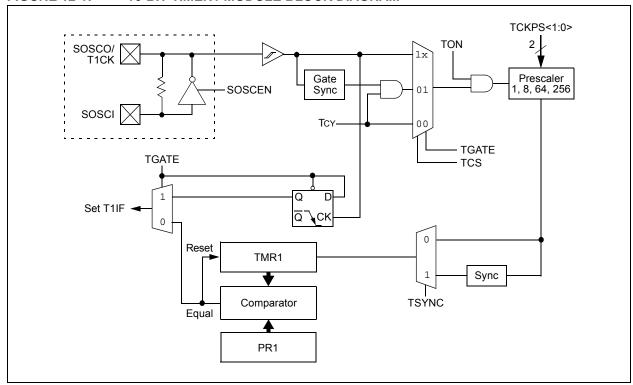
- · Timer gate operation
- · Selectable prescaler settings
- Timer operation during CPU Idle and Sleep modes
- Interrupt on 16-bit Period register match or falling edge of external gate signal

Figure 12-1 presents a block diagram of the 16-bit timer module.

To configure Timer1 for operation, do the following:

- 1. Set the TON bit (= 1) in the T1CON register.
- Select the timer prescaler ratio using the TCKPS<1:0> bits in the T1CON register.
- Set the Clock and Gating modes using the TCS and TGATE bits in the T1CON register.
- 4. Set or clear the TSYNC bit in T1CON to select synchronous or asynchronous operation.
- Load the timer period value into the PR1 register.
- 6. If interrupts are required, set the interrupt enable bit, T1IE. Use the 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

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
TON	_	TSIDL	_	_	_	_	_
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	TCKPS<1:0>		_	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: Stop in Idle Mode bit

1 = Discontinue module operation when device enters Idle mode

0 = Continue module operation in Idle mode

bit 12-7 Unimplemented: Read as '0'

bit 6 TGATE: Timer1 Gated Time Accumulation Enable bit

When T1CS = 1: This bit is ignored. When T1CS = 0:

1 = Gated time accumulation enabled 0 = Gated time accumulation 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'

bit 2 TSYNC: Timer1 External Clock Input Synchronization Select bit

When TCS = 1:

1 = Synchronize external clock input 0 = Do not synchronize external clock input

When TCS = 0: This bit is ignored.

bit 1 TCS: Timer1 Clock Source Select bit

1 = External clock from T1CK pin (on the rising edge)

0 = Internal clock (FCY)

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

Note:

# 13.0 TIMER2/3, TIMER4/5, TIMER6/7 AND TIMER8/9

- Note 1: This data sheet summarizes the features of the dsPIC33FJXXXMCX06A/X08A/X10A family of devices. However, it is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to Section 11. "Timers" (DS70205) in the "dsPIC33F/PIC24H Family Reference Manual", which is available from the Microchip web site (www.microchip.com).
  - 2: Some registers and associated bits described in this section may not be available on all devices. Refer to Section 4.0 "Memory Organization" in this data sheet for device-specific register and bit information.

The Timer2/3, Timer4/5, Timer6/7 and Timer8/9 modules are 32-bit timers that can also be configured as four independent 16-bit timers with selectable operating modes.

As a 32-bit timer, Timer2/3, Timer4/5, Timer6/7 and Timer8/9 operate in three modes:

- Two Independent 16-Bit Timers (e.g., Timer2 and Timer3) with all 16-Bit operating modes (except Asynchronous Counter mode)
- · Single 32-Bit Timer
- Single 32-Bit Synchronous Counter

They also support the following features:

- · Timer Gate Operation
- Selectable Prescaler Settings
- · Timer Operation during Idle and Sleep modes
- · Interrupt on a 32-Bit Period Register Match
- Time Base for Input Capture and Output Compare Modules (Timer2 and Timer3 only)
- ADC1 Event Trigger (Timer2/3 only)
- ADC2 Event Trigger (Timer4/5 only)

Individually, all eight of the 16-bit timers can function as synchronous timers or counters. They also offer the features listed above, except for the event trigger; this is implemented only with Timer2/3. The operating modes and enabled features are determined by setting the appropriate bit(s) in the T2CON, T3CON, T4CON, T5CON, T6CON, T7CON, T8CON and T9CON registers. T2CON, T4CON, T6CON and T8CON are shown in generic form in Register 13-1. T3CON, T5CON, T7CON and T9CON are shown in Register 13-2.

For 32-bit timer/counter operation, Timer2, Timer4, Timer6 or Timer8 is the least significant word; Timer3, Timer5, Timer7 or Timer9 is the most significant word of the 32-bit timers.

For 32-bit operation, T3CON, T5CON, T7CON and T9CON control bits are ignored. Only T2CON, T4CON, T6CON and T8CON control bits are used for setup and control. Timer2, Timer4, Timer6 and Timer8 clock and gate inputs are utilized for the 32-bit timer modules, but an interrupt is generated with the Timer3, Timer5, Ttimer7 and Timer9 interrupt flags.

To configure Timer2/3, Timer4/5, Timer6/7 or Timer8/9 for 32-bit operation, do the following:

- 1. Set the corresponding T32 control bit.
- 2. Select the prescaler ratio for Timer2, Timer4, Timer6 or Timer8 using the TCKPS<1:0> bits.
- 3. Set the Clock and Gating modes using the corresponding TCS and TGATE bits.
- 4. Load the timer period value. PR3, PR5, PR7 or PR9 contains the most significant word of the value, while PR2, PR4, PR6 or PR8 contains the least significant word.
- If interrupts are required, set the interrupt enable bit, T3IE, T5IE, T7IE or T9IE. Use the priority bits, T3IP<2:0>, T5IP<2:0>, T7IP<2:0> or T9IP<2:0>, to set the interrupt priority. While Timer2, Timer4, Timer6 or Timer8 control the timer, the interrupt appears as a Timer3, Timer5, Timer7 or Timer9 interrupt.
- 6. Set the corresponding TON bit.

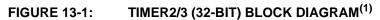
The timer value at any point is stored in the register pair, TMR3:TMR2, TMR5:TMR4, TMR7:TMR6 or TMR9:TMR8. TMR3, TMR5, TMR7 or TMR9 always contain the most significant word of the count, while TMR2, TMR4, TMR6 or TMR8 contain the least significant word.

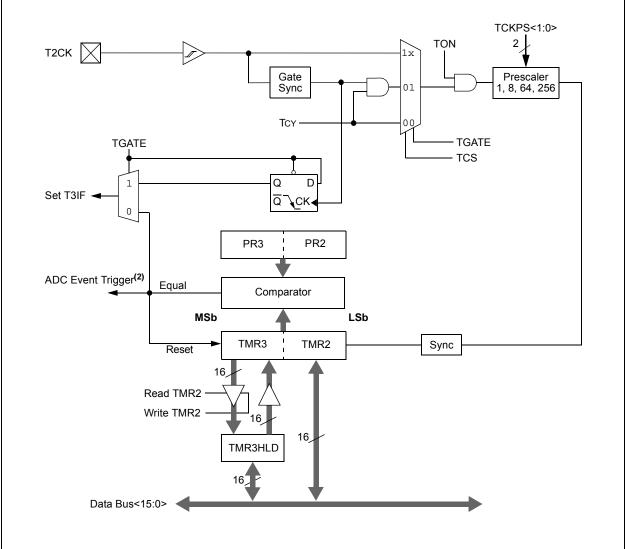
To configure any of the timers for individual 16-bit operation, do the following:

- 1. Clear the T32 bit corresponding to that timer.
- 2. Select the timer prescaler ratio using the TCKPS<1:0> bits.
- Set the Clock and Gating modes using the TCS and TGATE bits.
- 4. Load the timer period value into the PRx register.
- 5. If interrupts are required, set the interrupt enable bit, TxIE. Use the priority bits, TxIP<2:0>, to set the interrupt priority.
- 6. Set the TON bit.

A block diagram for a 32-bit timer pair (Timer4/5) example is shown in Figure 13-1, and a timer (Timer4) operating in 16-bit mode example is shown in Figure 13-2.

**Note:** Only Timer2 and Timer3 can trigger a DMA data transfer.

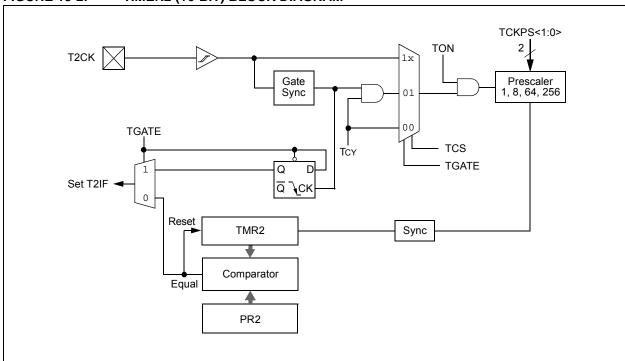




Note 1: The 32-Bit Timer Control bit, T32, must be set for 32-bit timer/counter operation. All control bits are respective to the T2CON register.

2: The ADC event trigger is available only on Timer2/3.

**TIMER2 (16-BIT) BLOCK DIAGRAM FIGURE 13-2:** 



#### REGISTER 13-1: TxCON (T2CON, T4CON, T6CON OR T8CON) CONTROL REGISTER

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
TON	_	TSIDL	_	_	_	_	_
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	TCKPS	S<1:0>	T32	_	TCS <sup>(1)</sup>	_
bit 7							bit 0

Legend:

bit 14

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 T32 = 1:

1 = Starts 32-bit Timerx/y
0 = Stops 32-bit Timerx/y

When T32 = 0:

1 = Starts 16-bit Timerx0 = Stops 16-bit Timerx

Unimplemented: Read as '0'

bit 13 TSIDL: Stop in Idle Mode bit

1 = Discontinue module operation when device enters Idle mode

0 = Continue module operation in Idle mode

bit 12-7 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 enabled0 = Gated time accumulation 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

bit 3 T32: 32-Bit Timer Mode Select bit

1 = Timerx and Timery form a single 32-bit timer 0 = Timerx and Timery act as two 16-bit timers

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

bit 1 TCS: Timerx Clock Source Select bit<sup>(1)</sup>

1 = External clock from TxCK pin (on the rising edge)

0 = Internal clock (Fcy)

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

Note 1: The TxCK pin is not available on all timers. Refer to the "Pin Diagrams" section for the available pins.

#### **REGISTER 13-2:** TyCON (T3CON, T5CON, T7CON OR T9CON) CONTROL REGISTER

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
TON <sup>(1)</sup>	_	TSIDL <sup>(2)</sup>	_	_	_	_	_
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 <sup>(1)</sup>	TCKPS:	<1:0>(1)	_	_	TCS <sup>(1,3)</sup>	_
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<sup>(1)</sup>

1 = Starts 16-bit Timery

0 = Stops 16-bit Timery

bit 14 Unimplemented: Read as '0'

TSIDL: Stop in Idle Mode bit(2) bit 13

1 = Discontinue module operation when device enters Idle mode

0 = Continue module operation in Idle mode

bit 12-7 Unimplemented: Read as '0'

**TGATE:** Timery Gated Time Accumulation Enable bit<sup>(1)</sup> bit 6

> When TCS = 1: This bit is ignored.

When TCS = 0:

1 = Gated time accumulation enabled 0 = Gated time accumulation disabled

TCKPS<1:0>: Timer3 Input Clock Prescale Select bits(1) bit 5-4

> 11 = 1:25610 = 1:64

01 = 1:8

00 = 1:1

bit 3-2 Unimplemented: Read as '0'

TCS: Timery Clock Source Select bit (1,3) bit 1

1 = External clock from TyCK pin (on the rising edge)

0 = Internal clock (Fcy)

bit 0 Unimplemented: Read as '0'

Note 1: When 32-bit operation is enabled (T2CON<3> = 1), these bits have no effect on Timery operation; all timer functions are set through T2CON.

- 2: When 32-bit timer operation is enabled (T32 = 1) in the Timer Control register (TxCON<3>), the TSIDL bit must be cleared to operate the 32-bit timer in Idle mode.
- 3: The TyCK pin is not available on all timers. Refer to the "Pin Diagrams" section for the available pins.

dsPIC33	dsPIC33FJXXXMCX06A/X08A/X10A							
NOTES:								

# 14.0 INPUT CAPTURE

Note 1: This data sheet summarizes the features of the dsPIC33FJXXXMCX06A/X08A/X10A family of devices. However, it is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to Section 12. "Input Capture" (DS70198) in the "dsPIC33F/PIC24H Family Reference Manual", which is available from the Microchip web site (www.microchip.com).

2: Some registers and associated bits described in this section may not be available on all devices. Refer to Section 4.0 "Memory Organization" in this data sheet for device-specific register and bit information.

The input capture module is useful in applications requiring frequency (period) and pulse measurement. The dsPIC33FJXXXMCX06A/X08A/X10A devices support up to eight input capture channels.

The input capture module captures the 16-bit value of the selected Time Base register when an event occurs at the ICx pin. The events that cause a capture event are listed below in three categories:

- 1. Simple Capture Event modes
  - Capture timer value on every falling edge of input at ICx pin
  - Capture timer value on every rising edge of input at ICx pin
- Capture timer value on every edge (rising and falling) of input at ICx pin
- 3. Prescaler Capture Event modes
  - Capture timer value on every 4th rising edge of input at ICx pin
  - Capture timer value on every 16th rising edge of input at ICx pin

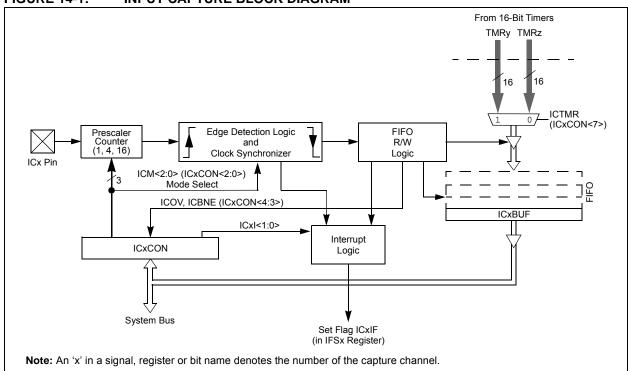
Each input capture channel can select between one of two 16-bit timers (Timer2 or Timer3) for the time base. The selected timer can use either an internal or external clock.

Other operational features include the following:

- Device wake-up from capture pin during CPU Sleep and Idle modes
- · Interrupt on input capture event
- · 4-word FIFO buffer for capture values
- Interrupt optionally generated after 1, 2, 3 or 4 buffer locations are filled
- Input capture can also be used to provide additional sources of external interrupts

Note: Only IC1 and IC2 can trigger a DMA data transfer. If DMA data transfers are required, the FIFO buffer size must be set to '1' (ICI<1:0> = 00).

FIGURE 14-1: INPUT CAPTURE BLOCK DIAGRAM



## 14.1 Input Capture Registers

#### REGISTER 14-1: ICxCON: INPUT CAPTURE x CONTROL REGISTER

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

R/W-0	R/W-0	R/W-0	R-0, HC	R-0, HC	R/W-0	R/W-0	R/W-0
ICTMR <sup>(1)</sup>	ICI<	1:0>	ICOV	ICBNE		ICM<2:0>	
bit 7							bit 0

Legend:	HC = Hardware Clearable bit					
R = Readable bit	W = Writable 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 ICSIDL: Input Capture Module Stop in Idle Control bit

1 = Input capture module will halt in CPU Idle mode
 0 = Input capture module will continue to operate in CPU Idle mode

bit 12-8 **Unimplemented:** Read as '0'

bit 7 ICTMR: Input Capture Timer Select bits<sup>(1)</sup>

 ${\tt 1}$  = TMR2 contents are captured on capture event  ${\tt 0}$  = TMR3 contents are captured on capture event

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 Overflow Status Flag bit (read-only)

1 = Input capture overflow occurred0 = No input capture overflow occurred

bit 3 **ICBNE:** Input Capture 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 Mode Select bits

111 = Input capture functions as interrupt pin only when device is in Sleep or Idle mode (Rising edge detect only, all other control bits are not applicable.)

110 = Unused (module disabled)

101 = Capture mode, every 16th rising edge100 = Capture mode, every 4th rising edge

011 = Capture mode, every rising edge010 = Capture mode, every falling edge

001 = Capture mode, every edge (rising and falling)

(ICI<1:0> bits do not control interrupt generation for this mode.)

000 = Input capture module turned off

#### 15.0 OUTPUT COMPARE

Note 1: This data sheet summarizes the features of the dsPIC33FJXXXMCX06A/X08A/X10A families of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to the "dsPIC33F/PIC24H Family Reference Manual", Section 13. "Output Compare" (DS70209), which is available on the Microchip web site (www.microchip.com).

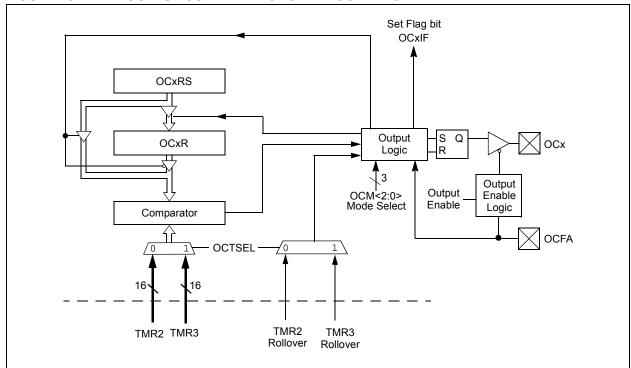
2: Some registers and associated bits described in this section may not be available on all devices. Refer to Section 4.0 "Memory Organization" in this data sheet for device-specific register and bit information.

The output compare module can select either Timer2 or Timer3 for its time base. The module compares the value of the timer with the value of one or two Compare registers depending on the operating mode selected. The state of the output pin changes when the timer value matches the Compare register value. The output compare module generates either a single output pulse, or a sequence of output pulses, by changing the state of the output pin on the compare match events. The output compare module can also generate interrupts on compare match events.

The output compare module has multiple operating modes:

- · Active-Low One-Shot mode
- · Active-High One-Shot mode
- · Toggle mode
- · Delayed One-Shot mode
- · Continuous Pulse mode
- · PWM mode without Fault Protection
- · PWM mode with Fault Protection

#### FIGURE 15-1: OUTPUT COMPARE MODULE BLOCK DIAGRAM



# 15.1 Output Compare Modes

Configure the Output Compare modes by setting the appropriate Output Compare Mode bits (OCM<2:0>) in the Output Compare Control register (OCxCON<2:0>). Table 15-1 lists the different bit settings for the Output Compare modes. Figure 15-2 illustrates the output compare operation for various modes. The user

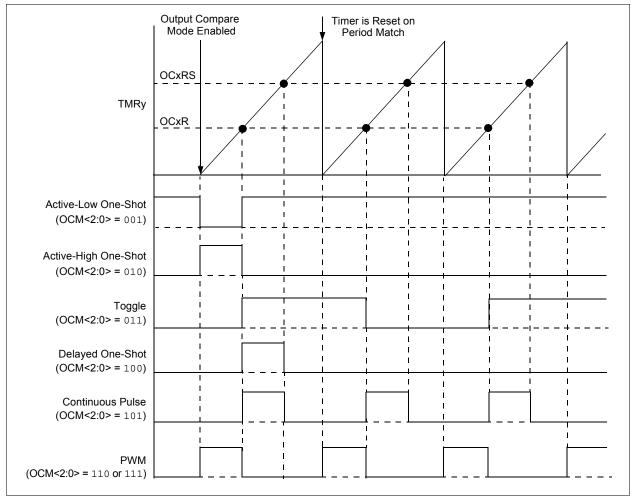
application must disable the associated timer when writing to the Output Compare Control registers to avoid malfunctions.

Note: See Section 13. "Output Compare" (DS70209) in the "dsPIC33F/PIC24H Family Reference Manual" for OCxR and OCxRS register restrictions.

TABLE 15-1: OUTPUT COMPARE MODES

OCM<2:0>	Mode	OCx Pin Initial State	OCx Interrupt Generation
000	Module Disabled	Controlled by GPIO register	_
001	Active-Low One-Shot	0	OCx rising edge
010	Active-High One-Shot	1	OCx falling edge
011	Toggle	Current output is maintained	OCx rising and falling edge
100	Delayed One-Shot	0	OCx falling edge
101	Continuous Pulse	0	OCx falling edge
110	PWM without Fault Protection	'0' if OCxR is zero, '1' if OCxR is non-zero	No interrupt
111	PWM with Fault Protection	'0' if OCxR is zero, '1' if OCxR is non-zero	OCFA falling edge for OC1 to OC4

FIGURE 15-2: OUTPUT COMPARE OPERATION



## REGISTER 15-1: OCxCON: OUTPUT COMPARE x CONTROL REGISTER (x = 1, 2)

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

U-0	U-0	U-0	R-0, HC	R/W-0	R/W-0	R/W-0	R/W-0
_	_	_	OCFLT	OCTSEL		OCM<2:0>	
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 15-14 Unimplemented: Read as '0'

bit 13 OCSIDL: Stop Output Compare 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 12-5 Unimplemented: Read as '0'

bit 4 OCFLT: PWM Fault Condition Status bit

1 = PWM Fault condition has occurred (cleared in hardware only)

0 = No PWM Fault condition has occurred (this bit is only used when OCM<2:0> = 111)

bit 3 OCTSEL: Output Compare Timer Select bit

1 = Timer3 is the clock source for Compare x

0 = Timer2 is the clock source for Compare x

bit 2-0 OCM<2:0>: Output Compare Mode Select bits

111 = PWM mode on OCx, Fault pin enabled

110 = PWM mode on OCx, Fault pin disabled

101 = Initialize OCx pin low, generate continuous output pulses on OCx pin

100 = Initialize OCx pin low, generate single output pulse on OCx pin

011 = Compare event toggles OCx pin

010 = Initialize OCx pin high, compare event forces OCx pin low

001 = Initialize OCx pin low, compare event forces OCx pin high

000 = Output compare channel is disabled

dsPIC33FJXXXWCXU6A/XU8A/X1UA									
NOTES:									

# 16.0 MOTOR CONTROL PWM MODULE

- Note 1: This data sheet summarizes the features of the dsPIC33FJXXXMCX06A/X08A/X10A family of devices. However, it is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to Section 14. "Motor Control PWM" (DS70187) in the "dsPIC33F/PIC24H Family Reference Manual", which is available from the Microchip web site (www.microchip.com).
  - 2: Some registers and associated bits described in this section may not be available on all devices. Refer to Section 4.0 "Memory Organization" in this data sheet for device-specific register and bit information.

This module simplifies the task of generating multiple, synchronized Pulse-Width Modulated (PWM) outputs. In particular, the following power and motion control applications are supported by the PWM module:

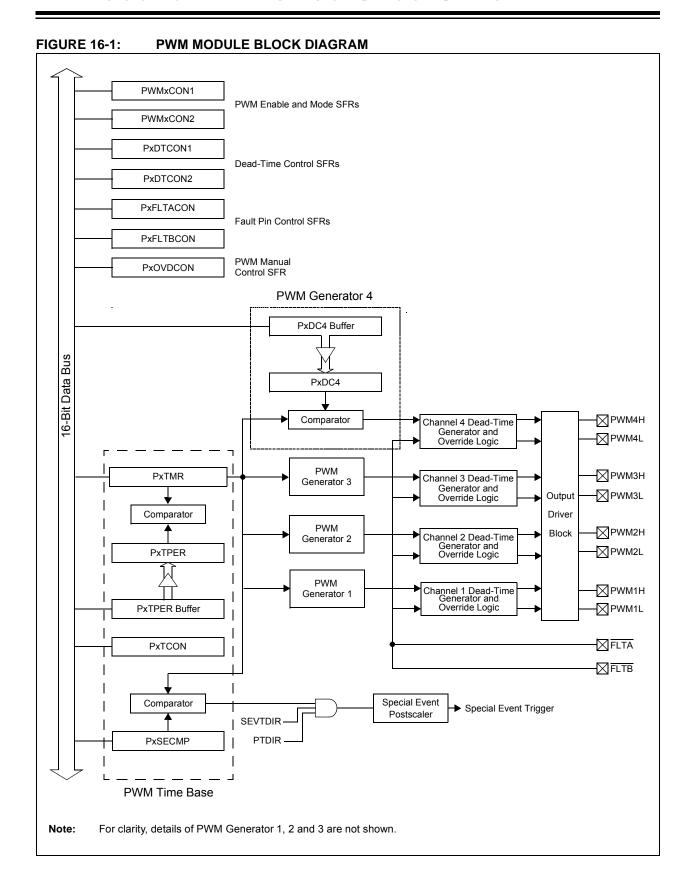
- · 3-Phase AC Induction Motor
- · Switched Reluctance (SR) Motor
- · Brushless DC (BLDC) Motor
- Uninterruptible Power Supply (UPS)

The PWM module has the following features:

- Eight PWM I/O pins with four duty cycle generators
- · Up to 16-bit resolution
- · 'On-the-fly' PWM frequency changes
- Edge and Center-Aligned Output modes
- · Single Pulse Generation mode
- Interrupt support for asymmetrical updates in Center-Aligned mode
- Output override control for Electrically Commutative Motor (ECM) operation
- 'Special Event' comparator for scheduling other peripheral events
- Fault pins to optionally drive each of the PWM output pins to a defined state
- Duty cycle updates are configurable to be immediate or synchronized to the PWM time base

This module contains four duty cycle generators, numbered 1 through 4. The module has eight PWM output pins, numbered PWM1H/PWM1L through PWM4H/PWM4L. The eight I/O pins are grouped into high/low numbered pairs, denoted by the suffix H or L, respectively. For complementary loads, the low PWM pins are always the complement of the corresponding high I/O pin.

The PWM module allows several modes of operation which are beneficial for specific power control applications.



### REGISTER 16-1: PXTCON: PWMX TIME BASE CONTROL REGISTER

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
PTEN	_	PTSIDL	_	_	_	_	_
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
	PTOPS<3:0>				PTCKPS<1:0>		D<1:0>
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 PTEN: PWM Time Base Timer Enable bit

1 = PWM time base is on 0 = PWM time base is off

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

bit 13 PTSIDL: PWM Time Base Stop in Idle Mode bit

1 = PWM time base halts in CPU Idle mode

0 = PWM time base runs in CPU Idle mode

bit 12-8 **Unimplemented:** Read as '0'

bit 7-4 PTOPS<3:0>: PWM Time Base Output Postscale Select bits

1111 = 1:16 postscale

.

0001 = 1:2 postscale

0000 = 1:1 postscale

bit 3-2 PTCKPS<1:0>: PWM Time Base Input Clock Prescale Select bits

11 = PWM time base input clock period is 64 Tcy (1:64 prescale)

10 = PWM time base input clock period is 16 Tcy (1:16 prescale)

01 = PWM time base input clock period is 4 Tcy (1:4 prescale)

00 = PWM time base input clock period is Tcy (1:1 prescale)

bit 1-0 PTMOD<1:0>: PWM Time Base Mode Select bits

11 = PWM time base operates in a Continuous Up/Down Count mode with interrupts for double PWM updates

10 = PWM time base operates in a Continuous Up/Down Count mode

01 = PWM time base operates in a Single Pulse mode

00 = PWM time base operates in a Free-Running mode

### REGISTER 16-2: PXTMR: PWMX TIMER COUNT VALUE REGISTER

R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTDIR				PTMR<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	
PTMR<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 **PTDIR:** PWM Time Base Count Direction Status bit (read-only)

1 = PWM time base is counting down0 = PWM time base is counting up

bit 14-0 PTMR <14:0>: PWM Time Base Register Count Value bits

### REGISTER 16-3: PXTPER: PWMx TIME BASE PERIOD REGISTER

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_				PTPER<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
			PTPE	R<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 PTPER<14:0>: PWM Time Base Period Value bits

### REGISTER 16-4: PXSECMP: PWMx SPECIAL EVENT COMPARE 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
SEVTDIR <sup>(1)</sup>			SI	EVTCMP<14:8	3>(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
			SEVTCM	P<7:0>(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 SEVTDIR: Special Event Trigger Time Base Direction bit<sup>(1)</sup>

1 = A Special Event Trigger will occur when the PWM time base is counting downwards

0 = A Special Event Trigger will occur when the PWM time base is counting upwards

bit 14-0 SEVTCMP<14:0>: Special Event Compare Value bits<sup>(2)</sup>

Note 1: SEVTDIR is compared with PTDIR (PTMR<15>) to generate the Special Event Trigger.

2: SEVTCMP<14:0> is compared with PTMR<14:0> to generate the Special Event Trigger.

### REGISTER 16-5: PWMxCON1: PWMx CONTROL REGISTER 1

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	_	_	PMOD4	PMOD3	PMOD2	PMOD1
bit 15							bit 8

| R/W-1                |
|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|
| PEN4H <sup>(1)</sup> | PEN3H <sup>(1)</sup> | PEN2H <sup>(1)</sup> | PEN1H <sup>(1)</sup> | PEN4L <sup>(1)</sup> | PEN3L <sup>(1)</sup> | PEN2L <sup>(1)</sup> | PEN1L <sup>(1)</sup> |
| 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 11-8 **PMOD<4:1>:** PWM I/O Pair Mode bits

1 = PWM I/O pin pair is in the Independent PWM Output mode 0 = PWM I/O pin pair is in the Complementary Output mode

bit 7-4 **PEN4H:PEN1H:** PWMxH I/O Enable bits<sup>(1)</sup>

1 = PWMxH pin is enabled for PWM output

0 = PWMxH pin is disabled; I/O pin becomes general purpose I/O

bit 3-0 **PEN4L:PEN1L:** PWMxL I/O Enable bits<sup>(1)</sup>

1 = PWMxL pin is enabled for PWM output

0 = PWMxL pin is disabled; I/O pin becomes general purpose I/O

**Note 1:** Reset condition of the PENxH and PENxL bits depends on the value of the PWMPIN Configuration bit in the FPOR Configuration register.

### REGISTER 16-6: PWMxCON2: PWMx CONTROL REGISTER 2

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	_	_		SEVOP	PS<3:0>	
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
_	_	_	_	_	IUE	OSYNC	UDIS
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 11-8 SEVOPS<3:0>: PWM Special Event Trigger Output Postscale Select bits

1111 = 1:16 postscale

.

0001 = 1:2 postscale 0000 = 1:1 postscale

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

bit 2 **IUE:** Immediate Update Enable bit

1 = Updates to the active PDC registers are immediate

0 = Updates to the active PDC registers are synchronized to the PWM time base

bit 1 **OSYNC:** Output Override Synchronization bit

1 = Output overrides via the OVDCON register are synchronized to the PWM time base

0 = Output overrides via the OVDCON register occur on next Tcy boundary

bit 0 UDIS: PWM Update Disable bit

1 = Updates from Duty Cycle and Period Buffer registers are disabled0 = Updates from Duty Cycle and Period Buffer registers are enabled

### REGISTER 16-7: PXDTCON1: PWMx DEAD-TIME 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
DTBPS	S<1:0>			DTB	<5:0>		
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
DTAPS	6<1:0>			DTA	<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 15-14 **DTBPS<1:0>:** Dead-Time Unit B Prescale Select bits

11 = Clock period for Dead-Time Unit B is 8 Tcy

10 = Clock period for Dead-Time Unit B is 4 Tcy

01 = Clock period for Dead-Time Unit B is 2 TcY

00 = Clock period for Dead-Time Unit B is Tcy

bit 13-8 **DTB<5:0>:** Unsigned 6-Bit Dead-Time Value for Dead-Time Unit B bits

bit 7-6 DTAPS<1:0>: Dead-Time Unit A Prescale Select bits

11 = Clock period for Dead-Time Unit A is 8 Tcy

10 = Clock period for Dead-Time Unit A is 4 TcY

01 = Clock period for Dead-Time Unit A is 2 TcY

00 = Clock period for Dead-Time Unit A is Tcy

bit 5-0 DTA<5:0>: Unsigned 6-Bit Dead-Time Value for Dead-Time Unit A bits

### REGISTER 16-8: PXDTCON2: PWMx DEAD-TIME CONTROL REGISTER 2

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

| R/W-0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| DTS4A | DTS4I | DTS3A | DTS3I | DTS2A | DTS2I | DTS1A | DTS1I |
| 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

DTS4A: Dead-Time Select for PWM4 Signal Going Active bit

1 = Dead time provided from Unit B

0 = Dead time provided from Unit A

bit 6

DTS4I: Dead-Time Select for PWM4 Signal Going Inactive bit

1 = Dead time provided from Unit B

0 = Dead time provided from Unit A

bit 5

DTS3A: Dead-Time Select for PWM3 Signal Going Active bit

1 = Dead time provided from Unit B

0 = Dead time provided from Unit A

bit 4 DTS3I: Dead-Time Select for PWM3 Signal Going Inactive bit

1 = Dead time provided from Unit B0 = Dead time provided from Unit A

bit 3 DTS2A: Dead-Time Select for PWM2 Signal Going Active bit

1 = Dead time provided from Unit B0 = Dead time provided from Unit A

bit 2 DTS2I: Dead-Time Select for PWM2 Signal Going Inactive bit

1 = Dead time provided from Unit B0 = Dead time provided from Unit A

bit 1 DTS1A: Dead-Time Select for PWM1 Signal Going Active bit

1 = Dead time provided from Unit B0 = Dead time provided from Unit A

bit 0 DTS1I: Dead-Time Select for PWM1 Signal Going Inactive bit

1 = Dead time provided from Unit B0 = Dead time provided from Unit A

### REGISTER 16-9: PXFLTACON: PWMX FAULT A CONTROL REGISTER

| R/W-0  |
|--------|--------|--------|--------|--------|--------|--------|--------|
| FAOV4H | FAOV4L | FAOV3H | FAOV3L | FAOV2H | FAOV2L | FAOV1H | FAOV1L |
| bit 15 |        |        |        |        |        |        | bit 8  |

R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
FLTAM	_	_	_	FAEN4	FAEN3	FAEN2	FAEN1
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 FAOVxH<4:1>: FAOVxL<4:1>: Fault Input A PWM Override Value bits

1 = The PWM output pin is driven active on an external Fault input event 0 = The PWM output pin is driven inactive on an external Fault input event

bit 7 FLTAM: Fault A Mode bit

1 = The Fault A input pin functions in the Cycle-by-Cycle mode

0 = The Fault A input pin latches all control pins to the states programmed in FLTACON<15:8>

bit 6-4 Unimplemented: Read as '0'

bit 3 FAEN4: Fault Input A Enable bit

1 = PWM4H/PWM4L pin pair is controlled by Fault Input A 0 = PWM4H/PWM4L pin pair is not controlled by Fault Input A

bit 2 FAEN3: Fault Input A Enable bit

1 = PWM3H/PWM3L pin pair is controlled by Fault Input A 0 = PWM3H/PWM3L pin pair is not controlled by Fault Input A

bit 1 FAEN2: Fault Input A Enable bit

1 = PWM2H/PWM2L pin pair is controlled by Fault Input A 0 = PWM2H/PWM2L pin pair is not controlled by Fault Input A

bit 0 FAEN1: Fault Input A Enable bit

1 = PWM1H/PWM1L pin pair is controlled by Fault Input A 0 = PWM1H/PWM1L pin pair is not controlled by Fault Input A

# REGISTER 16-10: PXFLTBCON: PWMx FAULT B CONTROL REGISTER

| R/W-0  |
|--------|--------|--------|--------|--------|--------|--------|--------|
| FBOV4H | FBOV4L | FBOV3H | FBOV3L | FBOV2H | FBOV2L | FBOV1H | FBOV1L |
| bit 15 |        |        |        |        |        |        | bit 8  |

R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
FLTBM	_	_	_	FBEN4 <sup>(1)</sup>	FBEN3 <sup>(1)</sup>	FBEN2 <sup>(1)</sup>	FBEN1 <sup>(1)</sup>
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	FBOVxH<4:1>:FBOVxL<4:1>: Fault Input B PWM Override Value bits  1 = The PWM output pin is driven active on an external Fault input event  0 = The PWM output pin is driven inactive on an external Fault input event
bit 7	<b>FLTBM:</b> Fault B Mode bit  1 = The Fault B input pin functions in the Cycle-by-Cycle mode
	0 = The Fault B input pin latches all control pins to the states programmed in FLTBCON<15:8>
bit 6-4	Unimplemented: Read as '0'
bit 3	FBEN4: Fault Input B Enable bit <sup>(1)</sup>
	1 = PWM4H/PWM4L pin pair is controlled by Fault Input B 0 = PWM4H/PWM4L pin pair is not controlled by Fault Input B
bit 2	FBEN3: Fault Input B Enable bit <sup>(1)</sup>
	<ul><li>1 = PWM3H/PWM3L pin pair is controlled by Fault Input B</li><li>0 = PWM3H/PWM3L pin pair is not controlled by Fault Input B</li></ul>
bit 1	FBEN2: Fault Input B Enable bit <sup>(1)</sup>
	<ul><li>1 = PWM2H/PWM2L pin pair is controlled by Fault Input B</li><li>0 = PWM2H/PWM2L pin pair is not controlled by Fault Input B</li></ul>
bit 0	FBEN1: Fault Input B Enable bit <sup>(1)</sup>
	<ul><li>1 = PWM1H/PWM1L pin pair is controlled by Fault Input B</li><li>0 = PWM1H/PWM1L pin pair is not controlled by Fault Input B</li></ul>

Note 1: Fault A pin has priority over Fault B pin, if enabled.

### REGISTER 16-11: PXOVDCON: PWMx OVERRIDE CONTROL REGISTER

| R/W-1  |
|--------|--------|--------|--------|--------|--------|--------|--------|
| POVD4H | POVD4L | POVD3H | POVD3L | POVD2H | POVD2L | POVD1H | POVD1L |
| bit 15 |        |        |        |        |        |        | bit 8  |

| R/W-0  |
|--------|--------|--------|--------|--------|--------|--------|--------|
| POUT4H | POUT4L | POUT3H | POUT3L | POUT2H | POUT2L | POUT1H | POUT1L |
| 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 POVDxH<4:1>:POVDxL<4:1>: PWM Output Override bits

1 = Output on PWMx I/O pin is controlled by the PWM generator

0 = Output on PWMx I/O pin is controlled by the value in the corresponding POUTxH:POUTxL bit

bit 7-0 **POUTxH<4:1>:POUTxL<4:1>:** PWM Manual Output bits

1 = PWMx I/O pin is driven active when the corresponding POVDxH:POVDxL bit is cleared

0 = PWMx I/O pin is driven inactive when the corresponding POVDxH:POVDxL bit is cleared

### REGISTER 16-12: PxDC1: PWMx DUTY CYCLE 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	
PDC1<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	R/W-0	
PDC1<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 15-0 PDC1<15:0>: PWM Duty Cycle #1 Value bits

### REGISTER 16-13: PxDC2: PWMx DUTY CYCLE 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
			PDC2	<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	R/W-0	
PDC2<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 15-0 PDC2<15:0>: PWM Duty Cycle #2 Value bits

### REGISTER 16-14: PxDC3: PWMx DUTY CYCLE REGISTER 3

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
			PDC3	<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	R/W-0
			PDC3	i<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 15-0 PDC3<15:0>: PWM Duty Cycle #3 Value bits

### REGISTER 16-15: PxDC4: PWMx DUTY CYCLE REGISTER 4

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	
PDC4<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	R/W-0				
	PDC4<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 15-0 PDC4<15:0>: PWM Duty Cycle #4 Value bits

# 17.0 QUADRATURE ENCODER INTERFACE (QEI) MODULE

Note 1: This data sheet summarizes the features of the dsPIC33FJXXXMCX06A/X08A/X10A family of devices. However, it is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to Section 15. "Quadrature Encoder Interface (QEI)" (DS70208) in the "dsPIC33F/PIC24H Family Reference Manual", which is available from the Microchip web site (www.microchip.com).

2: Some registers and associated bits described in this section may not be available on all devices. Refer to Section 4.0 "Memory Organization" in this data sheet for device-specific register and bit information.

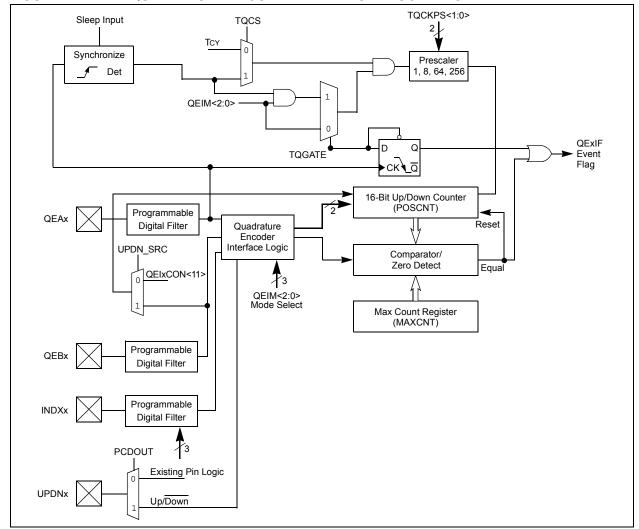
This section describes the Quadrature Encoder Interface (QEI) module and associated operational modes. The QEI module provides the interface to incremental encoders for obtaining mechanical position data.

The operational features of the QEI include the following:

- Three input channels for two phase signals and an index pulse
- · 16-bit up/down position counter
- · Count direction status
- Position Measurement (x2 and x4) mode
- · Programmable digital noise filters on inputs
- · Alternate 16-Bit Timer/Counter mode
- · Quadrature Encoder Interface interrupts

The QEI module's operating mode is determined by setting the appropriate bits, QEIM<2:0> (QEIxCON<10:8>). Figure 17-1 depicts the Quadrature Encoder Interface block diagram.

### FIGURE 17-1: QUADRATURE ENCODER INTERFACE BLOCK DIAGRAM



### REGISTER 17-1: QEIXCON: QEIX CONTROL REGISTER

R/W-0	U-0	R/W-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
CNTERR	_	QEISIDL	INDEX	UPDN		QEIM<2:0>	
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
SWPAB	PCDOUT	TQGATE	TQCKF	PS<1:0>	POSRES	TQCS	UPDN_SRC <sup>(1)</sup>
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 CNTERR: Count Error Status Flag bit

1 = Position count error has occurred

0 = No position count error has occurred

(CNTERR flag only applies when QEIM<2:0> = '110' or '100')

bit 14 Unimplemented: Read as '0'

bit 13 QEISIDL: Stop in Idle Mode bit

1 = Discontinue module operation when device enters Idle mode

0 = Continue module operation in Idle mode

bit 12 **INDEX:** Index Pin State Status bit (read-only)

1 = Index pin is High

0 = Index pin is Low

bit 11 **UPDN:** Position Counter Direction Status bit

1 = Position counter direction is positive (+)

0 = Position counter direction is negative (-)

(Read-only bit when QEIM<2:0> = '1xx'. Read/write bit when QEIM<2:0> = 001.)

bit 10-8 **QEIM<2:0>:** Quadrature Encoder Interface Mode Select bits

111 = Quadrature Encoder Interface enabled (x4 mode) with Position Counter Reset by match (MAXCNT)

110 = Quadrature Encoder Interface enabled (x4 mode) with Index Pulse Reset of position counter

101 = Quadrature Encoder Interface enabled (x2 mode) with Position Counter Reset by match (MAXCNT)

100 = Quadrature Encoder Interface enabled (x2 mode) with Index Pulse Reset of position counter

011 = Unused (module disabled)

010 = Unused (module disabled)

001 = Starts 16-bit Timer

000 = Quadrature Encoder Interface/timer off

bit 7 SWPAB: Phase A and Phase B Input Swap Select bit

1 = Phase A and Phase B inputs swapped

0 = Phase A and Phase B inputs not swapped

bit 6 **PCDOUT:** Position Counter Direction State Output Enable bit

1 = Position counter direction status output enable (QEI logic controls state of I/O pin)

0 = Position counter direction status output disabled (normal I/O pin operation)

bit 5 TQGATE: Timer Gated Time Accumulation Enable bit

1 = Timer gated time accumulation enabled

0 = Timer gated time accumulation disabled

Note 1: When configured for QEI mode, the control bit is a 'don't care'.

# REGISTER 17-1: QEIXCON: QEIX CONTROL REGISTER (CONTINUED)

bit 4-3 **TQCKPS<1:0>:** Timer Input Clock Prescale Select bits

11 = 1:256 prescale value 10 = 1:64 prescale value 01 = 1:8 prescale value 00 = 1:1 prescale value

(Prescaler utilized for 16-Bit Timer mode only.)

bit 2 **POSRES:** Position Counter Reset Enable bit

1 = Index pulse resets position counter

0 = Index pulse does not reset position counter (Bit only applies when QEIM<2:0> = 100 or 110.)

bit 1 TQCS: Timer Clock Source Select bit

1 = External clock from QEA pin (on the rising edge)

0 = Internal clock (Tcy)

bit 0 **UPDN\_SRC:** Position Counter Direction Selection Control bit<sup>(1)</sup>

1 = QEB pin state defines position counter direction

0 = Control/status bit, UPDN (QEICON<11>), defines Position Counter (POSxCNT) direction

Note 1: When configured for QEI mode, the control bit is a 'don't care'.

### REGISTER 17-2: DFLTxCON: DIGITAL FILTER x CONTROL REGISTER

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
_	_	_	_	_	IMV<	2:0>	CEID
bit 15							bit 8

R/W-0	R/W-0	U-0	U-0	U-0	U-0
QEOUT	QECK<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 15-11 **Unimplemented:** Read as '0'

bit 10-9 IMV<1:0>: Index Match Value bits

These bits allow the user to specify the state of the QEAx and QEBx input pins during an index pulse

when the POSxCNT register is to be reset.

In 4X Quadrature Count Mode:

IMV1 = Required state of Phase B input signal for match on index pulse

IMV0 = Required state of Phase A input signal for match on index pulse

In 2X Quadrature Count Mode:

IMV1 = Selects phase input signal for index state match (0 = Phase A, 1 = Phase B)

IMV0 = Required state of the selected Phase input signal for match on index pulse

bit 8 **CEID:** Count Error Interrupt Disable bit

1 = Interrupts due to count errors are disabled

0 = Interrupts due to count errors are enabled

bit 7 QEOUT: QEAx/QEBx/INDXx Pin Digital Filter Output Enable bit

1 = Digital filter outputs enabled

0 = Digital filter outputs disabled (normal pin operation)

bit 6-4 QECK<2:0>: QEAx/QEBx/INDXx Digital Filter Clock Divide Select Bits

111 = 1:256 clock divide

110 = 1:128 clock divide

101 = 1:64 clock divide

100 = 1:32 clock divide

011 = 1:16 clock divide

010 = 1:4 clock divide

001 = 1:2 clock divide 000 = 1:1 clock divide

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

Note:

# 18.0 SERIAL PERIPHERAL INTERFACE (SPI)

Note 1: This data sheet summarizes the features of the dsPIC33FJXXXMCX06A/X08A/X10A family of devices. However, it is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to Section 18. "Serial Peripheral Interface (SPI)" (DS70206) in the "dsPIC33F/PIC24H Family Reference Manual", which is available from the Microchip web site (www.microchip.com).

2: Some registers and associated bits described in this section may not be available on all devices. Refer to Section 4.0 "Memory Organization" in this data sheet for device-specific register and bit information.

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, ADC, etc. The SPI module is compatible with SPI and SIOP from Motorola<sup>®</sup>.

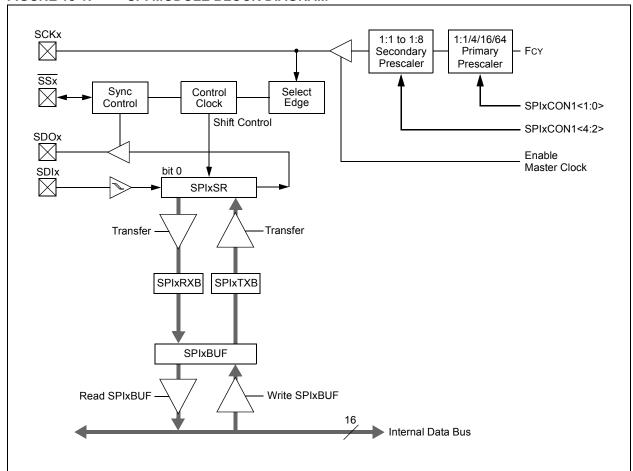
In this section, the SPI modules are referred to together as SPIx, or separately as SPI1 and SPI2. Special Function Registers will follow a similar notation. For example, SPIxCON refers to the control register for the SPI1 or SPI2 module.

Each SPI module consists of a 16-bit shift register, SPIxSR (where x = 1 or 2), used for shifting data in and out, and a buffer register, SPIxBUF. A control register, SPIxCON, configures the module. Additionally, a status register, SPIxSTAT, indicates various status conditions.

The serial interface consists of 4 pins: SDIx (Serial Data Input), SDOx (Serial Data Output), SCKx (Shift Clock Input or Output) and SSx (Active-Low Slave Select).

In Master mode operation, SCK is a clock output, but in Slave mode, it is a clock input.

FIGURE 18-1: SPI MODULE BLOCK DIAGRAM



### 18.1 SPI Helpful Tips

- In Frame mode, if there is a possibility that the master may not be initialized before the slave:
  - a) If FRMPOL (SPIxCON2<13>) = 1, use a pull-down resistor on SSx.
  - b) If FRMPOL = 0, use a pull-up resistor on  $\frac{1}{SSx}$ .

**Note:** This insures that the first frame transmission after initialization is not shifted or corrupted.

- 2. In non-framed 3-wire mode, (i.e., not using  $\overline{SSx}$  from a master):
  - a) If CKP (SPIxCON1<6>) = 1, always place a pull-up resistor on SSx.
  - b) If CKP = 0, always place a pull-down resistor on  $\overline{SSx}$ .

Note: This will insure that during power-up and initialization the master/slave will not lose sync due to an errant SCK transition that would cause the slave to accumulate data shift errors for both transmit and receive appearing as corrupted data.

3. FRMEN (SPIxCON2<15>) = 1 and SSEN (SPIxCON1<7>) = 1 are exclusive and invalid. In Frame mode, SCKx is continuous and the Frame sync pulse is active on the SSx pin, which indicates the start of a data frame.

Note: Not all third-party devices support Frame mode timing. Refer to the SPI electrical characteristics for details.

- 4. In Master mode only, set the SMP bit (SPIxCON1<9>) to a '1' for the fastest SPI data rate possible. The SMP bit can only be set at the same time or after the MSTEN bit (SPIxCON1<5>) is set.
- 5. To avoid invalid slave read data to the master, the user's master software must guarantee enough time for slave software to fill its write buffer before the user application initiates a master write/read cycle. It is always advisable to preload the SPIxBUF transmit register in advance of the next master transaction cycle. SPIxBUF is transferred to the SPI shift register and is empty once the data transmission begins.

### 18.2 SPI Resources

Many useful resources related to SPI are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page, which can be accessed using this link, contains the latest updates and additional information.

Note: In the event you are not able to access the product page using the link above, enter this URL in your browser:

http://www.microchip.com/wwwproducts/
Devices.aspx?dDocName=en546066

### 18.2.1 KEY RESOURCES

- Section 18. "Serial Peripheral Interface (SPI)" (DS70206)
- · Code Samples
- · Application Notes
- · Software Libraries
- Webinars
- All related dsPIC33F/PIC24H Family Reference Manuals Sections
- · Development Tools

# 18.3 SPI Control Registers

# REGISTER 18-1: SPIXSTAT: SPIX STATUS AND CONTROL REGISTER

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
SPIEN	_	SPISIDL	_	_	_	_	_
bit 15							bit 8

U-0	R/C-0	U-0	U-0	U-0	U-0	R-0	R-0
_	SPIROV	_	-	_	_	SPITBF	SPIRBF
bit 7							bit 0

Legend:C = 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 SPIEN: SPIx Enable bit

1 = Enables module and configures SCKx, SDOx, SDIx and SSx as serial port pins

0 = Disables module

bit 14 **Unimplemented:** Read as '0' bit 13 **SPISIDL:** Stop in Idle Mode bit

1 = Discontinue module operation when device enters Idle mode

0 = Continue module operation in Idle mode

bit 12-7 Unimplemented: Read as '0'

bit 6 SPIROV: Receive Overflow Flag bit

1 = A new byte/word is completely received and discarded. The user software has not read the previous data in the SPIxBUF register.

0 = No overflow has occurred

bit 5-2 **Unimplemented:** Read as '0'

bit 1 SPITBF: SPIx Transmit Buffer Full Status bit

1 = Transmit not yet started; SPIxTXB is full

0 = Transmit started; SPIxTXB is empty

Automatically set in hardware when CPU writes SPIxBUF location, loading SPIxTXB. Automatically cleared in hardware when SPIx module transfers data from SPIxTXB to SPIxSR.

bit 0 SPIRBF: SPIx Receive Buffer Full Status bit

1 = Receive complete; SPIxRXB is full

0 = Receive is not complete; SPIxRXB is empty

Automatically set in hardware when SPIx transfers data from SPIxSR to SPIxRXB. Automatically

cleared in hardware when core reads SPIxBUF location, reading SPIxRXB.

### REGISTER 18-2: SPIXCON1: SPIX CONTROL REGISTER 1

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
_	_	_	DISSCK	DISSDO	MODE16	SMP	CKE <sup>(1)</sup>
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 <sup>(3)</sup>	CKP	MSTEN		SPRE<2:0>(2	2)	PPRE<	<1:0> <sup>(2)</sup>
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 DISSCK: Disable SCKx Pin bit (SPI Master modes only)

1 = Internal SPI clock is disabled; pin functions as I/O

0 = Internal SPI clock is enabled

bit 11 DISSDO: Disable SDOx Pin bit

1 = SDOx pin is not used by module; pin functions as I/O

0 = SDOx pin is controlled by the module

bit 10 MODE16: Word/Byte Communication Select bit

1 = Communication is word-wide (16 bits)

0 = Communication is byte-wide (8 bits)

bit 9 SMP: SPIx Data Input Sample Phase bit

Master mode:

1 = Input data sampled at end of data output time

0 = Input data sampled at middle of data output time

Slave mode:

SMP must be cleared when SPIx is used in Slave mode.

bit 8 **CKE:** SPIx Clock Edge Select bit<sup>(1)</sup>

1 = Serial output data changes on transition from active clock state to Idle clock state (see bit 6)

0 = Serial output data changes on transition from Idle clock state to active clock state (see bit 6)

bit 7 SSEN: Slave Select Enable bit (Slave mode)(3)

 $1 = \overline{SSx}$  pin used for Slave mode

0 = SSx pin not used by module. Pin controlled by port function.

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

**Note 1:** The CKE bit is not used in the Framed SPI modes. The user should program this bit to '0' for the Framed SPI modes (FRMEN = 1).

2: Do not set both the primary and secondary prescalers to a value of 1:1.

3: This bit must be cleared when FRMEN = 1.

# REGISTER 18-2: SPIXCON1: SPIX CONTROL REGISTER 1 (CONTINUED)

```
bit 4-2

SPRE<2:0>: Secondary Prescale bits (Master mode)<sup>(2)</sup>

111 = Secondary prescale 1:1

110 = Secondary prescale 2:1

000 = Secondary prescale 8:1

bit 1-0

PPRE<1:0>: Primary Prescale bits (Master mode)<sup>(2)</sup>

11 = Primary prescale 1:1

10 = Primary prescale 4:1

01 = Primary prescale 16:1

00 = Primary prescale 64:1
```

- **Note 1:** The CKE bit is not used in the Framed SPI modes. The user should program this bit to '0' for the Framed SPI modes (FRMEN = 1).
  - 2: Do not set both the primary and secondary prescalers to a value of 1:1.
  - 3: This bit must be cleared when FRMEN = 1.

### REGISTER 18-3: SPIxCON2: SPIx CONTROL REGISTER 2

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0
FRMEN	SPIFSD	FRMPOL	_	_	_	_	_
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	U-0
_	_	_	_	_	_	FRMDLY	_
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 FRMEN: Framed SPIx Support bit

1 = Framed SPIx support enabled ( $\overline{SSx}$  pin used as frame Sync pulse input/output)

0 = Framed SPIx support disabled

bit 14 SPIFSD: Frame Sync Pulse Direction Control bit

1 = Frame Sync pulse input (slave)0 = Frame Sync pulse output (master)

bit 13 FRMPOL: Frame Sync Pulse Polarity bit

1 = Frame Sync pulse is active-high 0 = Frame Sync pulse is active-low

bit 12-2 Unimplemented: Read as '0'

bit 1 FRMDLY: Frame Sync Pulse Edge Select bit

1 = Frame Sync pulse coincides with first bit clock0 = Frame Sync pulse precedes first bit clock

bit 0 **Unimplemented:** This bit must not be set to '1' by the user application.

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

# Note 1: This data sheet summarizes the features of the dsPIC33FJXXXMCX06A/X08A/X10A family of devices. However, it is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to Section 19. "Inter-Integrated Circuit (I²C™)" (DS70195) in the "dsPIC33F/PIC24H Family Reference Manual", which is available from the Microchip web site (www.microchip.com).

2: Some registers and associated bits described in this section may not be available on all devices. Refer to Section 4.0 "Memory Organization" in this data sheet for device-specific register and bit information.

The Inter-Integrated Circuit (I<sup>2</sup>C) module, with its 16-bit interface, provides complete hardware support for both Slave and Multi-Master modes of the I<sup>2</sup>C serial communication standard.

The dsPIC33FJXXXMCX06A/X08A/X10A devices have up to two  $I^2C$  interface modules, denoted as I2C1 and I2C2. Each  $I^2C$  module has a 2-pin interface: the SCLx pin is clock and the SDAx pin is data.

Each  $I^2C$  module 'x' (x = 1 or 2) offers the following key features:

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

### 19.1 Operating Modes

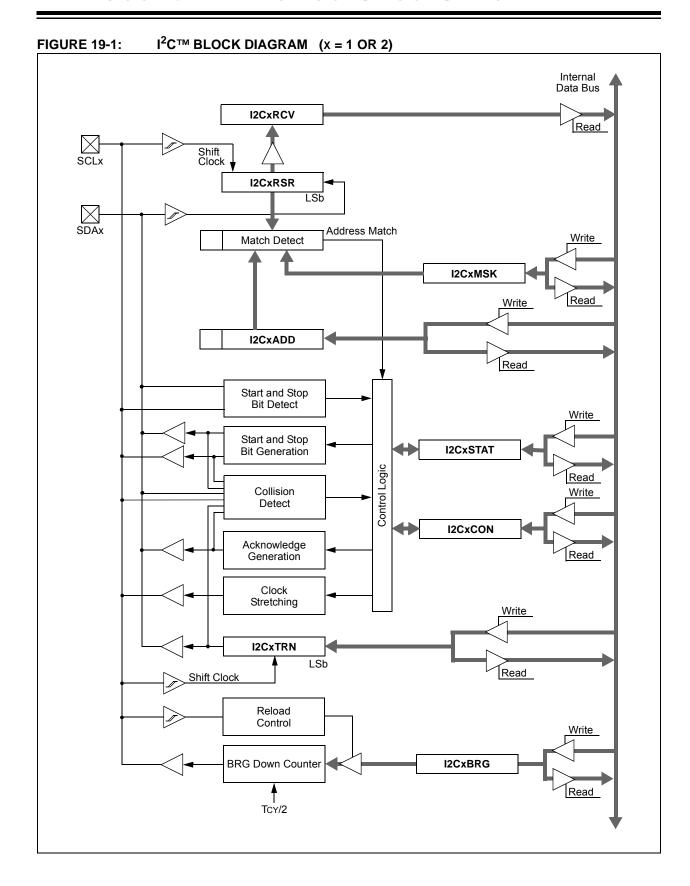
The hardware fully implements all the master and slave functions of the I<sup>2</sup>C Standard and Fast mode specifications, as well as 7 and 10-bit addressing.

The I<sup>2</sup>C module can operate either as a slave or a master on an I<sup>2</sup>C bus.

The following types of I<sup>2</sup>C operation are supported:

- I<sup>2</sup>C slave operation with 7-bit addressing
- I<sup>2</sup>C slave operation with 10-bit addressing
- I<sup>2</sup>C master operation with 7-bit or 10-bit addressing

For details about the communication sequence in each of these modes, please refer to the "dsPIC33F/PIC24H Family Reference Manual".



### 19.2 <sup>2</sup>C Resources

Many useful resources related to I<sup>2</sup>C are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page, which can be accessed using this link, contains the latest updates and additional information.

Note:

In the event you are not able to access the product page using the link above, enter this URL in your browser:

http://www.microchip.com/wwwproducts/ Devices.aspx?dDocName=en546066

### 19.2.1 KEY RESOURCES

- Section 11. "Inter-Integrated Circuit™ (I<sup>2</sup>C™)" (DS70195)
- · Code Samples
- Application Notes
- · Software Libraries
- Webinars
- All related dsPIC33F/PIC24H Family Reference Manuals Sections
- · Development Tools

# 19.3 I<sup>2</sup>C Control Registers

I2CxCON and I2CxSTAT are control and status registers, respectively. The I2CxCON register is readable and writable. The lower six bits of I2CxSTAT are read-only. The remaining bits of the I2CSTAT are read/write.

I2CxRSR is the shift register used for shifting data, whereas I2CxRCV is the buffer register to which data bytes are written, or from which data bytes are read. I2CxRCV is the receive buffer. I2CxTRN is the transmit register to which bytes are written during a transmit operation.

The I2CxADD register holds the slave address. A status bit, ADD10, indicates 10-bit Address mode. The I2CxBRG acts as the Baud Rate Generator (BRG) reload value.

In receive operations, I2CxRSR and I2CxRCV together form a double-buffered receiver. When I2CxRSR receives a complete byte, it is transferred to I2CxRCV and an interrupt pulse is generated.

### REGISTER 19-1: I2CxCON: I2Cx CONTROL REGISTER

R/W-0	U-0	R/W-0	R/W-1, HC	R/W-0	R/W-0	R/W-0	R/W-0
I2CEN	_	I2CSIDL	SCLREL	IPMIEN	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:	U = Unimplemented bit, re	U = Unimplemented bit, read as '0'						
R = Readable bit	W = Writable bit	HS = Hardware Settable bit	HC = Hardware Clearable bit					
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown					

bit 15 I2CEN: I2Cx Enable bit

1 = Enables the I2Cx module and configures the SDAx and SCLx pins as serial port pins

0 = Disables the I2Cx module. All I<sup>2</sup>C<sup>™</sup> pins are controlled by port functions.

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

bit 13 I2CSIDL: Stop in Idle Mode bit

1 = Discontinue module operation when device enters an Idle mode

0 = Continue module operation in Idle mode

bit 12 **SCLREL:** SCLx Release Control bit (when operating as I<sup>2</sup>C slave)

1 = Release SCLx clock

0 = Hold SCLx clock low (clock stretch)

If STREN = 1:

Bit is R/W (i.e., software may write '0' to initiate stretch and write '1' to release clock). Hardware clear at beginning of slave transmission. Hardware clear at end of slave reception.

If STREN = 0:

Bit is R/S (i.e., software may only write '1' to release clock). Hardware clear at beginning of slave transmission.

bit 11 IPMIEN: Intelligent Peripheral Management Interface (IPMI) Enable bit

1 = IPMI mode is enabled; all addresses Acknowledged

0 = IPMI mode disabled

bit 10 A10M: 10-Bit Slave Address bit

1 = I2CxADD is a 10-bit slave address 0 = I2CxADD is a 7-bit slave address

bit 9 DISSLW: Disable Slew Rate Control bit

1 = Slew rate control disabled

0 = Slew rate control enabled

bit 8 SMEN: SMBus Input Levels bit

1 = Enable I/O pin thresholds compliant with SMBus specification

0 = Disable SMBus input thresholds

bit 7 **GCEN:** General Call Enable bit (when operating as I<sup>2</sup>C slave)

1 = Enable interrupt when a general call address is received in the I2CxRSR (module is enabled for reception)

0 = General call address disabled

bit 6 STREN: SCLx Clock Stretch Enable bit (when operating as I<sup>2</sup>C slave)

Used in conjunction with the SCLREL bit.

1 = Enable software or receive clock stretching

0 = Disable software or receive clock stretching

### REGISTER 19-1: I2CxCON: I2Cx CONTROL REGISTER (CONTINUED)

bit 5 **ACKDT:** Acknowledge Data bit (when operating as I<sup>2</sup>C master, applicable during master receive)

Value that will be transmitted when the software initiates an Acknowledge sequence.

- 1 = Send NACK during Acknowledge
- 0 = Send ACK during Acknowledge
- bit 4 ACKEN: Acknowledge Sequence Enable bit

(when operating as I<sup>2</sup>C master, applicable during master receive)

- 1 = Initiate Acknowledge sequence on SDAx and SCLx pins and transmit ACKDT data bit. Hardware clear at end of master Acknowledge sequence
- 0 = Acknowledge sequence not in progress
- bit 3 **RCEN:** Receive Enable bit (when operating as I<sup>2</sup>C master)
  - 1 = Enables Receive mode for I<sup>2</sup>C. Hardware clear at end of eighth bit of master receive data byte
  - 0 = Receive sequence not in progress
- bit 2 **PEN**: Stop Condition Enable bit (when operating as I<sup>2</sup>C master)
  - 1 = Initiate Stop condition on SDAx and SCLx pins. Hardware clear at end of master Stop sequence
  - 0 = Stop condition not in progress
- bit 1 **RSEN**: Repeated Start Condition Enable bit (when operating as I<sup>2</sup>C master)
  - 1 = Initiate Repeated Start condition on SDAx and SCLx pins. Hardware clear at end of master Repeated Start sequence
  - 0 = Repeated Start condition not in progress
- bit 0 **SEN:** Start Condition Enable bit (when operating as I<sup>2</sup>C master)
  - 1 = Initiate Start condition on SDAx and SCLx pins. Hardware clear at end of master Start sequence
  - 0 = Start condition not in progress

### REGISTER 19-2: I2CxSTAT: I2Cx STATUS REGISTER

R-0, HSC	R-0, HSC	U-0	U-0	U-0	R/C-0, HS	R-0, HSC	R-0, HSC
ACKSTAT	TRSTAT	_	_	_	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:	U = Unimplemented bit, read as '0'		C = Clearable bit
R = Readable bit	W = Writable bit	HS = Hardware Settable bit	HSC = Hardware Settable/Clearable bit
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15 ACKSTAT: Acknowledge Status bit

(when operating as I<sup>2</sup>C<sup>™</sup> master, applicable to master transmit operation)

1 = NACK received from slave

0 = ACK received from slave

Hardware set or clear at end of slave Acknowledge.

bit 14 TRSTAT: Transmit Status bit

(when operating as I<sup>2</sup>C master, applicable to master transmit operation)

1 = Master transmit is in progress (8 bits + ACK)

0 = Master transmit is not in progress

Hardware set at beginning of master transmission. Hardware clear at end of slave Acknowledge.

bit 13-11 Unimplemented: Read as '0'

bit 10 BCL: Master Bus Collision Detect bit

1 = A bus collision has been detected during a master operation

0 = No collision

Hardware set at detection of bus collision.

bit 9 GCSTAT: General Call Status bit

1 = General call address was received

0 = General call address was not received

Hardware set when address matches general call address. Hardware clear at Stop detection.

bit 8 ADD10: 10-Bit Address Status bit

1 = 10-bit address was matched

0 = 10-bit address was not matched

Hardware set at match of 2nd byte of matched 10-bit address. Hardware clear at Stop detection.

bit 7 **IWCOL:** Write Collision Detect bit

1 = An attempt to write the I2CxTRN register failed because the I<sup>2</sup>C module is busy

0 = No collision

Hardware set at occurrence of write to I2CxTRN while busy (cleared by software).

bit 6 I2COV: Receive Overflow Flag bit

1 = A byte was received while the I2CxRCV register is still holding the previous byte

0 = No overflow

Hardware set at attempt to transfer I2CxRSR to I2CxRCV (cleared by software).

bit 5 **D\_A:** Data/Address bit (when operating as I<sup>2</sup>C slave)

1 = Indicates that the last byte received was data

0 = Indicates that the last byte received was a device address

Hardware clear at device address match. Hardware set by reception of slave byte.

### REGISTER 19-2: I2CxSTAT: I2Cx STATUS REGISTER (CONTINUED)

bit 4 **P:** Stop bit

1 = Indicates that a Stop bit has been detected last

0 = Stop bit was not detected last

Hardware set or clear when Start, Repeated Start or Stop detected.

bit 3 S: Start bit

1 = Indicates that a Start (or Repeated Start) bit has been detected last

0 = Start bit was not detected last

Hardware set or clear when Start, Repeated Start or Stop detected.

bit 2  $R_W$ : Read/Write Information bit (when operating as  $I^2$ C slave)

1 = Read – indicates data transfer is output from slave

0 = Write – indicates data transfer is input to slave

Hardware set or clear after reception of I<sup>2</sup>C device address byte.

bit 1 RBF: Receive Buffer Full Status bit

1 = Receive complete; I2CxRCV is full

0 = Receive not complete; I2CxRCV is empty

Hardware set when I2CxRCV is written with received byte. Hardware clear when software reads I2CxRCV.

bit 0 TBF: Transmit Buffer Full Status bit

1 = Transmit in progress, I2CxTRN is full

0 = Transmit complete, I2CxTRN is empty

Hardware set when software writes I2CxTRN. Hardware clear at completion of data transmission.

### REGISTER 19-3: 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
_	_	_	_	_	_	AMSK9	AMSK8
bit 15							bit 8

| R/W-0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| AMSK7 | AMSK6 | AMSK5 | AMSK4 | AMSK3 | AMSK2 | AMSK1 | AMSK0 |
| 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 AMSKx: Mask for Address bit x Select bits

1 = Enable masking for bit x of incoming message address; bit match not required in this position

0 = Disable masking for bit x; bit match required in this position

# 20.0 UNIVERSAL ASYNCHRONOUS RECEIVER TRANSMITTER (UART)

Note 1: This data sheet summarizes the features of the dsPIC33FJXXXMCX06A/X08A/X10A family of devices. However, it is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to Section 17. "UART" (DS70188) in the "dsPIC33F/PIC24H Family Reference Manual", which is available from the Microchip web site (www.microchip.com).

2: Some registers and associated bits described in this section may not be available on all devices. Refer to Section 4.0 "Memory Organization" in this data sheet for device-specific register and bit information.

The Universal Asynchronous Receiver Transmitter (UART) module is one of the serial I/O modules available in the dsPIC33FJXXXMCX06A/X08A/X10A 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 and also includes an IrDA® encoder and decoder.

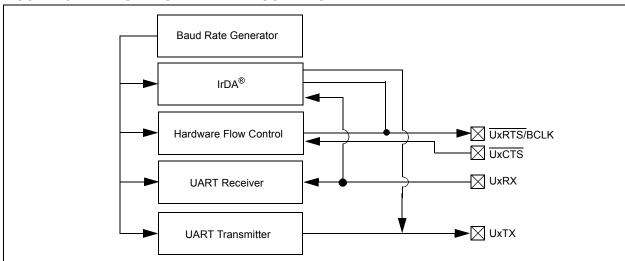
The primary features of the UART module are:

- Full-Duplex, 8-bit 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 UxCTS and UxRTS Pins
- Fully Integrated Baud Rate Generator with 16-Bit Prescaler
- Baud Rates Ranging from 10 Mbps to 38 bps at 40 MIPS
- 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)
- · Transmit and Receive Interrupts
- · A Separate Interrupt for all UART Error Conditions
- Loopback mode for Diagnostic Support
- · Support for Sync and Break Characters
- · Supports Automatic Baud Rate Detection
- · IrDA Encoder and Decoder Logic
- 16x Baud Clock Output for IrDA Support

A simplified block diagram of the UART is shown in Figure 20-1. The UART module consists of these key important hardware elements:

- Baud Rate Generator
- · Asynchronous Transmitter
- · Asynchronous Receiver

### FIGURE 20-1: UART SIMPLIFIED BLOCK DIAGRAM



- **Note 1:** Both UART1 and UART2 can trigger a DMA data transfer. If U1TX, U1RX, U2TX or U2RX is selected as a DMA IRQ source, a DMA transfer occurs when the U1TXIF, U1RXIF, U2TXIF or U2RXIF bit gets set as a result of a UART1 or UART2 transmission or reception.
  - 2: If DMA transfers are required, the UART TX/RX FIFO buffer must be set to a size of 1 byte/word (i.e., UTXISEL<1:0> = 00 and URXISEL<1:0> = 00).

### 20.1 UART Helpful Tips

- 1. In multi-node direct-connect UART networks, receive inputs react complementary logic level defined by the URXINV bit (UxMODE<4>), which defines the idle state, the default of which is logic high, (i.e., URXINV = 0). Because remote devices do not initialize at the same time, it is likely that one of the devices, because the RX line is floating, will trigger a start bit detection and will cause the first byte received after the device has been initialized to be invalid. To avoid this situation, the user should use a pull-up or pull-down resistor on the RX pin depending on the value of the URXINV bit.
  - a) If URXINV = 0, use a pull-up resistor on the RX pin.
  - b) If URXINV = 1, use a pull-down resistor on the RX pin.
- 2. The first character received on a wake-up from Sleep mode caused by activity on the UxRX pin of the UART module will be invalid. In Sleep mode, peripheral clocks are disabled. By the time the oscillator system has restarted and stabilized from Sleep mode, the baud rate bit sampling clock relative to the incoming UxRX bit timing is no longer synchronized, resulting in the first character being invalid. This is to be expected.

### 20.2 UART Resources

Many useful resources related to UART are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page, which can be accessed using this link, contains the latest updates and additional information.

Note:

In the event you are not able to access the product page using the link above, enter this URL in your browser:

http://www.microchip.com/wwwproducts/ Devices.aspx?dDocName=en546066

### 20.2.1 KEY RESOURCES

- Section 17. "UART" (DS70188)
- · Code Samples
- · Application Notes
- · Software Libraries
- Webinars
- All related dsPIC33F/PIC24H Family Reference Manuals Sections
- · Development Tools

# 20.3 UART Control Registers

### REGISTER 20-1: 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 <sup>(1)</sup>	_	USIDL	IREN <sup>(2)</sup>	RTSMD	_	UEN-	<1:0>
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	PDSEL	.<1:0>	STSEL
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 **UARTEN:** UARTx Enable bit<sup>(1)</sup>

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: Stop in Idle Mode bit

1 = Discontinue module operation when device enters Idle mode.

0 = Continue module operation in Idle mode

bit 12 IREN: IrDA® Encoder and Decoder Enable bit(2)

1 = IrDA encoder and decoder enabled0 = IrDA encoder and decoder disabled

bit 11 RTSMD: Mode Selection for UxRTS Pin bit

1 =  $\overline{\text{UxRTS}}$  pin in Simplex mode

 $0 = \overline{\text{UxRTS}}$  pin in Flow Control mode

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

bit 9-8 **UEN<1:0>:** UARTx Enable bits

11 = UxTX, UxRX and BCLK pins are enabled and used; UxCTS pin controlled by port latches

10 = UxTX, UxRX,  $\overline{\text{UxCTS}}$  and  $\overline{\text{UxRTS}}$  pins are enabled and used

01 = UxTX, UxRX and  $\overline{\text{UxRTS}}$  pins are enabled and used;  $\overline{\text{UxCTS}}$  pin controlled by port latches

00 = UxTX and UxRX pins are enabled and used and UxRTS/BCLK pins controlled by port latches

bit 7 WAKE: Wake-up on Start bit Detect During Sleep Mode Enable bit

1 = UARTx will continue to sample the UxRX pin. Interrupt generated on the falling edge; bit cleared in hardware on the following rising edge.

0 = No wake-up enabled

bit 6 LPBACK: UARTx Loopback Mode Select bit

1 = Enable Loopback mode

0 = Loopback mode is disabled

bit 5 ABAUD: Auto-Baud Enable bit

1 = Enable baud rate measurement on the next character – requires reception of a Sync field (0x55) before other data; cleared in hardware upon completion

0 = Baud rate measurement disabled or completed

**Note 1:** Refer to **Section 17. "UART"** (DS70188) in the "dsPIC33F/PIC24H Family Reference Manual" for information on enabling the UART module for receive or transmit operation.

2: This feature is only available for the 16x BRG mode (BRGH = 0).

### REGISTER 20-1: UXMODE: UARTX MODE REGISTER (CONTINUED)

bit 4 URXINV: 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 = BRG generates 4 clocks per bit period (4x baud clock, High-Speed mode)0 = BRG generates 16 clocks per bit period (16x baud clock, Standard mode)

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 bits0 = One Stop bit

**Note 1:** Refer to **Section 17. "UART"** (DS70188) in the "dsPIC33F/PIC24H Family Reference Manual" for information on enabling the UART module for receive or transmit operation.

2: This feature is only available for the  $16x BRG \mod (BRGH = 0)$ .

### REGISTER 20-2: UxSTA: UARTx STATUS AND CONTROL REGISTER

R/W-0	R/W-0	R/W-0	U-0	R/W-0 HC	R/W-0	R-0	R-1
UTXISEL1	UTXINV	UTXISEL0	_	UTXBRK	UTXEN <sup>(1)</sup>	UTXBF	TRMT
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R-1	R-0	R-0	R/C-0	R-0
URXISE	L<1:0>	ADDEN	RIDLE	PERR	FERR	OERR	URXDA
bit 7		•					bit 0

Legend:	HC = Hardware Clearable bit		C = Clearable bit	
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 15,13 UTXISEL<1:0>: 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: Transmit Polarity Inversion bit

If IREN = 0:

1 = UxTX Idle state is '0'

0 = UxTX Idle state is '1'

### If IREN = 1:

- 1 = IrDA® encoded UxTX Idle state is '1'
- 0 = IrDA encoded UxTX Idle state is '0'
- bit 12 **Unimplemented:** Read as '0'
- bit 11 UTXBRK: Transmit Break bit
  - 1 = Send 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 disabled or completed
- bit 10 **UTXEN:** Transmit Enable bit<sup>(1)</sup>
  - 1 = Transmit enabled, UxTX pin controlled by UARTx
  - 0 = Transmit disabled, any pending transmission is aborted and the buffer is reset. UxTX pin controlled by port.
- bit 9 **UTXBF:** 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 the 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:** Refer to **Section 17. "UART"** (DS70188) in the "dsPIC33F/PIC24H Family Reference Manual" for information on enabling the UART module for transmit operation.

# REGISTER 20-2: UxSTA: UARTx STATUS AND CONTROL REGISTER (CONTINUED)

- bit 7-6 URXISEL<1:0>: Receive Interrupt Mode Selection bits
  - 11 = Interrupt is set on the UxRSR transfer, making the receive buffer full (i.e., has 4 data characters)
  - 10 = Interrupt is set on the UxRSR 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 UxRSR 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 enabled. If 9-bit mode is not selected, this does not take effect.
  - 0 = Address Detect mode 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 (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 (character at the top of the receive FIFO)
  - 0 = Framing error has not been detected
- bit 1 OERR: Receive Buffer Overrun Error Status bit (read/clear 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 receiver buffer and the UxRSR to the empty state.
- bit 0 **URXDA:** 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:** Refer to **Section 17. "UART"** (DS70188) in the "dsPIC33F/PIC24H Family Reference Manual" for information on enabling the UART module for transmit operation.

### 21.0 ENHANCED CAN MODULE

Note 1: This data sheet summarizes the features of the dsPIC33FJXXXMCX06A/X08A/X10A family of devices. However, it is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to Section 15. "Enhanced Controller Area Network (ECAN™)" (DS70185) in the "dsPIC33F/PIC24H Family Reference Manual", which is available from the Microchip web site (www.microchip.com).

2: Some registers and associated bits described in this section may not be available on all devices. Refer to Section 4.0 "Memory Organization" in this data sheet for device-specific register and bit information.

### 21.1 Overview

The Enhanced Controller Area Network (ECAN™ technology) module is a serial interface, useful for communicating with other CAN modules or microcontroller devices. This interface/protocol was designed to allow communications within noisy environments. The dsPIC33FJXXXMCX06A/X08A/X10A devices contain up to two ECAN modules.

The CAN module is a communication controller implementing the CAN 2.0 A/B protocol, as defined in the BOSCH specification. The module will support CAN 1.2, CAN 2.0A, CAN 2.0B Passive and CAN 2.0B Active versions of the protocol. The module implementation is a full CAN system. The CAN specification is not covered within this data sheet. The reader may refer to the BOSCH CAN specification for further details.

The module features are as follows:

- Implementation of the CAN protocol, CAN 1.2, CAN 2.0A and CAN 2.0B
- · Standard and extended data frames
- · 0-8 bytes data length
- · Programmable bit rate up to 1 Mbit/sec
- Automatic response to remote transmission requests
- Up to eight transmit buffers with application specified prioritization and abort capability (each buffer may contain up to 8 bytes of data)
- Up to 32 receive buffers (each buffer may contain up to 8 bytes of data)
- Up to 16 full (standard/extended identifier) acceptance filters
- · Three full acceptance filter masks
- DeviceNet<sup>™</sup> addressing support
- Programmable wake-up functionality with integrated low-pass filter
- Programmable Loopback mode supports self-test operation

- Signaling via interrupt capabilities for all CAN receiver and transmitter error states
- · Programmable clock source
- Programmable link to input capture module (IC2 for both CAN1 and CAN2) for time-stamping and network synchronization
- · Low-power Sleep and Idle mode

The CAN bus module consists of a protocol engine and message buffering/control. The CAN protocol engine handles all functions for receiving and transmitting messages on the CAN bus. Messages are transmitted by first loading the appropriate data registers. Status and errors can be checked by reading the appropriate registers. Any message detected on the CAN bus is checked for errors and then matched against filters to see if it should be received and stored in one of the receive registers.

### 21.2 Frame Types

The CAN module transmits various types of frames which include data messages, or remote transmission requests initiated by the user, as other frames that are automatically generated for control purposes. The following frame types are supported:

· Standard Data Frame:

A standard data frame is generated by a node when the node wishes to transmit data. It includes an 11-bit Standard Identifier (SID), but not an 18-bit Extended Identifier (EID).

· Extended Data Frame:

An extended data frame is similar to a standard data frame, but includes an extended identifier as well.

· Remote Frame:

It is possible for a destination node to request the data from the source. For this purpose, the destination node sends a remote frame with an identifier that matches the identifier of the required data frame. The appropriate data source node will then send a data frame as a response to this remote request.

· Error Frame:

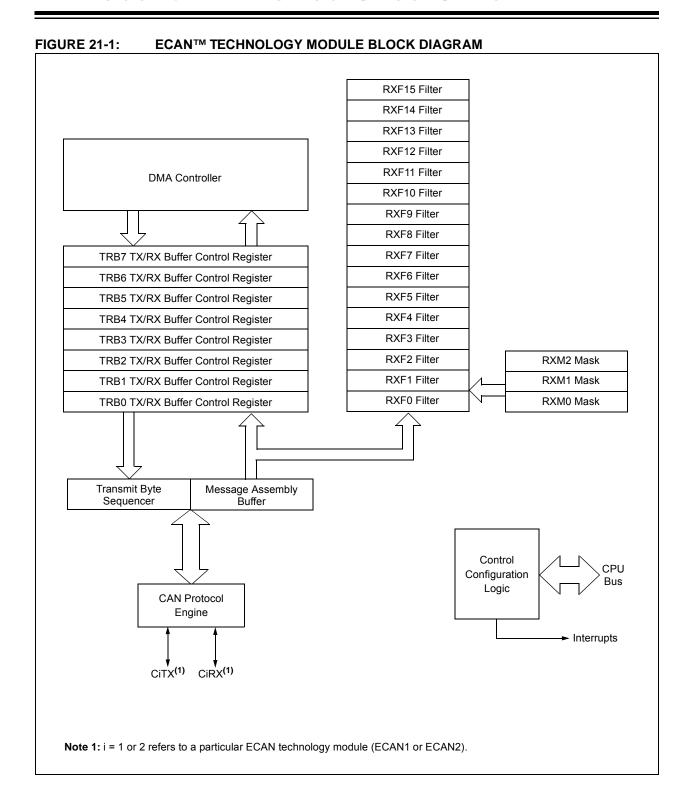
An error frame is generated by any node that detects a bus error. An error frame consists of two fields: an error flag field and an error delimiter field.

Overload Frame:

An overload frame can be generated by a node as a result of two conditions. First, the node detects a dominant bit during interframe space which is an illegal condition. Second, due to internal conditions, the node is not yet able to start reception of the next message. A node may generate a maximum of 2 sequential overload frames to delay the start of the next message.

Interframe Space:

Interframe space separates a proceeding frame (of whatever type) from a following data or remote frame.



### 21.3 Modes of Operation

The CAN module can operate in one of several operation modes selected by the user. These modes include:

- · Initialization Mode
- · Disable Mode
- · Normal Operation Mode
- · Listen Only Mode
- · Listen All Messages Mode
- · Loopback Mode

Modes are requested by setting the REQOP<2:0> bits (CiCTRL1<10:8>). Entry into a mode is Acknowledged by monitoring the OPMODE<2:0> bits (CiCTRL1<7:5>). The module will not change the mode and the OPMODE bits until a change in mode is acceptable, generally during bus Idle time, which is defined as at least 11 consecutive recessive bits.

### 21.3.1 INITIALIZATION MODE

In the Initialization mode, the module will not transmit or receive. The error counters are cleared and the interrupt flags remain unchanged. The programmer will have access to Configuration registers that are access restricted in other modes. The module will protect the user from accidentally violating the CAN protocol through programming errors. All registers which control the configuration of the module cannot be modified while the module is on-line. The CAN module will not be allowed to enter the Configuration mode while a transmission is taking place. The Configuration mode serves as a lock to protect the following registers:

- All Module Control Registers
- · Baud Rate and Interrupt Configuration Registers
- · Bus Timing Registers
- · Identifier Acceptance Filter Registers
- · Identifier Acceptance Mask Registers

### 21.3.2 DISABLE MODE

In Disable mode, the module will not transmit or receive. The module has the ability to set the WAKIF bit due to bus activity, however, any pending interrupts will remain and the error counters will retain their value.

If the REQOP<2:0> bits (CiCTRL1<10:8>) = 001, the module will enter the Module Disable mode. If the module is active, the module will wait for 11 recessive bits on the CAN bus, detect that condition as an Idle bus, then accept the module disable command. When the OPMODE<2:0> bits (CiCTRL1<7:5>) = 001, that indicates whether the module successfully went into Module Disable mode. The I/O pins will revert to normal I/O function when the module is in the Module Disable mode.

The module can be programmed to apply a low-pass filter function to the CiRX input line while the module or the CPU is in Sleep mode. The WAKFIL bit (CiCFG2<14>) enables or disables the filter.

Note:

Typically, if the CAN module is allowed to transmit in a particular mode of operation and a transmission is requested immediately after the CAN module has been placed in that mode of operation, the module waits for 11 consecutive recessive bits on the bus before starting transmission. If the user switches to Disable mode within this 11-bit period, then this transmission is aborted and the corresponding TXABT bit is set, and the TXREQ bit is cleared.

#### 21.3.3 NORMAL OPERATION MODE

Normal Operation mode is selected when REQOP<2:0> = 000. In this mode, the module is activated and the I/O pins will assume the CAN bus functions. The module will transmit and receive CAN bus messages via the CiTX and CiRX pins.

#### 21.3.4 LISTEN ONLY MODE

If the Listen Only mode is activated, the module on the CAN bus is passive. The transmitter buffers revert to the port I/O function. The receive pins remain inputs. For the receiver, no error flags or Acknowledge signals are sent. The error counters are deactivated in this state. The Listen Only mode can be used for detecting the baud rate on the CAN bus. To use this, it is necessary that there are at least two further nodes that communicate with each other.

### 21.3.5 LISTEN ALL MESSAGES MODE

The module can be set to ignore all errors and receive any message. The Listen All Messages mode is activated by setting REQOP<2:0> = 111. In this mode, the data which is in the message assembly buffer until the time an error occurred, is copied in the receive buffer and can be read via the CPU interface.

#### 21.3.6 LOOPBACK MODE

If the Loopback mode is activated, the module will connect the internal transmit signal to the internal receive signal at the module boundary. The transmit and receive pins revert to their port I/O function.

### REGISTER 21-1: CiCTRL1: ECAN™ CONTROL REGISTER 1

U-0	U-0	R/W-0	R/W-0	r-0	R/W-1	R/W-0	R/W-0
_	_	CSIDL	ABAT	_		REQOP<2:0>	
bit 15							bit 8

R-1	R-0	R-0	U-0	R/W-0	U-0	U-0	R/W-0
	OPMODE<2:0>		_	CANCAP	_	_	WIN
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 15-14 **Unimplemented:** Read as '0'

bit 13 CSIDL: Stop in Idle Mode bit

1 = Discontinue module operation when device enters Idle mode

0 = Continue module operation in Idle mode

bit 12 ABAT: Abort All Pending Transmissions bit

1 = Signal all transmit buffers to abort transmission

0 = Module will clear this bit when all transmissions are aborted

bit 11 Reserved: Do no use

bit 10-8 **REQOP<2:0>:** Request Operation Mode bits

111 = Set Listen All Messages mode

110 = Reserved – do not use

101 = Reserved – do not use

100 = Set Configuration mode

011 = Set Listen Only Mode

010 = Set Loopback mode

001 = Set Disable mode

000 = Set Normal Operation mode

bit 7-5 **OPMODE<2:0>:** Operation Mode bits

111 = Module is in Listen All Messages mode

110 = Reserved

101 = Reserved

100 = Module is in Configuration mode

011 = Module is in Listen Only mode

010 = Module is in Loopback mode

001 = Module is in Disable mode

000 = Module is in Normal Operation mode

bit 4 Unimplemented: Read as '0'

bit 3 CANCAP: CAN Message Receive Timer Capture Event Enable bit

1 = Enable input capture based on CAN message receive

0 = Disable CAN capture

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

bit 0 WIN: SFR Map Window Select bit

1 = Use filter window

0 = Use buffer window

### REGISTER 21-2: CiCTRL2: ECAN™ 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-0	R-0	R-0	R-0
_	_	_			DNCNT<4:0>		
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 15-5 **Unimplemented:** Read as '0'

bit 4-0 **DNCNT<4:0>:** DeviceNet™ Filter Bit Number bits

10010-11111 = Invalid selection

10001 = Compare up to data byte 3, bit 6 with EID<17>

•

•

•

00001 = Compare up to data byte 1, bit 7 with EID<0>

00000 = Do not compare data bytes

### REGISTER 21-3: CIVEC: ECAN™ INTERRUPT CODE REGISTER

U-0	U-0	U-0	R-0	R-0	R-0	R-0	R-0
_	_	_			FILHIT<4:0>		
bit 15							bit 8

U-0	R-1	R-0	R-0	R-0	R-0	R-0	R-0
_				ICODE<6:0>	•		
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 12-8 FILHIT<4:0>: Filter Hit Number bits

10000-11111 = Reserved

01111 = Filter 15

•

•

•

00001 = Filter 1 00000 = Filter 0

bit 7 Unimplemented: Read as '0'

bit 6-0 ICODE<6:0>: Interrupt Flag Code bits

1000101-1111111 = Reserved

1000100 = FIFO almost full interrupt

1000011 = Receiver overflow interrupt

1000010 = Wake-up interrupt

1000001 = Error interrupt

1000000 **= No interrupt** 

0010000-0111111 **= Reserved** 

0001111 = RB15 buffer Interrupt

•

\_ \_ \_ .

0001001 = RB9 buffer interrupt

0001000 = RB8 buffer interrupt

0000111 = TRB7 buffer interrupt

0000110 = TRB6 buffer interrupt

0000101 = TRB5 buffer interrupt

0000100 = TRB4 buffer interrupt 0000011 = TRB3 buffer interrupt

0000010 = TRB2 buffer interrupt

0000001 = TRB1 buffer interrupt

0000001 - TRBT bullet litterrup

0000000 = TRB0 Buffer interrupt

### REGISTER 21-4: CIFCTRL: ECAN™ FIFO CONTROL REGISTER

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0
	DMABS<2:0>		_	_	_	_	
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
_	_	_			FSA<4:0>		
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 DMABS<2:0>: DMA Buffer Size bits

111 = Reserved

110 = 32 buffers in DMA RAM

101 = 24 buffers in DMA RAM

100 = 16 buffers in DMA RAM

011 = 12 buffers in DMA RAM

010 = 8 buffers in DMA RAM

001 = 6 buffers in DMA RAM

000 = 4 buffers in DMA RAM

bit 12-5 Unimplemented: Read as '0'

bit 4-0 FSA<4:0>: FIFO Area Starts with Buffer bits

11111 = RB31 buffer

11110 = RB30 buffer

•

.

00001 = TRB1 buffer

00000 = TRB0 buffer

### REGISTER 21-5: CiFIFO: ECAN™ FIFO STATUS REGISTER

U-0	U-0	R-0	R-0	R-0	R-0	R-0	R-0
_	_			FBP	<5:0>		
bit 15							bit 8

U-0	U-0	R-0	R-0	R-0	R-0	R-0	R-0
_	_			FNR	3<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 15-14 Unimplemented: Read as '0'

bit 13-8 **FBP<5:0>:** FIFO Write Buffer Pointer bits

011111 = RB31 buffer 011110 = RB30 buffer

•

•

000001 = TRB1 buffer 000000 = TRB0 buffer

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

bit 5-0 FNRB<5:0>: FIFO Next Read Buffer Pointer bits

011111 = RB31 buffer 011110 = RB30 buffer

•

•

000001 = TRB1 buffer 000000 = TRB0 buffer

#### **CIINTF: ECAN™ INTERRUPT FLAG REGISTER** REGISTER 21-6:

U-0	U-0	R-0	R-0	R-0	R-0	R-0	R-0
_	_	TXBO	TXBP	RXBP	TXWAR	RXWAR	EWARN
bit 15							bit 8

R/C-0	R/C-0	R/C-0	U-0	R/C-0	R/C-0	R/C-0	R/C-0
IVRIF	WAKIF	ERRIF	_	FIFOIF	RBOVIF	RBIF	TBIF
bit 7							bit 0

C = Clearable 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 15-14 Unimplemented: Read as '0'

bit 13 TXBO: Transmitter in Error State Bus Off bit

> 1 = Transmitter is in Bus Off state 0 = Transmitter is not in Bus Off state

bit 12 TXBP: Transmitter in Error State Bus Passive bit

> 1 = Transmitter is in Bus Passive state 0 = Transmitter is not in Bus Passive state

**RXBP:** Receiver in Error State Bus Passive bit bit 11

> 1 = Receiver is in Bus Passive state 0 = Receiver is not in Bus Passive state

bit 10 **TXWAR:** Transmitter in Error State Warning bit

> 1 = Transmitter is in Error Warning state 0 = Transmitter is not in Error Warning state

bit 9 RXWAR: Receiver in Error State Warning bit

1 = Receiver is in Error Warning state 0 = Receiver is not in Error Warning state

bit 8 **EWARN:** Transmitter or Receiver in Error State Warning bit

> 1 = Transmitter or receiver is in Error Warning state 0 = Transmitter or receiver is not in Error Warning state

IVRIF: Invalid Message Received Interrupt Flag bit bit 7

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

bit 6 WAKIF: Bus Wake-up Activity Interrupt Flag bit

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

bit 5 ERRIF: Error Interrupt Flag bit (multiple sources in CiINTF<13:8> register)

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

bit 4 Unimplemented: Read as '0'

bit 3 FIFOIF: FIFO Almost Full Interrupt Flag bit

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

bit 2 **RBOVIF:** RX Buffer Overflow Interrupt Flag bit

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

bit 1 RBIF: RX Buffer Interrupt Flag bit

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

bit 0 TBIF: TX Buffer Interrupt Flag bit

1 = Interrupt request has occurred

0 = Interrupt request has not occurred

### REGISTER 21-7: CIINTE: ECAN™ 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	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
IVRIE	WAKIE	ERRIE	_	FIFOIE	RBOVIE	RBIE	TBIE
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 IVRIE: Invalid Message Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 6 WAKIE: Bus Wake-up Activity Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 5 ERRIE: Error Interrupt Enable bit

1 = Interrupt request enabled 0 = Interrupt request not enabled Unimplemented: Read as '0'

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

bit 3 FIFOIE: FIFO Almost Full Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 2 RBOVIE: RX Buffer Overflow Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 1 RBIE: RX Buffer Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

bit 0 TBIE: TX Buffer Interrupt Enable bit

1 = Interrupt request enabled0 = Interrupt request not enabled

### REGISTER 21-8: CIEC: ECAN™ TRANSMIT/RECEIVE ERROR COUNT REGISTER

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0			
	TERRCNT<7:0>									
bit 15							bit 8			

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
			RERRC	NT<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 15-8 **TERRCNT<7:0>:** Transmit Error Count bits bit 7-0 **RERRCNT<7:0>:** Receive Error Count bits

### REGISTER 21-9: CICFG1: ECAN™ BAUD RATE CONFIGURATION REGISTER 1

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
SJW	<1:0>			BRP	<sup>2</sup> <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 15-8 **Unimplemented:** Read as '0'

bit 7-6 **SJW<1:0>:** Synchronization Jump Width bits

11 = Length is 4 x TQ 10 = Length is 3 x TQ 01 = Length is 2 x TQ 00 = Length is 1 x TQ

bit 5-0 BRP<5:0>: Baud Rate Prescaler bits

11 1111 = TQ = 2 x 64 x 1/FCAN

•

00 0010 = TQ = 2 x 3 x 1/FCAN 00 0001 = TQ = 2 x 2 x 1/FCAN 00 0000 = TQ = 2 x 1 x 1/FCAN

### REGISTER 21-10: CICFG2: ECAN™ BAUD RATE CONFIGURATION REGISTER 2

U-0	R/W-x	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x
_	WAKFIL	_	_	_	:	SEG2PH<2:0>	
bit 15							bit 8

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
SEG2PHTS	SAM		SEG1PH<2:0>	>		PRSEG<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 WAKFIL: Select CAN bus Line Filter for Wake-up bit

1 = Use CAN bus line filter for wake-up

0 = CAN bus line filter is not used for wake-up

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

bit 10-8 **SEG2PH<2:0>:** Phase Buffer Segment 2 bits

111 = Length is 8 x TQ 000 = Length is 1 x TQ

bit 7 SEG2PHTS: Phase Segment 2 Time Select bit

1 = Freely programmable

0 = Maximum of SEG1PH bits or Information Processing Time (IPT), whichever is greater

bit 6 SAM: Sample of the CAN bus Line bit

1 = Bus line is sampled three times at the sample point0 = Bus line is sampled once at the sample point

\_\_\_\_\_

bit 5-3 **SEG1PH<2:0>:** Phase Buffer Segment 1 bits

111 = Length is 8 x TQ 000 = Length is 1 x TQ

bit 2-0 PRSEG<2:0>: Propagation Time Segment bits

111 = Length is 8 x TQ 000 = Length is 1 x TQ

### REGISTER 21-11: CIFEN1: ECAN™ ACCEPTANCE FILTER ENABLE REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
FLTEN15	FLTEN14	FLTEN13	FLTEN12	FLTEN11	FLTEN10	FLTEN9	FLTEN8
bit 15							bit 8

| R/W-1  |
|--------|--------|--------|--------|--------|--------|--------|--------|
| FLTEN7 | FLTEN6 | FLTEN5 | FLTEN4 | FLTEN3 | FLTEN2 | FLTEN1 | FLTEN0 |
| 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 FLTENn: Enable Filter n to Accept Messages bits

1 = Enable Filter n0 = Disable Filter n

### REGISTER 21-12: CIBUFPNT1: ECAN™ FILTER 0-3 BUFFER POINTER 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	
	F3BP<	<3:0>		F2BP<3:0>				
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		
	F1BP<	<3:0>		F0BP<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 15-12 **F3BP<3:0>:** RX Buffer Written when Filter 3 Hits bits

1111 = Filter hits received in RX FIFO buffer

1110 = Filter hits received in RX Buffer 14

•

0001 = Filter hits received in RX Buffer 1

0000 = Filter hits received in RX Buffer 0

bit 11-8 F2BP<3:0>: RX Buffer Written when Filter 2 Hits bits

1111 = Filter hits received in RX FIFO buffer

1110 = Filter hits received in RX Buffer 14

•

•

•

0001 = Filter hits received in RX Buffer 1

0000 = Filter hits received in RX Buffer 0

bit 7-4 F1BP<3:0>: RX Buffer Written when Filter 1 Hits bits

1111 = Filter hits received in RX FIFO buffer

1110 = Filter hits received in RX Buffer 14

.

.

0001 = Filter hits received in RX Buffer 1

0000 = Filter hits received in RX Buffer 0

bit 3-0 **F0BP<3:0>:** RX Buffer Written when Filter 0 Hits bits

1111 = Filter hits received in RX FIFO buffer 1110 = Filter hits received in RX Buffer 14

.

•

•

0001 = Filter hits received in RX Buffer 1

### REGISTER 21-13: CIBUFPNT2: ECAN™ FILTER 4-7 BUFFER POINTER 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	
	F7BP<	<3:0>		F6BP<3:0>				
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		
	F5BP<	<3:0>		F4BP<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 15-12 F7BP<3:0>: RX Buffer Written when Filter 7 Hits bits

1111 = Filter hits received in RX FIFO buffer

1110 = Filter hits received in RX Buffer 14

0001 = Filter hits received in RX Buffer 1

0000 = Filter hits received in RX Buffer 0

bit 11-8 F6BP<3:0>: RX Buffer Written when Filter 6 Hits bits

1111 = Filter hits received in RX FIFO buffer

1110 = Filter hits received in RX Buffer 14

0001 = Filter hits received in RX Buffer 1 0000 = Filter hits received in RX Buffer 0

F5BP<3:0>: RX Buffer Written when Filter 5 Hits bits bit 7-4

1111 = Filter hits received in RX FIFO buffer

1110 = Filter hits received in RX Buffer 14

0001 = Filter hits received in RX Buffer 1

0000 = Filter hits received in RX Buffer 0

bit 3-0 F4BP<3:0>: RX Buffer Written when Filter 4 Hits bits

> 1111 = Filter hits received in RX FIFO buffer 1110 = Filter hits received in RX Buffer 14

0001 = Filter hits received in RX Buffer 1

### REGISTER 21-14: CIBUFPNT3: ECAN™ FILTER 8-11 BUFFER POINTER 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	
	F11BP	<3:0>		F10BP<3:0>				
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		
	F9BP<	<3:0>		F8BP<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 15-12 **F11BP<3:0>:** RX Buffer Written when Filter 11 Hits bits

1111 = Filter hits received in RX FIFO buffer

1110 = Filter hits received in RX Buffer 14

•

0001 = Filter hits received in RX Buffer 1

0000 = Filter hits received in RX Buffer 0

bit 11-8 F10BP<3:0>: RX Buffer Written when Filter 10 Hits bits

1111 = Filter hits received in RX FIFO buffer

1110 = Filter hits received in RX Buffer 14

•

•

0001 = Filter hits received in RX Buffer 1

0000 = Filter hits received in RX Buffer 0

bit 7-4 F9BP<3:0>: RX Buffer Written when Filter 9 Hits bits

1111 = Filter hits received in RX FIFO buffer

1110 = Filter hits received in RX Buffer 14

.

•

0001 = Filter hits received in RX Buffer 1

0000 = Filter hits received in RX Buffer 0

bit 3-0 **F8BP<3:0>:** RX Buffer Written when Filter 8 Hits bits

1111 = Filter hits received in RX FIFO buffer

1110 = Filter hits received in RX Buffer 14

•

•

0001 = Filter hits received in RX Buffer 1

### REGISTER 21-15: CIBUFPNT4: ECAN™ FILTER 12-15 BUFFER POINTER 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	
	F15BP	<3:0>		F14BP<3:0>				
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	
	F13BP	<3:0>		F12BP<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 15-12 F15BP<3:0>: RX Buffer Written when Filter 15 Hits bits

1111 = Filter hits received in RX FIFO buffer 1110 = Filter hits received in RX Buffer 14

•

•

0001 = Filter hits received in RX Buffer 1

0000 = Filter hits received in RX Buffer 0

bit 11-8 F14BP<3:0>: RX Buffer Written when Filter 14 Hits bits

1111 = Filter hits received in RX FIFO buffer

1110 = Filter hits received in RX Buffer 14

•

•

0001 = Filter hits received in RX Buffer 1

0000 = Filter hits received in RX Buffer 0

bit 7-4 F13BP<3:0>: RX Buffer Written when Filter 13 Hits bits

1111 = Filter hits received in RX FIFO buffer

1110 = Filter hits received in RX Buffer 14

\_

•

0001 = Filter hits received in RX Buffer 1

0000 = Filter hits received in RX Buffer 0

bit 3-0 F12BP<3:0>: RX Buffer Written when Filter 12 Hits bits

1111 = Filter hits received in RX FIFO buffer 1110 = Filter hits received in RX Buffer 14

•

•

•

0001 = Filter hits received in RX Buffer 1

### REGISTER 21-16: CIRXFnSID: ECAN™ ACCEPTANCE FILTER n STANDARD IDENTIFIER (n = 0, 1, ...,15)

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x		
SID<10:3>									
bit 15							bit 8		

R/W-x	R/W-x	R/W-x	U-0	R/W-x	U-0	R/W-x	R/W-x
	SID<2:0>		_	EXIDE	_	EID<1	17:16>
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 15-5 SID<10:0>: Standard Identifier bits

1 = Message address bit, SIDx, must be '1' to match filter 0 = Message address bit, SIDx, must be '0' to match filter

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

bit 3 **EXIDE:** Extended Identifier Enable bit

If MIDE = 1, then:

1 = Match only messages with extended identifier addresses0 = Match only messages with standard identifier addresses

If MIDE = 0, then: Ignore EXIDE bit.

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

bit 1-0 EID<17:16>: Extended Identifier bits

1 = Message address bit, EIDx, must be '1' to match filter 0 = Message address bit, EIDx, must be '0' to match filter

### REGISTER 21-17: CIRXFnEID: ECAN™ ACCEPTANCE FILTER n EXTENDED IDENTIFIER (n = 0, 1, ..., 15)

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x		
EID<15:8>									
bit 15							bit 8		

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x		
EID<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 15-0 **EID<15:0>:** Extended Identifier bits

1 = Message address bit, EIDx, must be '1' to match filter

0 = Message address bit, EIDx, must be '0' to match filter

### REGISTER 21-18: CIFMSKSEL1: ECAN™ FILTER 7-0 MASK SELECTION 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
F7MSk	<<1:0>	F6MSI	<<1:0>	F5MS	K<1:0>	F4MS	K<1:0>
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
F3MSk	<1:0>	F2MSł	<b>&lt;&lt;1:0&gt;</b>	F1MS	K<1:0>	F0MSI	K<1:0>
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 F7MSK<1:0>: Mask Source for Filter 7 bit

11 = Reserved; do not use

10 = Acceptance Mask 2 registers contain mask

01 = Acceptance Mask 1 registers contain mask

00 = Acceptance Mask 0 registers contain mask

bit 13-12 F6MSK<1:0>: Mask Source for Filter 6 bit

11 = Reserved; do not use

10 = Acceptance Mask 2 registers contain mask

01 = Acceptance Mask 1 registers contain mask

00 = Acceptance Mask 0 registers contain mask

bit 11-10 F5MSK<1:0>: Mask Source for Filter 5 bit

11 = Reserved; do not use

10 = Acceptance Mask 2 registers contain mask

01 = Acceptance Mask 1 registers contain mask

00 = Acceptance Mask 0 registers contain mask

bit 9-8 F4MSK<1:0>: Mask Source for Filter 4 bit

11 = Reserved; do not use

10 = Acceptance Mask 2 registers contain mask

01 = Acceptance Mask 1 registers contain mask

00 = Acceptance Mask 0 registers contain mask

bit 7-6 F3MSK<1:0>: Mask Source for Filter 3 bit

11 = Reserved; do not use

10 = Acceptance Mask 2 registers contain mask

01 = Acceptance Mask 1 registers contain mask

00 = Acceptance Mask 0 registers contain mask

bit 5-4 F2MSK<1:0>: Mask Source for Filter 2 bit

11 = Reserved; do not use

10 = Acceptance Mask 2 registers contain mask

01 = Acceptance Mask 1 registers contain mask

00 = Acceptance Mask 0 registers contain mask

bit 3-2 F1MSK<1:0>: Mask Source for Filter 1 bit

11 = Reserved; do not use

10 = Acceptance Mask 2 registers contain mask

01 = Acceptance Mask 1 registers contain mask

00 = Acceptance Mask 0 registers contain mask

bit 1-0 FOMSK<1:0>: Mask Source for Filter 0 bit

11 = Reserved; do not use

10 = Acceptance Mask 2 registers contain mask

01 = Acceptance Mask 1 registers contain mask

00 = Acceptance Mask 0 registers contain mask

### REGISTER 21-19: CIFMSKSEL2: ECAN™ FILTER 15-8 MASK SELECTION 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
F15MS	K<1:0>	F14MS	K<1:0>	F13MS	SK<1:0>	F12MS	K<1:0>
bit 15	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
F11MS	K<1:0>	F10MS	K<1:0>	F9MS	K<1:0>	F8MSI	K<1:0>
bit 7							bit 0

Legend:			
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

i vala	1 511 661 6 5
bit 15-14	F15MSK<1:0>: Mask Source for Filter 15 bit
	11 = Reserved; do not use
	10 = Acceptance Mask 2 registers contain mask
	01 = Acceptance Mask 1 registers contain mask
	00 = Acceptance Mask 0 registers contain mask
bit 13-12	F14MSK<1:0>: Mask Source for Filter 14 bit
	11 = Reserved; do not use
	10 = Acceptance Mask 2 registers contain mask
	01 = Acceptance Mask 1 registers contain mask
	00 = Acceptance Mask 0 registers contain mask
bit 11-10	F13MSK<1:0>: Mask Source for Filter 13 bit
	11 = Reserved; do not use
	10 = Acceptance Mask 2 registers contain mask
	01 = Acceptance Mask 1 registers contain mask
	00 = Acceptance Mask 0 registers contain mask
bit 9-8	F12MSK<1:0>: Mask Source for Filter 12 bit
	11 = Reserved; do not use
	10 = Acceptance Mask 2 registers contain mask
	01 = Acceptance Mask 1 registers contain mask
	00 = Acceptance Mask 0 registers contain mask
bit 7-6	F11MSK<1:0>: Mask Source for Filter 11 bit
	11 = Reserved; do not use
	10 = Acceptance Mask 2 registers contain mask
	01 = Acceptance Mask 1 registers contain mask
	00 = Acceptance Mask 0 registers contain mask
bit 5-4	F10MSK<1:0>: Mask Source for Filter 10 bit
	11 = Reserved; do not use
	10 = Acceptance Mask 2 registers contain mask
	01 = Acceptance Mask 1 registers contain mask
	00 = Acceptance Mask 0 registers contain mask
bit 3-2	F9MSK<1:0>: Mask Source for Filter 9 bit
	11 = Reserved; do not use
	10 = Acceptance Mask 2 registers contain mask
	01 = Acceptance Mask 1 registers contain mask
	00 = Acceptance Mask 0 registers contain mask
bit 1-0	F8MSK<1:0>: Mask Source for Filter 8 bit
	11 = Reserved; do not use
	10 = Acceptance Mask 2 registers contain mask
	01 = Acceptance Mask 1 registers contain mask
	00 = Acceptance Mask 0 registers contain mask

### REGISTER 21-20: CIRXMnSID: ECAN™ ACCEPTANCE FILTER MASK n STANDARD IDENTIFIER

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
			SID<	10:3>			
bit 15							bit 8

R/W-x	R/W-x	R/W-x	U-0	R/W-x	U-0	R/W-x	R/W-x
	SID<2:0>		_	MIDE	_	EID<1	17:16>
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 15-5 SID<10:0>: Standard Identifier bits

1 = Include bit, SIDx, in filter comparison

0 = Bit, SIDx, is a don't care in filter comparison

bit 4 Unimplemented: Read as '0'

bit 3 MIDE: Identifier Receive Mode bit

1 = Match only message types (standard or extended address) that correspond to the EXIDE bit in filter

0 = Match either standard or extended address message if filters match

(i.e., if (Filter SID) = (Message SID) or if (Filter SID/EID) = (Message SID/EID))

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

bit 1-0 EID<17:16>: Extended Identifier bits

1 = Include bit, EIDx, in filter comparison

0 = Bit, EIDx, is a don't care in filter comparison

### REGISTER 21-21: CIRXMnEID: ECAN™ ACCEPTANCE FILTER MASK n EXTENDED IDENTIFIER

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
			EID<	15:8>			
bit 15							bit 8

| R/W-x |
|-------|-------|-------|-------|-------|-------|-------|-------|
|       |       |       | EID<  | 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 15-0 **EID<15:0>:** Extended Identifier bits

1 = Include bit, EIDx, in filter comparison

0 = Bit, EIDx, is a don't care in filter comparison

### REGISTER 21-22: CIRXFUL1: ECAN™ RECEIVE BUFFER FULL REGISTER 1

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXFUL15	RXFUL14	RXFUL13	RXFUL12	RXFUL11	RXFUL10	RXFUL9	RXFUL8
bit 15							bit 8

| R/C-0  |
|--------|--------|--------|--------|--------|--------|--------|--------|
| RXFUL7 | RXFUL6 | RXFUL5 | RXFUL4 | RXFUL3 | RXFUL2 | RXFUL1 | RXFUL0 |
| bit 7  |        |        |        |        |        |        | bit 0  |

**Legend:** 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 15-0 RXFUL15:RXFUL0: Receive Buffer n Full bits

1 = Buffer is full (set by module)

0 = Buffer is empty (clear by application software)

### REGISTER 21-23: CIRXFUL2: ECAN™ RECEIVE BUFFER FULL REGISTER 2

| R/C-0   |
|---------|---------|---------|---------|---------|---------|---------|---------|
| RXFUL31 | RXFUL30 | RXFUL29 | RXFUL28 | RXFUL27 | RXFUL26 | RXFUL25 | RXFUL24 |
| bit 15  |         |         |         |         |         |         | bit 8   |

| R/C-0   |
|---------|---------|---------|---------|---------|---------|---------|---------|
| RXFUL23 | RXFUL22 | RXFUL21 | RXFUL20 | RXFUL19 | RXFUL18 | RXFUL17 | RXFUL16 |
| bit 7   |         |         |         |         |         |         | bit 0   |

**Legend:** 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 15-0 RXFUL31:RXFUL16: Receive Buffer n Full bits

1 = Buffer is full (set by module)

0 = Buffer is empty (clear by application software)

### REGISTER 21-24: CIRXOVF1: ECAN™ RECEIVE BUFFER OVERFLOW REGISTER 1

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXOVF15	RXOVF14	RXOVF13	RXOVF12	RXOVF11	RXOVF10	RXOVF9	RXOVF8
bit 15							bit 8

| R/C-0  |
|--------|--------|--------|--------|--------|--------|--------|--------|
| RXOVF7 | RXOVF6 | RXOVF5 | RXOVF4 | RXOVF3 | RXOVF2 | RXOVF1 | RXOVF0 |
| bit 7  |        |        |        |        |        |        | bit 0  |

**Legend:** 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 15-0 **RXOVF15:RXOVF0:** Receive Buffer n Overflow bits

1 = Module pointed a write to a full buffer (set by module)

0 = Overflow is cleared (clear by application software)

### REGISTER 21-25: CIRXOVF2: ECAN™ RECEIVE BUFFER OVERFLOW REGISTER 2

| R/C-0   |
|---------|---------|---------|---------|---------|---------|---------|---------|
| RXOVF31 | RXOVF30 | RXOVF29 | RXOVF28 | RXOVF27 | RXOVF26 | RXOVF25 | RXOVF24 |
| bit 15  |         |         |         |         |         |         | bit 8   |

| R/C-0   |
|---------|---------|---------|---------|---------|---------|---------|---------|
| RXOVF23 | RXOVF22 | RXOVF21 | RXOVF20 | RXOVF19 | RXOVF18 | RXOVF17 | RXOVF16 |
| bit 7   |         |         |         |         |         |         | bit 0   |

**Legend:** 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 15-0 RXOVF31:RXOVF16: Receive Buffer n Overflow bits

1 = Module pointed a write to a full buffer (set by module)

0 = Overflow is cleared (clear by application software)

### REGISTER 21-26: CITRMnCON: ECAN™ TX/RX BUFFER mn CONTROL REGISTER (m = 0,2,4,6; n = 1,3,5,7)

R/W-0	R-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
TXENn	TXABTn	TXLARBn	TXERRn	TXREQn	RTRENn	TXnPRI<1:0>	
bit 15							bit 8

R/W-0	R-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
TXENm	TXABTm <sup>(1)</sup>	TXLARBm <sup>(1)</sup>	TXERRm <sup>(1)</sup>	TXREQm	RTRENm	TXmPR	l<1:0>
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 15-8 See Definition for Bits 7-0, Controls Buffer n

bit 7 **TXENm:** TX/RX Buffer Selection bit

1 = Buffer TRBn is a transmit buffer 0 = Buffer TRBn is a receive buffer

bit 6 **TXABTm:** Message Aborted bit<sup>(1)</sup>

1 = Message was aborted

0 = Message completed transmission successfully

bit 5 **TXLARBm:** Message Lost Arbitration bit<sup>(1)</sup>

1 = Message lost arbitration while being sent

0 = Message did not lose arbitration while being sent

bit 4 **TXERRm:** Error Detected During Transmission bit<sup>(1)</sup>

1 = A bus error occurred while the message was being sent

0 = A bus error did not occur while the message was being sent

bit 3 TXREQm: Message Send Request bit

Setting this bit to '1' requests sending a message. The bit will automatically clear when the message

is successfully sent. Clearing the bit to '0' while set will request a message abort.

bit 2 RTRENm: Auto-Remote Transmit Enable bit

1 = When a remote transmit is received, TXREQ will be set

0 = When a remote transmit is received, TXREQ will be unaffected

bit 1-0 **TXmPRI<1:0>:** Message Transmission Priority bits

11 = Highest message priority

10 = High intermediate message priority

01 = Low intermediate message priority

00 = Lowest message priority

Note 1: This bit is cleared when TXREQ is set.

Note: The buffers, SID, EID, DLC, Data Field and Receive Status registers, are located in DMA RAM.

### REGISTER 21-27: CITRBnSID: ECAN™ BUFFER n STANDARD IDENTIFIER (n = 0, 1, ..., 31)

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
_	_	_			SID<10:6>		
bit 15							bit 8

| R/W-x |
|-------|-------|-------|-------|-------|-------|-------|-------|
|       |       | SRR   | IDE   |       |       |       |       |
| 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 12-2 **SID<10:0>:** Standard Identifier bits bit 1 **SRR:** Substitute Remote Request bit

1 = Message will request remote transmission

0 = Normal message

bit 0 **IDE:** Extended Identifier bit

1 = Message will transmit extended identifier0 = Message will transmit standard identifier

### REGISTER 21-28: CITRBnEID: ECAN™ BUFFER n EXTENDED IDENTIFIER (n = 0, 1, ..., 31)

U-0	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x
_	_	_	_		EID<1	7:14>	
bit 15							bit 8

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	
EID<13:6>								
bit 7	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 15-12 **Unimplemented:** Read as '0' bit 11-0 **EID<17:6>:** Extended Identifier bits

### REGISTER 21-29: CiTRBnDLC: ECAN™ BUFFER n DATA LENGTH CONTROL (n = 0, 1, ..., 31)

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID5	EID4	EID3	EID2	EID1	EID0	RTR	RB1
bit 15							bit 8

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
_	_	_	RB0	DLC3	DLC2	DLC1	DLC0
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 **EID<5:0>:** Extended Identifier bits

bit 9 RTR: Remote Transmission Request bit

1 = Message will request remote transmission

0 = Normal message

bit 8 RB1: Reserved Bit 1

User must set this bit to '0' per CAN protocol.

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

bit 4 RB0: Reserved Bit 0

User must set this bit to '0' per CAN protocol.

bit 3-0 DLC<3:0>: Data Length Code bits

### REGISTER 21-30: CiTRBnDm: ECAN<sup>TM</sup> BUFFER n DATA FIELD BYTE m (n = 0, 1, ..., 31; m = 0, 1, ..., 7)<sup>(1)</sup>

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	
	TRBnDm<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 7-0 TRnDm<7:0>: Data Field Buffer 'n' Byte 'm' bits

**Note 1:** The Most Significant Byte contains byte (m + 1) of the buffer.

### REGISTER 21-31: CiTRBnSTAT: ECAN™ RECEIVE BUFFER n STATUS (n = 0, 1, ..., 31)

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
_	_	_			FILHIT<4:0>		
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 15-13 **Unimplemented:** Read as '0'

bit 12-8 FILHIT<4:0>: Filter Hit Code bits (only written by module for receive buffers, unused for transmit buffers)

Encodes number of filter that resulted in writing this buffer.

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

### 22.0 10-BIT/12-BIT ANALOG-TO-DIGITAL CONVERTER (ADC)

Note 1: This data sheet summarizes the features of the dsPIC33FJXXXMCX06A/X08A/X10A family of devices. However, it is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to Section 16. "Analog-to-Digital Converter (ADC)" (DS70183) in the "dsPIC33F/PIC24H Family Reference Manual", which is available from the Microchip web site (www.microchip.com).

2: Some registers and associated bits described in this section may not be available on all devices. Refer to Section 4.0 "Memory Organization" in this data sheet for device-specific register and bit information.

The dsPIC33FJXXXMCX06A/X08A/X10A devices have up to 32 ADC input channels. These devices also have up to 2 ADC modules (ADCx, where 'x' = 1 or 2), each with its own set of Special Function Registers.

The AD12B bit (ADxCON1<10>) allows each of the ADC modules to be configured by the user as either a 10-bit, 4-sample/hold ADC (default configuration) or a 12-bit, 1-sample/hold ADC.

**Note:** The ADC module needs to be disabled before modifying the AD12B bit.

### 22.1 Key Features

The 10-bit ADC configuration has the following key features:

- · Successive Approximation (SAR) conversion
- Conversion speeds of up to 1.1 Msps
- Up to 32 analog input pins
- · External voltage reference input pins
- Simultaneous sampling of up to four analog input pins
- · Automatic Channel Scan mode
- · Selectable conversion trigger source
- Selectable Buffer Fill modes
- Four result alignment options (signed/unsigned, fractional/integer)
- · Operation during CPU Sleep and Idle modes

The 12-bit ADC configuration supports all the above features, except:

- In the 12-bit configuration, conversion speeds of up to 500 ksps are supported.
- There is only 1 sample/hold amplifier in the 12-bit configuration, so simultaneous sampling of multiple channels is not supported.

Depending on the particular device pinout, the ADC can have up to 32 analog input pins, designated AN0 through AN31. In addition, there are two analog input pins for external voltage reference connections. These voltage reference inputs may be shared with other analog input pins. The actual number of analog input pins and external voltage reference input configuration will depend on the specific device.

A block diagram of the ADC is shown in Figure 22-1.

### 22.2 ADC Initialization

The following configuration steps should be performed.

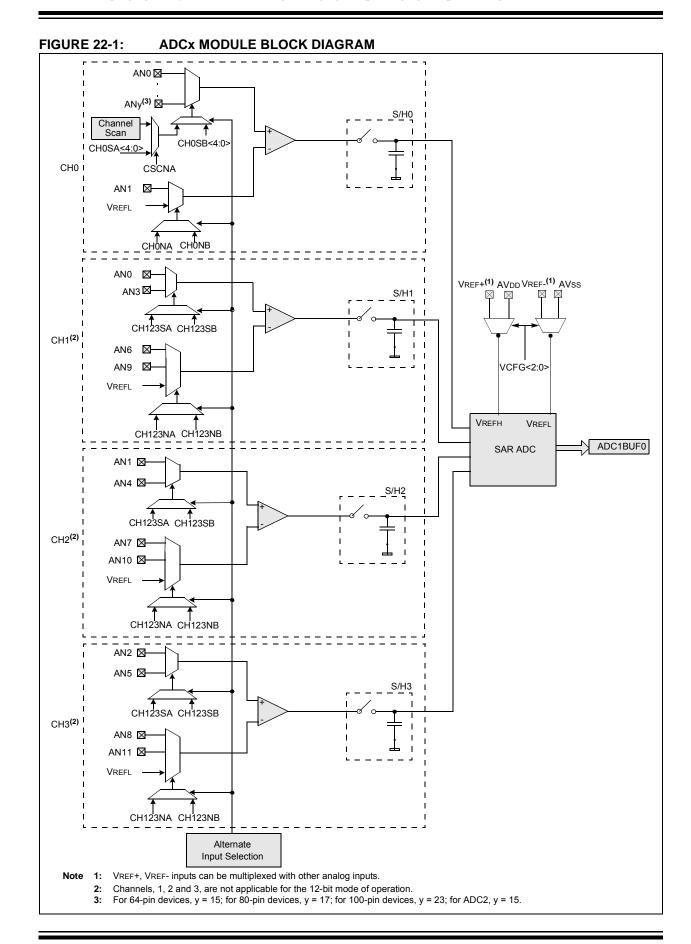
- 1. Configure the ADC module:
  - a) Select port pins as analog inputs (ADxPCFGH<15:0> or ADxPCFGL<15:0>)
  - Select voltage reference source to match expected range on analog inputs (ADxCON2<15:13>)
  - Select the analog conversion clock to match desired data rate with processor clock (ADxCON3<7:0>)
  - d) Determine how many S/H channels will be used (ADxCON2<9:8> and ADxPCFGH<15:0> or ADxPCFGL<15:0>)
  - e) Select the appropriate sample/conversion sequence (ADxCON1<7:5> and ADxCON3<12:8>)
  - f) Select how conversion results are presented in the buffer (ADxCON1<9:8>)
  - g) Turn on ADC module (ADxCON1<15>)
- 2. Configure ADC interrupt (if required):
  - a) Clear the ADxIF bit
  - b) Select ADC interrupt priority

#### 22.3 ADC and DMA

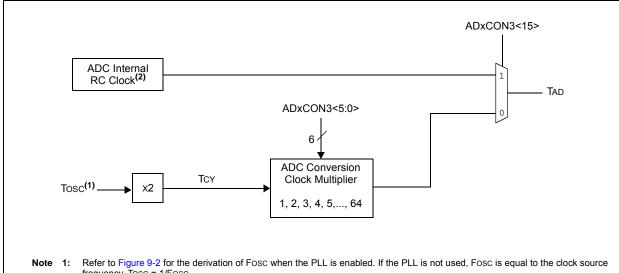
If more than one conversion result needs to be buffered before triggering an interrupt, DMA data transfers can be used. Both ADC1 and ADC2 can trigger a DMA data transfer. If ADC1 or ADC2 is selected as the DMA IRQ source, a DMA transfer occurs when the AD1IF or AD2IF bit gets set as a result of an ADC1 or ADC2 sample conversion sequence.

The SMPI<3:0> bits (ADxCON2<5:2>) are used to select how often the DMA RAM Buffer Pointer is incremented.

The ADDMABM bit (ADxCON1<12>) determines how the conversion results are filled in the DMA RAM buffer area being used for ADC. If this bit is set, DMA buffers are written in the order of conversion. The module will provide an address to the DMA channel that is the same as the address used for the non-DMA stand-alone buffer. If the ADDMABM bit is cleared, then DMA buffers are written in Scatter/Gather mode. The module will provide a scatter/gather address to the DMA channel, based on the index of the analog input and the size of the DMA buffer.



#### **FIGURE 22-2:** ADC CONVERSION CLOCK PERIOD BLOCK DIAGRAM



- frequency, Tosc = 1/Fosc.
  - 2: See the ADC electrical specifications for the exact RC clock value.

### 22.4 ADC Helpful Tips

- 1. The SMPI<3:0> (AD1CON2<5:2>) control bits:
  - Determine when the ADC interrupt flag is set and an interrupt is generated if enabled.
  - b) When the CSCNA bit (AD1CON2<10>) is set to '1', determines when the ADC analog scan channel list defined in the AD1CSSL/ AD1CSSH registers starts over from the beginning.
  - c) On devices without a DMA peripheral, determines when ADC result buffer pointer to ADC1BUF0-ADC1BUFF, gets reset back to the beginning at ADC1BUF0.
- 2. On devices without a DMA module, the ADC has 16 result buffers. ADC conversion results are stored sequentially in ADC1BUF0-ADC1BUFF regardless of which analog inputs are being used subject to the SMPI<3:0> bits (AD1CON2<5:2>) and the condition described in 1c above. There is no relationship between the ANx input being measured and which ADC buffer (ADC1BUF0-ADC1BUFF) that the conversion results will be placed in.
- On devices with a DMA module, the ADC module has only 1 ADC result buffer, (i.e., ADC1BUF0), per ADC peripheral and the ADC conversion result must be read either by the CPU or DMA controller before the next ADC conversion is complete to avoid overwriting the previous value.
- 4. The DONE bit (AD1CON1<0>) is only cleared at the start of each conversion and is set at the completion of the conversion, but remains set indefinitely even through the next sample phase until the next conversion begins. If application code is monitoring the DONE bit in any kind of software loop, the user must consider this behavior because the CPU code execution is faster than the ADC. As a result, in manual sample mode, particularly where the users code is setting the SAMP bit (AD1CON1<1>), the DONE bit should also be cleared by the user application just before setting the SAMP bit.
- 5. On devices with two ADC modules, the ADCxPCFG registers for both ADC modules must be set to a logic '1' to configure a target I/O pin as a digital I/O pin. Failure to do so means that any alternate digital input function will always see only a logic '0' as the digital input buffer is held in Disable mode.

### 22.5 ADC Resources

Many useful resources related to ADC are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page, which can be accessed using this link, contains the latest updates and additional information.

Note:

In the event you are not able to access the product page using the link above, enter this URL in your browser:

http://www.microchip.com/wwwproducts/ Devices.aspx?dDocName=en546066

### 22.5.1 KEY RESOURCES

- Section 16. "Analog-to-Digital Converter (ADC)" (DS70183)
- · Code Samples
- · Application Notes
- · Software Libraries
- Webinars
- All related dsPIC33F/PIC24H Family Reference Manuals Sections
- · Development Tools

### REGISTER 22-1: ADxCON1: ADCx CONTROL REGISTER 1 (where x = 1 or 2)

R/W-0	U-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
ADON	_	ADSIDL	ADDMABM	-	AD12B	FORM	1<1:0>
bit 15							bit 8

R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0, HC,HS	R/C-0, HC, HS
	SSRC<2:0>		_	SIMSAM	ASAM	SAMP	DONE
bit 7		_					bit 0

Legend:	HC = Hardware Clearable bit	HS = Hardware Settable 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 ADON: ADC Operating Mode bit

1 = ADC module is operating

0 = ADC is off

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

bit 13 ADSIDL: Stop in Idle Mode bit

1 = Discontinue module operation when device enters Idle mode

0 = Continue module operation in Idle mode

bit 12 ADDMABM: DMA Buffer Build Mode bit

1 = DMA buffers are written in the order of conversion. The module will provide an address to the DMA channel that is the same as the address used for the non-DMA stand-alone buffer

0 = DMA buffers are written in Scatter/Gather mode. The module will provide a scatter/gather address to the DMA channel, based on the index of the analog input and the size of the DMA buffer

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

bit 10 AD12B: 10-Bit or 12-Bit Operation Mode bit

1 = 12-bit, 1-channel ADC operation

0 = 10-bit, 4-channel ADC operation

bit 9-8 **FORM<1:0>:** Data Output Format bits

For 10-Bit Operation:

11 = Signed fractional (Dout = sddd dddd dd00 0000, where s = .NOT.d<9>)

10 = Fractional (Dout = dddd dddd dd00 0000)

01 = Signed integer (Dout = ssss sssd dddd dddd, where s = .NOT.d<9>)

00 = Integer (Dout = 0000 00dd dddd dddd)

For 12-Bit Operation:

11 = Signed fractional (Dout = sddd dddd dddd 0000, where s = .NOT.d<11>)

10 = Fractional (Dout = dddd dddd dddd 0000)

01 = Signed Integer (Dout = ssss sddd dddd, where s = .NOT.d<11>)

00 = Integer (Dout = 0000 dddd dddd dddd)

bit 7-5 SSRC<2:0>: Sample Clock Source Select bits

111 = Internal counter ends sampling and starts conversion (auto-convert)

110 = Reserved

101 = Reserved

100 = GP timer (Timer5 for ADC1, Timer3 for ADC2) compare ends sampling and starts conversion

011 = MPWM interval ends sampling and starts conversion

010 = GP timer (Timer3 for ADC1, Timer5 for ADC2) compare ends sampling and starts conversion

001 = Active transition on INTO pin ends sampling and starts conversion

000 = Clearing sample bit ends sampling and starts conversion

bit 4 Unimplemented: Read as '0'

### REGISTER 22-1: ADXCON1: ADCx CONTROL REGISTER 1 (where x = 1 or 2) (CONTINUED)

bit 3 SIMSAM: Simultaneous Sample Select bit (only applicable when CHPS<1:0> = 01 or 1x)

When AD12B = 1, SIMSAM is: U-0, Unimplemented, Read as '0'.

- 1 = Samples CH0, CH1, CH2, CH3 simultaneously (when CHPS<1:0> = 1x); or samples CH0 and CH1 simultaneously (when CHPS<1:0> = 01)
- 0 = Samples multiple channels individually in sequence
- bit 2 ASAM: ADC Sample Auto-Start bit
  - 1 = Sampling begins immediately after last conversion. SAMP bit is auto-set.
  - 0 = Sampling begins when SAMP bit is set
- bit 1 SAMP: ADC Sample Enable bit
  - 1 = ADC sample/hold amplifiers are sampling
  - 0 = ADC sample/hold amplifiers are holding

If ASAM = 0, software may write '1' to begin sampling. Automatically set by hardware if ASAM = 1.

If SSRC = 000, software may write '0' to end sampling and start conversion. If SSRC ≠ 000,

automatically cleared by hardware to end sampling and start conversion.

- bit 0 **DONE**: ADC Conversion Status bit
  - 1 = ADC conversion cycle is completed
  - 0 = ADC conversion not started or in progress

Automatically set by hardware when ADC conversion is complete. Software may write '0' to clear DONE status (software not allowed to write '1'). Clearing this bit will NOT affect any operation in progress. Automatically cleared by hardware at start of a new conversion.

### REGISTER 22-2: ADxCON2: ADCx CONTROL REGISTER 2 (where x = 1 or 2)

R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0
VCFG<2:0>		_	_	CSCNA	CHPS	S<1:0>	
bit 15							bit 8

R-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BUFS	_		SMPI<3:0>				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 15-13 **VCFG<2:0>:** Converter Voltage Reference Configuration bits

	VREF+	VREF-
000	AVDD	Avss
001	External VREF+	Avss
010	Avdd	External VREF-
011	External VREF+	External VREF-
1xx	Avdd	Avss

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

bit 10 CSCNA: Scan Input Selections for CH0+ during Sample A bit

1 = Scan inputs

0 = Do not scan inputs

bit 9-8 CHPS<1:0>: Selects Channels Utilized bits

When AD12B = 1, CHPS<1:0> is: U-0, Unimplemented, Read as '0'.

1x = Converts CH0, CH1, CH2 and CH3

01 = Converts CH0 and CH1

00 = Converts CH0

bit 7 **BUFS:** Buffer Fill Status bit (only valid when BUFM = 1)

1 = ADC is currently filling second half of buffer, user should access data in the first half
 0 = ADC is currently filling first half of buffer, user should access data in the second half

bit 6 Unimplemented: Read as '0'

bit 5-2 **SMPI<3:0>:** Selects Increment Rate for DMA Address Bits or Number of Sample/Conversion Operations per Interrupt bits

1111 = Increments the DMA address or generates interrupt after completion of every 16th sample/conversion operation

1110 = Increments the DMA address or generates interrupt after completion of every 15th sample/conversion operation

•

0001 = Increments the DMA address or generates interrupt after completion of every 2nd sample/conversion operation

0000 = Increments the DMA address or generates interrupt after completion of every sample/conversion operation

bit 1 BUFM: Buffer Fill Mode Select bit

1 = Starts filling first half of buffer on first interrupt and the second half of buffer on next interrupt

0 = Always starts filling buffer from the beginning

### REGISTER 22-2: ADxCON2: ADCx CONTROL REGISTER 2 (CONTINUED) (where x = 1 or 2)

bit 0 ALTS: Alternate Input Sample Mode Select bit

- 1 = Uses channel input selects for Sample A on first sample and Sample B on next sample
- 0 = Always uses channel input selects for Sample A

#### REGISTER 22-3: ADxCON3: ADCx CONTROL REGISTER 3

R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADRC	_	_			SAMC<4:0>(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		
ADCS<7:0> <sup>(2)</sup>									
bit 7							bit 0		

Legend:							
R = Readable bit	W = Writable bit	U = Unimplemented 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: ADC Conversion Clock Source bit
                1 = ADC internal RC clock
                0 = Clock derived from system clock
bit 14-13
               Unimplemented: Read as '0'
                SAMC<4:0>: Auto-Sample Time bits<sup>(1)</sup>
bit 12-8
                11111 = 31 TAD
                00001 = 1 TAD
                00000 = 0 TAD
                ADCS<7:0>: ADC Conversion Clock Select bits(2)
bit 7-0
                11111111 = Reserved
                01000000 = Reserved
                001111111 = Tcy \cdot (ADCS<7:0> + 1) = 64 \cdot Tcy = TAD
                00000010 = Tcy \cdot (ADCS < 7:0 > + 1) = 3 \cdot Tcy = Tad
                00000001 = Tcy \cdot (ADCS < 7:0 > + 1) = 2 \cdot Tcy = Tad
                000000000 = Tcy \cdot (ADCS < 7:0 > + 1) = 1 \cdot Tcy = Tad
```

**Note 1:** This bit is only used if ADxCON1<7:5> (SSRC<2:0>) = 111.

2: This bit is not used if ADxCON3<15>(ADRC) = 1.

#### REGISTER 22-4: ADxCON4: ADCx 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>	
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 DMABL<2:0>: Selects Number of DMA Buffer Locations per Analog Input bits

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

#### REGISTER 22-5: ADxCHS123: ADCx INPUT CHANNEL 1, 2, 3 SELECT REGISTER

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
_	_	_	_	_	CH123NB<1:0>		CH123SB
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
_	_	_	_	_	CH123NA<1:0>		CH123SA
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 10-9 CH123NB<1:0>: Channel 1, 2, 3 Negative Input Select for Sample B bits

When AD12B = 1, CHxNB is: U-0, Unimplemented, Read as '0'.

11 = CH1 negative input is AN9; CH2 negative input is AN10; CH3 negative input is AN11 10 = CH1 negative input is AN6; CH2 negative input is AN7; CH3 negative input is AN8

0x = CH1, CH2, CH3 negative input is VREF-

bit 8 CH123SB: Channel 1, 2, 3 Positive Input Select for Sample B bit

When AD12B = 1, CHxSB is: U-0, Unimplemented, Read as '0'.

1 = CH1 positive input is AN3; CH2 positive input is AN4; CH3 positive input is AN5 0 = CH1 positive input is AN0; CH2 positive input is AN1; CH3 positive input is AN2

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

bit 2-1 CH123NA<1:0>: Channel 1, 2, 3 Negative Input Select for Sample A bits

When AD12B = 1, CHxNA is: U-0, Unimplemented, Read as '0'.

11 = CH1 negative input is AN9; CH2 negative input is AN10; CH3 negative input is AN11 10 = CH1 negative input is AN6; CH2 negative input is AN7; CH3 negative input is AN8

0x = CH1, CH2, CH3 negative input is VREF-

bit 0 CH123SA: Channel 1, 2, 3 Positive Input Select for Sample A bit

When AD12B = 1, CHxSA is: U-0, Unimplemented, Read as '0'.

1 = CH1 positive input is AN3; CH2 positive input is AN4; CH3 positive input is AN5 0 = CH1 positive input is AN0; CH2 positive input is AN1; CH3 positive input is AN2

#### REGISTER 22-6: ADxCHS0: ADCx INPUT CHANNEL 0 SELECT REGISTER

R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CH0NB	_				CH0SB<4:0>		
bit 15							bit 8

R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CH0NA	_	_			CH0SA<4:0>(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 CH0NB: Channel 0 Negative Input Select for Sample B bit

Same definition as bit 7.

bit 14-13 Unimplemented: Read as '0'

bit 12-8 CH0SB<4:0>: Channel 0 Positive Input Select for Sample B bits

Same definition as bit<4:0>.

bit 7 CHONA: Channel 0 Negative Input Select for Sample A bit

1 = Channel 0 negative input is AN10 = Channel 0 negative input is VREF-

bit 6-5 **Unimplemented:** Read as '0'

bit 4-0 CH0SA<4:0>: Channel 0 Positive Input Select for Sample A bits<sup>(1)</sup>

11111 = Channel 0 positive input is AN31 11110 = Channel 0 positive input is AN30

•

00010 = Channel 0 positive input is AN2 00001 = Channel 0 positive input is AN1 00000 = Channel 0 positive input is AN0

Note 1: ADC2 can only select AN0-AN15 as positive inputs.

#### REGISTER 22-7: ADxCSSH: ADCx INPUT SCAN SELECT REGISTER HIGH(1,2)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CSS31	CSS30	CSS29	CSS28	CSS27	CSS26	CSS25	CSS24
bit 15							bit 8

| R/W-0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| CSS23 | CSS22 | CSS21 | CSS20 | CSS19 | CSS18 | CSS17 | CSS16 |
| 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 CSS<31:16>: ADC Input Scan Selection bits

1 = Select ANx for input scan0 = Skip ANx for input scan

**Note 1:** On devices without 32 analog inputs, all ADxCSSH bits may be selected by user. However, inputs selected for scan without a corresponding input on the device will convert VREFL.

2: CSSx = ANx, where x = 16 through 31.

#### REGISTER 22-8: ADXCSSL: ADCx INPUT SCAN SELECT REGISTER LOW(1,2)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CSS15	CSS14	CSS13	CSS12	CSS11	CSS10	CSS9	CSS8
bit 15							bit 8

| R/W-0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| CSS7  | CSS6  | CSS5  | CSS4  | CSS3  | CSS2  | CSS1  | CSS0  |
| 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 CSS<15:0>: ADC Input Scan Selection bits

1 = Select ANx for input scan0 = Skip ANx for input scan

**Note 1:** On devices without 16 analog inputs, all ADxCSSL bits may be selected by user. However, inputs selected for scan without a corresponding input on the device will convert VREF-.

2: CSSx = ANx, where x = 0 through 15.

#### REGISTER 22-9: ADxPCFGH: ADCx PORT CONFIGURATION REGISTER HIGH<sup>(1,2,3,4)</sup>

| R/W-0  |
|--------|--------|--------|--------|--------|--------|--------|--------|
| PCFG31 | PCFG30 | PCFG29 | PCFG28 | PCFG27 | PCFG26 | PCFG25 | PCFG24 |
| bit 15 |        |        |        |        |        |        | bit 8  |

| R/W-0  |
|--------|--------|--------|--------|--------|--------|--------|--------|
| PCFG23 | PCFG22 | PCFG21 | PCFG20 | PCFG19 | PCFG18 | PCFG17 | PCFG16 |
| 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 **PCFG<31:16>:** ADC Port Configuration Control bits

- 1 = Port pin in Digital mode; port read input enabled; ADC input multiplexer connected to AVss
- 0 = Port pin in Analog mode; port read input disabled; ADC samples pin voltage
- **Note 1:** On devices without 32 analog inputs, all PCFG bits are R/W by user. However, PCFG bits are ignored on ports without a corresponding input on the device.
  - 2: ADC2 only supports analog inputs, AN0-AN15; therefore, no ADC2 port Configuration register exists.
  - 3: PCFGx = ANx, where x = 16 through 31.
  - **4:** The PCFGx bits have no effect if the ADC module is disabled by setting the ADxMD bit in the PMDx register. In this case, all port pins multiplexed with ANx will be in Digital mode.

#### REGISTER 22-10: ADXPCFGL: ADCX PORT CONFIGURATION REGISTER LOW(1,2,3,4)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PCFG15	PCFG14	PCFG13	PCFG12	PCFG11	PCFG10	PCFG9	PCFG8
bit 15							bit 8

| R/W-0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| PCFG7 | PCFG6 | PCFG5 | PCFG4 | PCFG3 | PCFG2 | PCFG1 | PCFG0 |
| 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 **PCFG<15:0>:** ADC Port Configuration Control bits

- 1 = Port pin in Digital mode; port read input enabled; ADC input multiplexer connected to AVss
- 0 = Port pin in Analog mode; port read input disabled; ADC samples pin voltage
- **Note 1:** On devices without 16 analog inputs, all PCFG bits are R/W by user. However, PCFG bits are ignored on ports without a corresponding input on the device.
  - 2: On devices with two analog-to-digital modules, both AD1PCFGL and AD2PCFGL will affect the configuration of port pins multiplexed with AN0-AN15.
  - 3: PCFGx = ANx, where x = 0 through 15.
  - **4:** The PCFGx bits have no effect if the ADC module is disabled by setting the ADxMD bit in the PMDx register. In this case, all port pins multiplexed with ANx will be in Digital mode.

#### 23.0 SPECIAL FEATURES

# Note 1: This data sheet summarizes the features of the dsPIC33FJXXXMCX06A/X08A/X10A family of devices. However, it is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to Section 23. "CodeGuard™ Security" (DS70199), Section 24. "Programming and Diagnostics" (DS70207) and Section 25. "Device Configuration" (DS70194) in the "dsPIC33F/PIC24H Family Reference Manual", which are available from the Microchip web site (www.microchip.com).

2: Some registers and associated bits described in this section may not be available on all devices. Refer to Section 4.0 "Memory Organization" in this data sheet for device-specific register and bit information.

dsPIC33FJXXXMCX06A/X08A/X10A 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 and CodeGuard™ Security
- · JTAG Boundary Scan Interface
- In-Circuit Serial Programming™ (ICSP™)
- In-Circuit Emulation

#### 23.1 Configuration Bits

dsPIC33FJXXXMCX06A/X08A/X10A devices provide nonvolatile memory implementation for device configuration bits. Refer to **Section 25. "Device Configuration"** (DS70194) of the "dsPIC33F/PIC24H Family Reference Manual", for more information on this implementation.

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 0xF80000.

The device Configuration register map is shown in Table 23-1.

The individual Configuration bit descriptions for the Configuration registers are shown in Table 23-2.

Note that address, 0xF80000, is beyond the user program memory space. In fact, it belongs to the configuration memory space (0x800000-0xFFFFFF) which can only be accessed using table reads and table writes.

TABLE 23-1: DEVICE CONFIGURATION REGISTER MAP

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0xF80000	FBS	RBS	<1:0>	_		BSS<2:0>		BWRP	
0xF80002	FSS	RSS	<1:0>	_	_		SSS<2:0>		SWRP
0xF80004	FGS	_	_	_	_	_	GSS1	GSS0	GWRP
0xF80006	FOSCSEL	IESO	Reserved <sup>(2)</sup>	_	_	_	FNOSC<2:0>		
0xF80008	FOSC	FCKSM<1:0>		_		_	OSCIOFNC POSCMD<1:		1D<1:0>
0xF8000A	FWDT	FWDTEN	WINDIS	PLLKEN <sup>(3)</sup>	WDTPRE		WDTPOST<3:0>		
0xF8000C	FPOR	PWMPIN	HPOL	LPOL	_	_	FPW	/RT<2:0>	
0xF8000E	FICD	Rese	rved <sup>(1)</sup>	JTAGEN		_	_	ICS<	:1:0>
0xF80010	FUID0		User Unit ID Byte 0						
0xF80012	FUID1		User Unit ID Byte 1						
0xF80014	FUID2		User Unit ID Byte 2						
0xF80016	FUID3			l	Iser Unit ID	Byte 3			

**Legend:** — = unimplemented bit, reads as '0'.

Note 1: These bits are reserved for use by development tools and must be programmed as '1'.

- 2: When read, this bit returns the current programmed value.
- 3: This bit is unimplemented on dsPIC33FJ64MCX06A/X08A/X10A and dsPIC33FJ128MCX06A/X08A/X10A devices and reads as '0'.

**TABLE 23-2: CONFIGURATION BITS DESCRIPTION** 

Bit Field	Register	RTSP Effect	Description
BWRP	FBS	Immediate	Boot Segment Program Flash Write Protection bit  1 = Boot segment may be written  0 = Boot segment is write-protected
BSS<2:0>	FBS	Immediate	Boot Segment Program Flash Code Protection Size bits  x11 = No boot program Flash segment  Boot space is 1K IW less VS:  110 = Standard security; boot program Flash segment starts at end of VS, ends at 0007FEh  010 = High security; boot program Flash segment starts at end of VS, ends at 0007FEh  Boot space is 4K IW less VS:
			101 = Standard security; boot program Flash segment starts at end of VS, ends at 001FFEh  001 = High security; boot program Flash segment starts at end of VS, ends at 001FFEh  Boot space is 8K IW less VS:  100 = Standard security; boot program Flash segment starts at end of VS, ends at 003FFEh  000 = High security; boot program Flash segment starts at end of VS, ends at 003FFEh
RBS<1:0>	FBS	Immediate	Boot Segment RAM Code Protection bits  11 = No boot RAM defined  10 = Boot RAM is 128 bytes  01 = Boot RAM is 256 bytes  00 = Boot RAM is 1024 bytes
SWRP	FSS	Immediate	Secure Segment Program Flash Write Protection bit  1 = Secure segment may be written  0 = Secure segment is write-protected

TABLE 23-2: CONFIGURATION BITS DESCRIPTION (CONTINUED)

Bit Field	Register	RTSP Effect	Description
SSS<2:0>	FSS	Immediate	Secure Segment Program Flash Code Protection Size bits
			(FOR 128K and 256K DEVICES)
			x11 = No secure program Flash segment
			Secure space is 8K IW less BS:  110 = Standard security; secure program Flash segment starts at end of BS, ends at 0x003FFE  010 = High security; secure program Flash segment starts at end of BS, ends at 0x003FFE
			Secure space is 16K IW less BS:  101 = Standard security; secure program Flash segment starts at end of BS, ends at 0x007FFE  001 = High security; secure program Flash segment starts at end of BS, ends at 0x007FFE
			Secure space is 32K IW less BS:  100 = Standard security; secure program Flash segment starts at end of
			BS, ends at 0x00FFFE  000 = High security; secure program Flash segment starts at end of BS, ends at 0x00FFFE
			(FOR 64K DEVICES)  x11 = No Secure program Flash segment
			Secure space is 4K IW less BS:  110 = Standard security; secure program Flash segment starts at end of BS, ends at 0x001FFE  010 = High security; secure program Flash segment starts at end of BS, ends at 0x001FFE
			Secure space is 8K IW less BS:  101 = Standard security; secure program Flash segment starts at end of BS, ends at 0x003FFE  001 = High security; secure program Flash segment starts at end of BS, ends at 0x003FFE
			Secure space is 16K IW less BS:  100 = Standard security; secure program Flash segment starts at end of BS, ends at 007FFEh  000 = High security; secure program Flash segment starts at end of BS, ends at 0x007FFE
RSS<1:0>	FSS	Immediate	Secure Segment RAM Code Protection bits  11 = No secure RAM defined  10 = Secure RAM is 256 bytes less BS RAM  01 = Secure RAM is 2048 bytes less BS RAM  00 = Secure RAM is 4096 bytes less BS RAM
GSS<1:0>	FGS	Immediate	General Segment Code-Protect bits  11 = User program memory is not code-protected  10 = Standard security; general program Flash segment starts at end of SS, ends at EOM  0x = High security; general program Flash segment starts at end of SS, ends at EOM

TABLE 23-2: CONFIGURATION BITS DESCRIPTION (CONTINUED)

Bit Field	Register	RTSP Effect	Description
GWRP	FGS	Immediate	General Segment Write-Protect bit  1 = User program memory is not write-protected  0 = User program memory is write-protected
IESO	FOSCSEL	Immediate	Two-Speed Oscillator Start-up Enable bit  1 = Start-up device with FRC, then automatically switch to the user-selected oscillator source when ready  0 = Start-up device with user-selected oscillator source
FNOSC<2:0>	FOSCSEL	If clock switch is enabled, RTSP effect is on any device Reset; otherwise, Immediate	Initial Oscillator Source Selection bits  111 = Internal Fast RC (FRC) oscillator with postscaler  110 = Internal Fast RC (FRC) oscillator with divide-by-16  101 = LPRC oscillator  100 = Secondary (LP) oscillator  011 = Primary (XT, HS, EC) oscillator with PLL  010 = Primary (XT, HS, EC) oscillator  001 = Internal Fast RC (FRC) oscillator with PLL  000 = FRC oscillator
FCKSM<1:0>	FOSC	Immediate	Clock Switching Mode bits  1x = Clock switching is disabled, Fail-Safe Clock Monitor is disabled  01 = Clock switching is enabled, Fail-Safe Clock Monitor is disabled  00 = Clock switching is enabled, Fail-Safe Clock Monitor is enabled
OSCIOFNC	FOSC	Immediate	OSC2 Pin Function bit (except in XT and HS modes)  1 = OSC2 is clock output  0 = OSC2 is general purpose digital I/O pin
POSCMD<1:0>	FOSC	Immediate	Primary Oscillator Mode Select bits  11 = Primary oscillator disabled  10 = HS Crystal Oscillator mode  01 = XT Crystal Oscillator mode  00 = EC (External Clock) mode
FWDTEN	FWDT	Immediate	Watchdog Timer Enable bit  1 = Watchdog Timer always enabled (LPRC oscillator cannot be disabled.  Clearing the SWDTEN bit in the RCON register will have no effect.)  0 = Watchdog Timer enabled/disabled by user software (LPRC can be disabled by clearing the SWDTEN bit in the RCON register.)
WINDIS	FWDT	Immediate	Watchdog Timer Window Enable bit  1 = Watchdog Timer in Non-Window mode  0 = Watchdog Timer in Window mode
PLLKEN	FWDT	Immediate	PLL Lock Enable bit  1 = Clock switch to PLL source will wait until the PLL lock signal is valid  0 = Clock switch will not wait for the PLL lock signal
WDTPRE	FWDT	Immediate	Watchdog Timer Prescaler bit 1 = 1:128 0 = 1:32
WDT- POST<3:0>	FWDT	Immediate	Watchdog Timer Postscaler bits  1111 = 1:32,768  1110 = 1:16,384

TABLE 23-2: CONFIGURATION BITS DESCRIPTION (CONTINUED)

17(BEE 20 2:		.,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,	3 DESCRIPTION (CONTINUED)
Bit Field	Register	RTSP Effect	Description
PWMPIN	FPOR	Immediate	Motor Control PWM Module Pin Mode bit  1 = PWM module pins controlled by PORT register at device Reset (tri-stated)  0 = PWM module pins controlled by PWM module at device Reset (configured as output pins)
HPOL	FPOR	Immediate	Motor Control PWM High Side Polarity bit  1 = PWM module high side output pins have active-high output polarity  0 = PWM module high side output pins have active-low output polarity
LPOL	FPOR	Immediate	Motor Control PWM Low Side Polarity bit  1 = PWM module low side output pins have active-high output polarity  0 = PWM module low side output pins have active-low output polarity
FPWRT<2:0>	FPOR	Immediate	Power-on Reset Timer Value Select bits  111 = PWRT = 128 ms  110 = PWRT = 64 ms  101 = PWRT = 32 ms  100 = PWRT = 16 ms  011 = PWRT = 8 ms  010 = PWRT = 4 ms  001 = PWRT = 2 ms  000 = PWRT = Disabled
JTAGEN	FICD	Immediate	JTAG Enable bit  1 = JTAG enabled  0 = JTAG disabled
ICS<1:0>	FICD	Immediate	ICD Communication Channel Select bits  11 = Communicate on PGEC1 and PGED1  10 = Communicate on PGEC2 and PGED2  01 = Communicate on PGEC3 and PGED3  00 = Reserved

#### 23.2 On-Chip Voltage Regulator

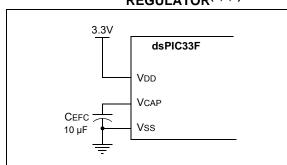
All of the dsPIC33FJXXXMCX06A/X08A/X10A devices power their core digital logic at a nominal 2.5V. This may create an issue for designs that are required to operate at a higher typical voltage, such as 3.3V. To simplify system design, all devices in the dsPIC33FJXXXMCX06A/X08A/X10A family incorporate an on-chip regulator that allows the device to run its core logic from VDD.

The regulator provides power to the core from the other VDD pins. The regulator requires that a low-ESR (less than 5 ohms) capacitor (such as tantalum or ceramic) be connected to the VCAP pin (Figure 23-1). This helps to maintain the stability of the regulator. The recommended value for the filter capacitor is provided in Table 26-13 of Section 26.1 "DC Characteristics".

Note: It is important for the low-ESR capacitor to be placed as close as possible to the VCAP pin.

On a POR, it takes approximately 20  $\mu s$  for the on-chip voltage regulator to generate an output voltage. During this time, designated as TSTARTUP, code execution is disabled. TSTARTUP is applied every time the device resumes operation after any power-down.

FIGURE 23-1: CONNECTIONS FOR THE ON-CHIP VOLTAGE REGULATOR<sup>(1,2,3)</sup>



- Note 1: These are typical operating voltages. Refer to TABLE 26-13: "Internal Voltage Regulator Specifications" located in Section 26.1 "DC Characteristics" for the full operating ranges of VDD and VCAP.
  - It is important for the low-ESR capacitor to be placed as close as possible to the VCAP pin.
  - 3: Typical VCAP pin voltage = 2.5V when VDD ≥ VDDMIN.

#### 23.3 BOR: Brown-out Reset

The BOR (Brown-out Reset) module is based on an internal voltage reference circuit that monitors the regulated supply voltage, VCAP. The main purpose of the BOR module is to generate a device Reset when a brown-out condition occurs. Brown-out conditions are generally caused by glitches on the AC mains (i.e., missing portions of the AC cycle waveform due to bad power transmission lines or voltage sags due to excessive current draw when a large inductive load is turned on).

A BOR will generate a Reset pulse which will reset the device. The BOR will select the clock source, based on the device Configuration bit values (FNOSC<2:0> and POSCMD<1:0>). Furthermore, if an oscillator mode is selected, the BOR will activate the Oscillator Start-up Timer (OST). The system clock is held until OST expires. If the PLL is used, then the clock will be held until the LOCK bit (OSCCON<5>) is '1'.

Concurrently, the PWRT time-out (TPWRT) will be applied before the internal Reset is released. If TPWRT = 0 and a crystal oscillator is being used, then a nominal delay of TFSCM = 100 is applied. The total delay in this case is TFSCM.

The BOR Status bit (RCON<1>) will be set to indicate that a BOR has occurred. The BOR circuit continues to operate while in Sleep or Idle modes and will reset the device should VDD fall below the BOR threshold voltage.

#### 23.4 Watchdog Timer (WDT)

For dsPIC33FJXXXMCX06A/X08A/X10A 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 32 kHz. This feeds a prescaler than can be configured for either 5-bit (divide-by-32) or 7-bit (divide-by-128) operation. The prescaler is set by the WDTPRE Configuration bit. With a 32 kHz input, the prescaler yields a nominal WDT time-out period (TWDT) 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 WDTPOST<3:0> Configuration bits (FWDT<3:0>) which allow the selection of a total of 16 settings, from 1:1 to 1:32,768. Using the prescaler and postscaler, time-out periods ranging 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 NOSC 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 bits (RCON<3,2>) 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.

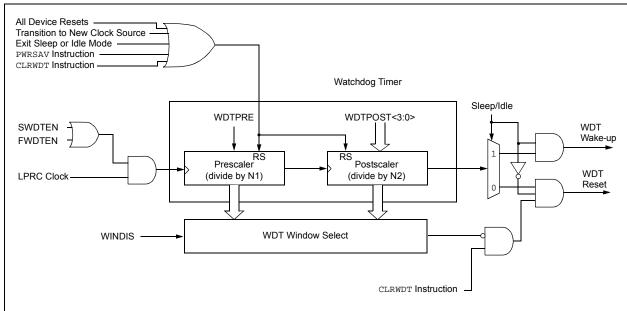
The WDT is enabled or disabled by the FWDTEN Configuration bit in the FWDT Configuration register. When the FWDTEN Configuration bit is set, the WDT is always enabled.

The WDT can be optionally controlled in software when the FWDTEN Configuration bit has been programmed to '0'. 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.

Note:

If the WINDIS bit (FWDT<6>) is cleared, the CLRWDT instruction should be executed by the application software only during the last 1/4 of the WDT period. This CLRWDT window can be determined by using a timer. If a CLRWDT instruction is executed before this window, a WDT Reset occurs.

#### FIGURE 23-2: WDT BLOCK DIAGRAM



#### 23.5 JTAG Interface

dsPIC33FJXXXMCX06A/X08A/X10A devices implement a JTAG interface, which supports boundary scan device testing, as well as in-circuit programming. Detailed information on the interface will be provided in future revisions of the document.

## 23.6 Code Protection and CodeGuard™ Security

The dsPIC33FJXXXMCX06A/X08A/X10A devices offer the advanced implementation of CodeGuard™ Security. CodeGuard Security enables multiple parties to securely share resources (memory, interrupts and peripherals) on a single chip. This feature helps protect individual Intellectual Property (IP) in collaborative system designs.

When coupled with software encryption libraries, CodeGuard™ Security can be used to securely update Flash even when multiple IPs are resident on the single chip. The code protection features vary depending on the actual device implemented. The following sections provide an overview of these features.

The code protection features are controlled by the Configuration registers: FBS, FSS and FGS.

Note: Refer to Section 23. "CodeGuard™ Security" (DS70199) in the "dsPIC33F/PIC24H Family Reference Manual" for further information on usage, configuration and operation of CodeGuard Security.

#### 23.7 In-Circuit Serial Programming

dsPIC33FJXXXMCX06A/X08A/X10A family digital signal controllers can be serially programmed while in the end application circuit. This is simply done with two lines for clock and data, and three other lines for power, ground and the programming sequence. This allows customers to manufacture boards with unprogrammed devices and then program the digital signal controller just before shipping the product. This also allows the most recent firmware, or a custom firmware, to be programmed. Please refer to the "dsPIC33F/PIC24H Flash Programming Specification" (DS70152) document for details about ICSP.

Any one out of three pairs of programming clock/data pins may be used:

- PGEC1 and PGED1
- PGEC2 and PGED2
- · PGEC3 and PGED3

#### 23.8 In-Circuit Debugger

When MPLAB® ICD 2 is selected as a debugger, the in-circuit debugging functionality is enabled. This function allows simple debugging functions when used with MPLAB IDE. Debugging functionality is controlled through the PGECx (Emulation/Debug Clock) and PGEDx (Emulation/Debug Data) pin functions.

Any one out of three pairs of debugging clock/data pins may be used:

- PGEC1 and PGED1
- PGEC2 and PGED2
- · PGEC3 and PGED3

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. 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.

#### 24.0 INSTRUCTION SET SUMMARY

Note:

This data sheet summarizes the features of the dsPIC33FJXXXMCX06A/X08A/X10A family of devices. However, it is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to the related section in the "dsPIC33F/PIC24H Family Reference Manual", which is available from the Microchip web site (www.microchip.com).

The dsPIC33F instruction set is identical to that of the dsPIC30F.

Most instructions are a single program memory word (24 bits). 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 five basic categories:

- · Word or byte-oriented operations
- · Bit-oriented operations
- · Literal operations
- · DSP operations
- · Control operations

Table 24-1 shows the general symbols used in describing the instructions.

The dsPIC33F instruction set summary in Table 24-2 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 MAC class of DSP instructions may use some of the following operands:

- The accumulator (A or B) to be used (required operand)
- The W registers to be used as the two operands
- The X and Y address space prefetch operations
- The X and Y address space prefetch destinations
- · The accumulator write back destination

The other DSP instructions do not involve any multiplication and may include:

- · The accumulator to be used (required)
- The source or destination operand (designated as Wso or Wdo, respectively) with or without an address modifier
- The amount of shift specified by a W register 'Wn' or a literal value

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 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.

Note: For more details on the instruction set, refer to the "16-bit MCU and DSC Programmer's Reference Manual" (DS70157).

TABLE 24-1: SYMBOLS USED IN OPCODE DESCRIPTIONS

Field	Description
#text	Means literal defined by "text"
(text)	Means "content of text"
[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)
Acc	One of two accumulators {A, B}
AWB	Accumulator Write-Back Destination Address register ∈ {W13, [W13]+ = 2}
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 ∈ {0x00000x1FFF}
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 $\in \{0255\}$ for Byte mode, $\{0:1023\}$ for Word mode
lit14	14-bit unsigned literal ∈ {016384}
lit16	16-bit unsigned literal ∈ {065535}
lit23	23-bit unsigned literal ∈ {08388608}; LSb must be '0'
None	Field does not require an entry, may be blank
OA, OB, SA, SB	DSP Status bits: AccA Overflow, AccB Overflow, AccA Saturate, AccB Saturate
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)

#### TABLE 24-1: SYMBOLS USED IN OPCODE DESCRIPTIONS (CONTINUED)

Field	Description
Wm*Wm	Multiplicand and Multiplier working register pair for Square instructions ∈ {W4 * W4,W5 * W5,W6 * W6,W7 * W7}
Wm*Wn	Multiplicand and Multiplier working register pair for DSP instructions ∈ {W4 * W5,W4 * W6,W4 * W7,W5 * W6,W5 * W7,W6 * W7}
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+Wb]}
Wx	X Data Space Prefetch Address register for DSP instructions ∈ {[W8]+ = 6, [W8]+ = 4, [W8]+ = 2, [W8], [W8]- = 6, [W8]- = 4, [W8]- = 2, [W9]+ = 6, [W9]+ = 4, [W9]+ = 2, [W9], [W9]- = 6, [W9]- = 4, [W9]- = 2, [W9 + W12], none}
Wxd	X Data Space Prefetch Destination register for DSP instructions ∈ {W4W7}
Wy	Y Data Space Prefetch Address register for DSP instructions ∈ {[W10]+ = 6, [W10]+ = 4, [W10]+ = 2, [W10], [W10]- = 6, [W10]- = 4, [W10]- = 2, [W11]+ = 6, [W11]+ = 4, [W11]+ = 2, [W11], [W11]- = 6, [W11]- = 4, [W11]- = 2, [W11 + W12], none}
Wyd	Y Data Space Prefetch Destination register for DSP instructions ∈ {W4W7}

**TABLE 24-2: INSTRUCTION SET OVERVIEW** 

Base Instr #	Instr Assembly Mnemonic					# of Cycles	Status Flags Affected
1	ADD	ADD	Acc	Add Accumulators	1	1	OA,OB,SA,SB
		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
		ADD	Wso,#Slit4,Acc	16-bit Signed Add to Accumulator	1	1	OA,OB,SA,SB
2	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
3	AND	AND	f	f = f .AND. WREG	1	1	N,Z
•	12.2	AND	f,WREG	WREG = f .AND. WREG	1	1	N,Z
		AND	#lit10,Wn	Wd = lit10 .AND. Wd	1	1	N,Z
				Wd = Wb .AND. Ws	1	1	N,Z
		AND	Wb, Ws, Wd	Wd = Wb .AND. lit5	1	1	N,Z
4			Wb,#lit5,Wd		+	-	
4	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
5	BCLR	BCLR	f,#bit4	Bit Clear f	1	1	None
		BCLR	Ws,#bit4	Bit Clear Ws	1	1	None
6	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		Branch if Negative	1	1 (2)	None
			N, Expr	•	1	` ′	None
		BRA	NC,Expr	Branch if Not Carry	+	1 (2)	
		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	OA,Expr	Branch if Accumulator A overflow	1	1 (2)	None
		BRA	OB,Expr	Branch if Accumulator B overflow	1	1 (2)	None
		BRA	OV,Expr	Branch if Overflow	1	1 (2)	None
		BRA	SA,Expr	Branch if Accumulator A saturated	1	1 (2)	None
		BRA	SB,Expr	Branch if Accumulator B saturated	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
7	BSET	BSET	f,#bit4	Bit Set f	1	1	None
		BSET	Ws,#bit4	Bit Set Ws	1	1	None
8	BSW	BSW.C	Ws,Wb	Write C bit to Ws <wb></wb>	1	1	None
~	2011	BSW.Z	Ws,Wb	Write 2 bit to Ws <wb></wb>	1	1	None
9	DEC						
J	BTG	BTG	f,#bit4	Bit Toggle f	1	1	None
10	DECC	BTG	Ws,#bit4	Bit Toggle Ws	1	1	None
10	BTSC	BTSC	f,#bit4	Bit Test f, Skip if Clear	1	1 (2 or 3)	None
	1	BTSC	Ws,#bit4	Bit Test Ws, Skip if Clear	1	1	None

Base Instr #	Assembly Mnemonic		Assembly Syntax	Description	# of Words	# of Cycles	Status Flags Affected
11	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
12	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
13	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
14	CALL	CALL	lit23	Call Subroutine	2	2	None
•	GILLE	CALL	Wn	Call Indirect Subroutine	1	2	None
15	CLR	CLR	f	f = 0x0000	1	1	None
13	CLR	CLR	WREG	WREG = 0x0000	1	1	None
					1	1	
		CLR	Ws	Ws = 0x0000			None
		CLR	Acc, Wx, Wxd, Wy, Wyd, AWB	Clear Accumulator	1	1	OA,OB,SA,SB
16	CLRWDT	CLRWDT		Clear Watchdog Timer	1	1	WDTO,Sleep
17	COM	COM	f	$f = \overline{f}$	1	1	N,Z
		COM	f,WREG	WREG = $\overline{f}$	1	1	N,Z
		COM	Ws,Wd	$Wd = \overline{Ws}$	1	1	N,Z
18	CP	CP	f	Compare f with WREG	1	1	C,DC,N,OV,Z
		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
19	CP0	CP0	f	Compare f with 0x0000	1	1	C,DC,N,OV,Z
19	CPU			· ·			
20		CP0	Ws	Compare Ws with 0x0000	1	1	C,DC,N,OV,Z
20	CPB	CPB	f	Compare f with WREG, with Borrow	1	1	C,DC,N,OV,Z
		CPB	Wb,#lit5	Compare Wb with lit5, with Borrow	1	1	C,DC,N,OV,Z
		CPB	Wb,Ws	Compare Wb with Ws, with Borrow (Wb – Ws – C)	1	1	C,DC,N,OV,Z
21	CPSEQ	CPSEQ	Wb, Wn	Compare Wb with Wn, Skip if =	1	1 (2 or 3)	None
22	CPSGT	CPSGT	Wb, Wn	Compare Wb with Wn, Skip if >	1	1 (2 or 3)	None
23	CPSLT	CPSLT	Wb, Wn	Compare Wb with Wn, Skip if <	1	1 (2 or 3)	None
24	CPSNE	CPSNE	Wb, Wn	Compare Wb with Wn, Skip if ≠	1	1 (2 or 3)	None
25	DAW	DAW	Wn	Wn = Decimal Adjust Wn	1	1	С
26	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
27	DEC2	DEC2	f	f = f - 2	1	1	C,DC,N,OV,Z
	DECZ				1	1	
		DEC2	f,WREG	WREG = f - 2			C,DC,N,OV,Z
20	DIGI	DEC2	Ws,Wd	Wd = Ws - 2	1	1	C,DC,N,OV,Z
28	DISI	DISI	#lit14	Disable Interrupts for k Instruction Cycles	1	1	None
29	DIV	DIV.S	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.U	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
30	DIVF	DIVF	Wm,Wn	Signed 16/16-bit Fractional Divide	1	18	N,Z,C,OV
31	DO	DO	#lit14,Expr	Do Code to PC + Expr, lit14 + 1 Times	2	2	None
		DO	Wn,Expr	Do Code to PC + Expr, (Wn) + 1 Times	2	2	None
32	ED	ED	Wm*Wm,Acc,Wx,Wy,Wxd	Euclidean Distance (no accumulate)	1	1	OA,OB,OAB, SA,SB,SAB
33	EDAC	EDAC	Wm*Wm,Acc,Wx,Wy,Wxd	Euclidean Distance	1	1	OA,OB,OAB, SA,SB,SAB

Base Instr #	Assembly Mnemonic		Assembly Syntax	Description	# of Words	# of Cycles	Status Flags Affected
34	EXCH	EXCH	Wns, Wnd	Swap Wns with Wnd	1	1	None
35	FBCL	FBCL	Ws, Wnd	Find Bit Change from Left (MSb) Side	1	1	С
36	FF1L	FF1L	Ws,Wnd	Find First One from Left (MSb) Side	1	1	С
37	FF1R	FF1R	Ws,Wnd	Find First One from Right (LSb) Side	1	1	С
38	GOTO	GOTO	Expr	Go to Address	2	2	None
		GOTO	Wn	Go to Indirect	1	2	None
39	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
40	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
41	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
42	LAC	LAC	Wso,#Slit4,Acc	Load Accumulator	1	1	OA,OB,OAB, SA,SB,SAB
43	LNK	LNK	#lit14	Link Frame Pointer	1	1	None
44	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
45	MAC	MAC Wm*Wn,Acc,Wx,Wxd,Wy,Wyd , AWB		Multiply and Accumulate	1	1	OA,OB,OAB, SA,SB,SAB
		MAC	Wm*Wm,Acc,Wx,Wxd,Wy,Wyd	Square and Accumulate	1	1	OA,OB,OAB, SA,SB,SAB
46	MOV	MOV	f,Wn	Move f to Wn	1	1	None
		MOV	f	Move f to f	1	1	None
		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	Wso,Wdo	Move Ws to Wd	1	1	None
		MOV	WREG, f	Move WREG to f	1	1	None
		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
47	MOVSAC	MOVSAC	Acc, Wx, Wxd, Wy, Wyd, AWB	Prefetch and Store Accumulator	1	1	None
48	MPY	MPY Wm*Wn,A	cc,Wx,Wxd,Wy,Wyd	Multiply Wm by Wn to Accumulator	1	1	OA,OB,OAB, SA,SB,SAB
		MPY Wm*Wm,A	cc,Wx,Wxd,Wy,Wyd	Square Wm to Accumulator	1	1	OA,OB,OAB, SA,SB,SAB
49	MPY.N	MPY.N Wm*Wn,A	cc,Wx,Wxd,Wy,Wyd	-(Multiply Wm by Wn) to Accumulator	1	1	None
50	MSC	MSC	Wm*Wm,Acc,Wx,Wxd,Wy,Wyd,AWB	Multiply and Subtract from Accumulator	1	1	OA,OB,OAB, SA,SB,SAB

Base Instr #	Assembly Mnemonic	Y ASSEMBLY SYNTAY I DESCRIPTION		Description	# of Words	# of Cycles	Status Flags Affected
51	MUL	MUL.SS	Wb, Ws, Wnd	{Wnd + 1, Wnd} = signed(Wb) * signed(Ws)	1	1	None
		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
52	NEG	NEG	Acc	Negate Accumulator	1	1	OA,OB,OAB, SA,SB,SAB
		NEG	f	$f = \bar{f} + 1$	1	1	C,DC,N,OV,Z
		NEG	f,WREG	WREG = $\overline{f}$ + 1	1	1	C,DC,N,OV,Z
		NEG	Ws,Wd	Wd = Ws + 1	1	1	C,DC,N,OV,Z
53	NOP	NOP		No Operation	1	1	None
		NOPR		No Operation	1	1	None
54	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
55	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
56	PWRSAV	PWRSAV	#lit1	Go into Sleep or Idle mode	1	1	WDTO,Sleep
57	RCALL	RCALL	Expr	Relative Call	1	2	None
		RCALL	Wn	Computed Call	1	2	None
58	REPEAT	REPEAT	#lit14	Repeat Next Instruction lit14 + 1 Times	1	1	None
		REPEAT	Wn	Repeat Next Instruction (Wn) + 1 Times	1	1	None
59	RESET	RESET		Software Device Reset	1	1	None
60	RETFIE	RETFIE		Return from Interrupt	1	3 (2)	None
61	RETLW	RETLW	#lit10,Wn	Return with Literal in Wn	1	3 (2)	None
62	RETURN	RETURN		Return from Subroutine	1	3 (2)	None
63	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
64	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
65	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
66	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
67	SAC	SAC	Acc,#Slit4,Wdo	Store Accumulator	1	1	None
		SAC.R	Acc,#Slit4,Wdo	Store Rounded Accumulator	1	1	None
68	SE	SE	Ws,Wnd	Wnd = Sign-Extended Ws	1	1	C,N,Z
69	SETM	SETM	f	f = 0xFFFF	1	1	None
		SETM	WREG	WREG = 0xFFFF	1	1	None
		SETM	Ws	Ws = 0xFFFF	1	1	None
70	SFTAC	SFTAC	Acc, Wn	Arithmetic Shift Accumulator by (Wn)	1	1	OA,OB,OAB, SA,SB,SAB
		SFTAC	Acc,#Slit6	Arithmetic Shift Accumulator by Slit6	1	1	OA,OB,OAB, SA,SB,SAB

Base Instr #	Assembly Mnemonic		Assembly Syntax	Description	# of Words	# of Cycles	Status Flags Affected
71	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	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
72	SUB	SUB	Acc	Subtract Accumulators		1	OA,OB,OAB, SA,SB,SAB
		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
73	SUBB	SUBB	f	$f = f - WREG - (\overline{C})$	1	1	C,DC,N,OV,Z
		SUBB	f,WREG	WREG = $f - WREG - (\overline{C})$	1	1	C,DC,N,OV,Z
		SUBB	#lit10,Wn	$Wn = Wn - lit10 - (\overline{C})$	1	1	C,DC,N,OV,Z
		SUBB	Wb,Ws,Wd	$Wd = Wb - Ws - (\overline{C})$	1	1	C,DC,N,OV,Z
		SUBB	Wb,#lit5,Wd	$Wd = Wb - Iit5 - (\overline{C})$	1	1	C,DC,N,OV,Z
74	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
75	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
76	SWAP	SWAP.b	Wn	Wn = Nibble Swap Wn	1	1	None
		SWAP	Wn	Wn = Byte Swap Wn	1	1	None
77	TBLRDH	TBLRDH	Ws,Wd	Read Prog<23:16> to Wd<7:0>	1	2	None
78	TBLRDL	TBLRDL	Ws,Wd	Read Prog<15:0> to Wd	1	2	None
79	TBLWTH	TBLWTH	Ws,Wd	Write Ws<7:0> to Prog<23:16>	1	2	None
80	TBLWTL	TBLWTL	Ws,Wd	Write Ws to Prog<15:0>	1	2	None
81	ULNK	ULNK		Unlink Frame Pointer	1	1	None
82	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
83	ZE	ZE	Ws, Wnd	Wnd = Zero-Extend Ws	1	1	C,Z,N

#### 25.0 DEVELOPMENT SUPPORT

The PIC<sup>®</sup> microcontrollers and dsPIC<sup>®</sup> digital signal controllers are supported with a full range of software and hardware development tools:

- · Integrated Development Environment
  - MPLAB® IDE Software
- · Compilers/Assemblers/Linkers
  - MPLAB C Compiler for Various Device Families
  - HI-TECH C® for Various Device Families
  - MPASM™ Assembler
  - MPLINK<sup>TM</sup> Object Linker/ MPLIB<sup>TM</sup> Object Librarian
  - MPLAB Assembler/Linker/Librarian for Various Device Families
- Simulators
  - MPLAB SIM Software Simulator
- Emulators
  - MPLAB REAL ICE™ In-Circuit Emulator
- · In-Circuit Debuggers
  - MPLAB ICD 3
  - PICkit™ 3 Debug Express
- · Device Programmers
  - PICkit™ 2 Programmer
  - MPLAB PM3 Device Programmer
- Low-Cost Demonstration/Development Boards, Evaluation Kits, and Starter Kits

## 25.1 MPLAB Integrated Development Environment Software

The MPLAB IDE software brings an ease of software development previously unseen in the 8/16/32-bit microcontroller market. The MPLAB IDE is a Windows® operating system-based application that contains:

- A single graphical interface to all debugging tools
  - Simulator
  - Programmer (sold separately)
  - In-Circuit Emulator (sold separately)
  - In-Circuit Debugger (sold separately)
- · A full-featured editor with color-coded context
- · A multiple project manager
- Customizable data windows with direct edit of contents
- · High-level source code debugging
- · Mouse over variable inspection
- Drag and drop variables from source to watch windows
- · Extensive on-line help
- Integration of select third party tools, such as IAR C Compilers

The MPLAB IDE allows you to:

- Edit your source files (either C or assembly)
- One-touch compile or assemble, and download to emulator and simulator tools (automatically updates all project information)
- · Debug using:
  - Source files (C or assembly)
  - Mixed C and assembly
  - Machine code

MPLAB IDE supports multiple debugging tools in a single development paradigm, from the cost-effective simulators, through low-cost in-circuit debuggers, to full-featured emulators. This eliminates the learning curve when upgrading to tools with increased flexibility and power.

## 25.2 MPLAB C Compilers for Various Device Families

The MPLAB C Compiler code development systems are complete ANSI C compilers for Microchip's PIC18, PIC24 and PIC32 families of microcontrollers and the dsPIC30 and dsPIC33 families of digital signal controllers. These compilers provide powerful integration capabilities, superior code optimization and ease of use.

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

## 25.3 HI-TECH C for Various Device Families

The HI-TECH C Compiler code development systems are complete ANSI C compilers for Microchip's PIC family of microcontrollers and the dsPIC family of digital signal controllers. These compilers provide powerful integration capabilities, omniscient code generation and ease of use.

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

The compilers include a macro assembler, linker, preprocessor, and one-step driver, and can run on multiple platforms.

#### 25.4 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 IDE projects
- User-defined macros to streamline assembly code
- Conditional assembly for multi-purpose source files
- Directives that allow complete control over the assembly process

#### 25.5 MPLINK Object Linker/ MPLIB Object Librarian

The MPLINK Object Linker combines relocatable objects created by the MPASM Assembler and the MPLAB C18 C Compiler. 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

## 25.6 MPLAB Assembler, Linker and Librarian for Various Device Families

MPLAB Assembler produces relocatable machine code from symbolic assembly language for PIC24, PIC32 and dsPIC devices. MPLAB C 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 IDE compatibility

#### 25.7 MPLAB SIM Software Simulator

The MPLAB 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 SIM Software Simulator fully supports symbolic debugging using the MPLAB C 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.

#### 25.8 MPLAB REAL ICE In-Circuit Emulator System

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 PIC<sup>®</sup> Flash MCUs and dsPIC<sup>®</sup> Flash DSCs with the easy-to-use, powerful graphical user interface of the MPLAB Integrated Development Environment (IDE), included with each kit.

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 incircuit debugger systems (RJ11) 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 IDE. In upcoming releases of MPLAB IDE, new devices will be supported, and new features will be added. MPLAB REAL ICE offers significant advantages over competitive emulators including low-cost, full-speed emulation, run-time variable watches, trace analysis, complex breakpoints, a ruggedized probe interface and long (up to three meters) interconnection cables.

## 25.9 MPLAB ICD 3 In-Circuit Debugger System

MPLAB ICD 3 In-Circuit Debugger System is Microchip's most cost effective high-speed hardware debugger/programmer for Microchip Flash Digital Signal Controller (DSC) and microcontroller (MCU) devices. It debugs and programs PIC® Flash microcontrollers and dsPIC® DSCs with the powerful, yet easy-to-use graphical user interface of MPLAB Integrated Development Environment (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.

# 25.10 PICkit 3 In-Circuit Debugger/ Programmer and PICkit 3 Debug Express

The MPLAB PICkit 3 allows debugging and programming of PIC<sup>®</sup> and dsPIC<sup>®</sup> Flash microcontrollers at a most affordable price point using the powerful graphical user interface of the MPLAB Integrated Development Environment (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 an 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™.

The PICkit 3 Debug Express include the PICkit 3, demo board and microcontroller, hookup cables and CDROM with user's guide, lessons, tutorial, compiler and MPLAB IDE software.

# 25.11 PICkit 2 Development Programmer/Debugger and PICkit 2 Debug Express

The PICkit™ 2 Development Programmer/Debugger is a low-cost development tool with an easy to use interface for programming and debugging Microchip's Flash families of microcontrollers. The full featured Windows® programming interface supports baseline (PIC10F, PIC12F5xx, PIC16F5xx), midrange (PIC12F6xx, PIC16F), PIC18F, PIC24, dsPIC30, dsPIC33, and PIC32 families of 8-bit, 16-bit, and 32-bit microcontrollers, and many Microchip Serial EEPROM products. With Microchip's powerful MPLAB Integrated Development Environment (IDE) the PICkit™ 2 enables in-circuit debugging on most PIC® microcontrollers. In-Circuit-Debugging runs, halts and single steps the program while the PIC microcontroller is embedded in the application. When halted at a breakpoint, the file registers can be examined and modified.

The PICkit 2 Debug Express include the PICkit 2, demo board and microcontroller, hookup cables and CDROM with user's guide, lessons, tutorial, compiler and MPLAB IDE software.

#### 25.12 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.

#### 25.13 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<sup>TM</sup> and dsPICDEM<sup>TM</sup> 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, Seevaluation 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.

#### 26.0 ELECTRICAL CHARACTERISTICS

This section provides an overview of dsPIC33FJXXXMCX06A/X08A/X10A electrical characteristics. Additional information will be provided in future revisions of this document as it becomes available.

Absolute maximum ratings for the dsPIC33FJXXXMCX06A/X08A/X10A 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**

#### (See Note 1)

Ambient temperature under bias	40°C to +125°C
Storage temperature	65°C to +160°C
Voltage on VDD with respect to Vss	0.3V to +4.0V
Voltage on any pin that is not 5V tolerant with respect to Vss <sup>(4)</sup>	0.3V to (VDD + 0.3V)
Voltage on any 5V tolerant pin with respect to Vss when $VDD \ge 3.0V^{(4)}$	0.3V to +5.6V
Voltage on any 5V tolerant pin with respect to Vss when VDD < 3.0V <sup>(4)</sup>	0.3V to 3.6V
Maximum current out of Vss pin	300 mA
Maximum current into VDD pin <sup>(2)</sup>	250 mA
Maximum current sourced/sunk by any 2x I/O pin(3)	8 mA
Maximum current sourced/sunk by any 4x I/O pin <sup>(3)</sup>	15 mA
Maximum current sourced/sunk by any 8x I/O pin <sup>(3)</sup>	25 mA
Maximum current sunk by all ports	200 mA
Maximum current sourced by all ports <sup>(2)</sup>	200 mA

- **Note 1:** 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.
  - 2: Maximum allowable current is a function of device maximum power dissipation (see Table 26-2).
  - **3:** Exceptions are CLKOUT, which is able to sink/source 25 mA, and the VREF+, VREF-, SCLx, SDAx, PGECx and PGEDx pins, which are able to sink/source 12 mA.
  - 4: See the "Pin Diagrams" section for 5V tolerant pins.

#### 26.1 DC Characteristics

TABLE 26-1: OPERATING MIPS vs. VOLTAGE

Param	VDD Range	Temp Range	Max MIPS
No.	(in Volts)	(in °C)	dsPIC33FJXXXMCX06A/X08A/X10A
_	VBOR-3.6V <sup>(1)</sup>	-40°C to +85°C	40
_	VBOR-3.6V <sup>(1)</sup>	-40°C to +125°C	40

**Note 1:** Device is functional at VBORMIN < VDD < VDDMIN. Analog modules such as the ADC will have degraded performance. Device functionality is tested but not characterized. Refer to parameter BO10 in Table 26-11 for the minimum and maximum BOR values.

#### **TABLE 26-2: THERMAL OPERATING CONDITIONS**

Rating	Symbol	Min	Тур	Max	Unit
dsPIC33FJXXXMCX06A/X08A/X10A					
Operating Junction Temperature Range	TJ	-40	_	+125	°C
Operating Ambient Temperature Range	TA	-40	_	+85	°C
Extended Temperature Devices					
Operating Junction Temperature Range	TJ	-40	_	+155	°C
Operating Ambient Temperature Range	TA	-40	_	+125	°C
Power Dissipation: Internal Chip Power Dissipation: $PINT = VDD \ x \ (IDD - \Sigma \ IOH)$ I/O Pin Power Dissipation:	Pb	PINT + PI/O			W
$I/O = \Sigma (\{VDD - VOH\} \times IOH) + \Sigma (VOL \times IOL)$					
Maximum Allowed Power Dissipation	PDMAX	(	ΓJ − TA)/θJ	IA	W

#### **TABLE 26-3: THERMAL PACKAGING CHARACTERISTICS**

Characteristic	Symbol	Тур	Max	Unit	Notes
Package Thermal Resistance, 100-pin TQFP (14x14x1 mm)	θЈА	40		°C/W	1
Package Thermal Resistance, 100-pin TQFP (12x12x1 mm)	θЈА	40	_	°C/W	1
Package Thermal Resistance, 80-pin TQFP (12x12x1 mm)	θЈА	40	_	°C/W	1
Package Thermal Resistance, 64-pin TQFP (10x10x1 mm)	θЈА	40	_	°C/W	1
Package Thermal Resistance, 64-pin QFN (9x9x0.9 mm)	θЈА	28	_	°C/W	1

**Note 1:** Junction to ambient thermal resistance, Theta-JA (θJA) numbers are achieved by package simulations.

TABLE 26-4: DC TEMPERATURE AND VOLTAGE SPECIFICATIONS

DC CHARACTERISTICS				d Operat otherwis	e stated ature -	<b>l)</b> 40°C ≤ 1	3.0V to 3.6V  TA ≤ +85°C for Industrial TA ≤ +125°C for Extended
Param No.	Symbol	Characteristic	Min	Typ <sup>(1)</sup>	Max	Units	Conditions
Operati	ng Voltage	e					
DC10	Supply V	oltage					
	VDD	_	3.0	_	3.6	V	_
DC12	VDR	RAM Data Retention Voltage <sup>(2)</sup>	1.8	_	_	V	_
DC16	VPOR	VDD Start Voltage to Ensure Internal Power-on Reset Signal	_	_	Vss	V	_
DC17	SVDD	VDD Rise Rate to Ensure Internal Power-on Reset Signal	0.03	_	_	V/ms	0-3.0V in 0.1s

**Note 1:** Data in "Typ" column is at 3.3V, 25°C unless otherwise stated.

<sup>2:</sup> This is the limit to which VDD can be lowered without losing RAM data.

TABLE 26-5: DC CHARACTERISTICS: OPERATING CURRENT (IDD)

DC CHARACTERISTICS			(unless oth	•	s: 3.0V to 3.6V ≤ TA ≤ +85°C for Ind ≤ TA ≤ +125°C for Exp		
Parameter No. <sup>(3)</sup>	Typical <sup>(2)</sup>	Max	Units Conditions				
Operating Cur	rent (IDD) <sup>(1)</sup>						
DC20d	27	30	mA	-40°C			
DC20a	27	30	mA	+25°C	3.3V	10 MIPS	
DC20b	27	30	mA	+85°C	3.34	10 WIFS	
DC20c	27	35	mA	+125°C			
DC21d	36	40	mA	-40°C		16 MIPS	
DC21a	37	40	mA	+25°C	3.3V		
DC21b	38	45	mA	+85°C		16 MIPS	
DC21c	39	45	mA	+125°C			
DC22d	43	50	mA	-40°C		OO MIDO	
DC22a	46	50	mA	+25°C	3.3V		
DC22b	46	55	mA	+85°C	3.34	20 MIPS	
DC22c	47	55	mA	+125°C			
DC23d	65	70	mA	-40°C			
DC23a	65	70	mA	+25°C	3.3V	30 MIPS	
DC23b	65	70	mA	+85°C	3.34	30 WIPS	
DC23c	65	70	mA	+125°C			
DC24d	84	90	mA	-40°C			
DC24a	84	90	mA	+25°C	2.2)/	40 MIDC	
DC24b	84	90	mA	+85°C	3.3V	40 MIPS	
DC24c	84	90	mA	+125°C			

- **Note 1:** IDD is primarily a function of the operating voltage and frequency. Other factors, such as I/O pin loading and switching rate, oscillator type, internal code execution pattern and temperature, also have an impact on the current consumption. The test conditions for all IDD measurements are as follows:
  - Oscillator is configured in EC mode with PLL, OSC1 is driven with external square wave from rail-to-rail (EC clock overshoot/undershoot < 250 mV required)</li>
  - CLKO is configured as an I/O input pin in the Configuration word
  - · All I/O pins are configured as inputs and pulled to Vss
  - MCLR = VDD, WDT and FSCM are disabled
  - CPU, SRAM, program memory and data memory are operational
  - No peripheral modules are operating; however, every peripheral is being clocked (defined PMDx bits are set to zero and unimplemented PMDx bits are set to one)
  - CPU executing while(1) statement
  - · JTAG is disabled
  - **2:** These parameters are characterized but not tested in manufacturing.
  - 3: Data in "Typ" column is at 3.3V, +25°C unless otherwise stated.

TABLE 26-6: DC CHARACTERISTICS: IDLE CURRENT (IIDLE)

DC CHARACTERISTICS			(unless other	perating Condition erwise stated) emperature -40°C ≤ -40°C ≤				
Parameter No. <sup>(3)</sup>	Typical <sup>(2)</sup>	Max	Units	Conditions				
Idle Current (III	DLE): Core Of	f, Clock On E	Base Current	(1)				
DC40d	3	25	mA	-40°C				
DC40a	3	25	mA	+25°C		10 MIPS		
DC40b	3	25	mA	+85°C	3.3V	10 MIFS		
DC40c	3	25	mA	+125°C				
DC41d	4	25	mA	-40°C		16 MIPS		
DC41a	5	25	mA	+25°C	3.3V			
DC41b	6	25	mA	+85°C				
DC41c	6	25	mA	+125°C				
DC42d	8	25	mA	-40°C		20 MIPS		
DC42a	9	25	mA	+25°C	3.3V			
DC42b	10	25	mA	+85°C	3.3 V	20 MIP3		
DC42c	10	25	mA	+125°C				
DC43a	15	25	mA	+25°C				
DC43d	15	25	mA	-40°C	3.3V	30 MIPS		
DC43b	15	25	mA	+85°C	3.34	30 MIFS		
DC43c	15	25	mA	+125°C				
DC44d	16	25	mA	-40°C				
DC44a	16	25	mA	+25°C	2 2 1	40 MIDS		
DC44b	16	25	mA	+85°C	3.3V	40 MIPS		
DC44c	16	25	mA	+125°C				

Note 1: Base IIDLE current is measured as follows:

- CPU core is off, oscillator is configured in EC mode and external clock active, OSC1 is driven with external square wave from rail-to-rail (EC clock overshoot/undershoot < 250 mV required)
- CLKO is configured as an I/O input pin in the Configuration word
- · All I/O pins are configured as inputs and pulled to Vss
- MCLR = VDD, WDT and FSCM are disabled
- No peripheral modules are operating; however, every peripheral is being clocked (defined PMDx bits are set to zero and unimplemented PMDx bits are set to one)
- · JTAG is disabled
- 2: These parameters are characterized but not tested in manufacturing.
- 3: Data in "Typ" column is at 3.3V, +25°C unless otherwise stated.

TABLE 26-7: DC CHARACTERISTICS: POWER-DOWN CURRENT (IPD)

DC CHARACT	ERISTICS		Standard Operating Conditions: 3.0V 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. <sup>(3)</sup>	Typical <sup>(2)</sup>	Units	Conditions					
Power-Down (	Current (IPD) <sup>(</sup>	1)						
DC60d	50	200	μА	-40°C		Base Power-Down Current <sup>(3)</sup>		
DC60a	50	200	μА	+25°C	3.3V			
DC60b	200	500	μА	+85°C	3.34	base Fower-Down Current		
DC60c	600	1000	μА	+125°C	=			
DC61d	8	13	μА	-40°C				
DC61a	10	15	μА	+25°C	2 2)/	Match dog Timon Compate Alway (3)		
DC61b	12	20	μА	+85°C	3.3V	Watchdog Timer Current: ∆IwDT <sup>(3)</sup>		
DC61c	13	25	μΑ	+125°C				

Note 1: IPD (Sleep) current is measured as follows:

- CPU core is off, oscillator is configured in EC mode and external clock active, OSC1 is driven with external square wave from rail-to-rail (EC clock overshoot/undershoot < 250 mV required)
- CLKO is configured as an I/O input pin in the Configuration word
- All I/O pins are configured as inputs and pulled to Vss
- MCLR = VDD, WDT and FSCM are disabled, all peripheral modules except the ADC are disabled (PMDx bits are all '1's). The following ADC settings are enabled for each ADC module (ADCx) prior to executing the PWRSAV instruction: ADON = 1, VCFG = 1, AD12B = 1 and ADxMD = 0.
- VREGS bit (RCON<8>) = 0 (i.e., core regulator is set to stand-by while the device is in Sleep mode)
- · RTCC is disabled.
- · JTAG is disabled
- **2:** Data in the "Typ" column is at 3.3V, +25°C unless otherwise stated.
- **3:** The Watchdog Timer Current is the additional current consumed when the WDT module is enabled. This current should be added to the base IPD current.
- **4:** These currents are measured on the device containing the most memory in this family.
- **5:** These parameters are characterized, but are not tested in manufacturing.

TABLE 26-8: DC CHARACTERISTICS: DOZE CURRENT (IDOZE)

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V 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 <sup>(2)</sup>	Max	Doze Ratio Units Conditions						
Doze Current	(IDOZE) <sup>(1)</sup>								
DC73a	11	35	1:2	mA		3.3V			
DC73f	11	30	1:64	mA	-40°C		40 MIPS		
DC73g	11	30	1:128	mA					
DC70a	42	50	1:2	mA					
DC70f	26	30	1:64	mA	+25°C	3.3V	40 MIPS		
DC70g	25	30	1:128	mA					
DC71a	41	50	1:2	mA					
DC71f	25	30	1:64	mA	+85°C	3.3V	40 MIPS		
DC71g	24	30	1:128	mA					
DC72a	42	50	1:2	mA					
DC72f	26	30	1:64	mA	+125°C	3.3V	40 MIPS		
DC72g	25	30	1:128	mA					

- **Note 1:** IDOZE is primarily a function of the operating voltage and frequency. Other factors, such as I/O pin loading and switching rate, oscillator type, internal code execution pattern and temperature, also have an impact on the current consumption. The test conditions for all IDOZE measurements are as follows:
  - Oscillator is configured in EC mode and external clock active, OSC1 is driven with external square wave from rail-to-rail with overshoot/undershoot < 250 mV</li>
  - · CLKO is configured as an I/O input pin in the Configuration word
  - All I/O pins are configured as inputs and pulled to Vss
  - MCLR = VDD, WDT and FSCM are disabled
  - CPU, SRAM, program memory and data memory are operational
  - No peripheral modules are operating; however, every peripheral is being clocked (defined PMDx bits are set to zero and unimplemented PMDx bits are set to one)
  - CPU executing while(1) statement
  - · JTAG is disabled
  - 2: Data in the "Typ" column is at 3.3V, +25°C unless otherwise stated.

TABLE 26-9: DC CHARACTERISTICS: I/O PIN INPUT SPECIFICATIONS

DC CHARACTERISTICS		Standard Operating Conditions: 3.0V 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 <sup>(1)</sup>	Max	Units	Conditions	
	VIL	Input Low Voltage						
DI10		I/O Pins	Vss	_	0.2 VDD	V		
DI15		MCLR	Vss	_	0.2 VDD	V		
DI16		I/O Pins with OSC1 or SOSCI	Vss	_	0.2 VDD	V		
DI18		I/O Pins with I <sup>2</sup> C™	Vss	_	0.3 VDD	V	SMBus disabled	
DI19		I/O Pins with I <sup>2</sup> C	Vss	_	0.8 V	V	SMBus enabled	
	VIH	Input High Voltage						
DI20		I/O Pins Not 5V Tolerant <sup>(4)</sup> I/O Pins 5V Tolerant <sup>(4)</sup>		_ _	VDD 5.5	V V		
DI28		SDAx, SCLx	0.7 VDD	_	5.5	V	SMBus disabled	
DI29		SDAx, SCLx	2.1	_	5.5	V	SMBus enabled	
	ICNPU	CNx Pull-up Current						
DI30			50	250	400	μΑ	VDD = 3.3V, VPIN = VSS	
DI50	lıL	Input Leakage Current <sup>(2,3)</sup> I/O Pins 5V Tolerant <sup>(4)</sup>	_	_	±2	μА	Vss ≤ Vpin ≤ Vdd,	
DI51		I/O Pins Not 5V Tolerant <sup>(4)</sup>	_	_	±1	μΑ	Pin at high-impedance  VSS ≤ VPIN ≤ VDD,  Pin at high-impedance,  -40°C ≤ TA ≤ +85°C	
DI51a		I/O Pins Not 5V Tolerant <sup>(4)</sup>	_	_	±2	μА	Shared with external reference pins, $-40^{\circ}C \le TA \le +85^{\circ}C$	
DI51b		I/O Pins Not 5V Tolerant <sup>(4)</sup>	_	_	±3.5	μΑ	Vss ≤ Vpin ≤ Vdd, Pin at high-impedance, -40°C ≤ Ta ≤ +125°C	
DI51c		I/O Pins Not 5V Tolerant <sup>(4)</sup>		_	±8	μΑ	Analog pins shared with external reference pins, -40°C ≤ TA ≤ +125°C	
DI55		MCLR	_	_	±2	μΑ	$Vss \leq Vpin \leq Vdd$	
DI56		OSC1	_	_	±2	μА	$\label{eq:VSS} \begin{array}{l} \text{VSS} \leq \text{VPIN} \leq \text{VDD}, \\ \text{XT and HS modes} \end{array}$	

- **Note 1:** Data in "Typ" column is at 3.3V, 25°C unless otherwise stated.
  - 2: The leakage current on the MCLR pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.
  - **3:** Negative current is defined as current sourced by the pin.
  - 4: See "Pin Diagrams" for a list of 5V tolerant pins.
  - 5: VIL source < (Vss 0.3). Characterized but not tested.
  - **6:** Non-5V tolerant pins VIH source > (VDD + 0.3), 5V tolerant pins VIH source > 5.5V. Characterized but not tested.
  - 7: Digital 5V tolerant pins cannot tolerate any "positive" input injection current from input sources > 5.5V.
  - 8: Injection currents > | 0 | can affect the ADC results by approximately 4-6 counts.
  - **9:** Any number and/or combination of I/O pins not excluded under IICL or IICH conditions are permitted provided the mathematical "absolute instantaneous" sum of the input injection currents from all pins do not exceed the specified limit. Characterized but not tested.

TABLE 26-9: DC CHARACTERISTICS: I/O PIN INPUT SPECIFICATIONS (CONTINUED)

DC CHARACTERISTICS		Standard Operating Conditions: 3.0V 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 <sup>(1)</sup>	Max Units		Conditions	
DI60a	licl	Input Low Injection Current	0	_	<sub>-5</sub> (5,8)	mA	All pins except VDD, VSS, AVDD, AVSS, MCLR, VCAP, SOSCI, SOSCO, and RB11	
DI60b	lich	Input High Injection Current	0		+5(6,7,8)	mA	All pins except VDD, VSS, AVDD, AVSS, MCLR, VCAP, SOSCI, SOSCO, RB11, and all 5V tolerant pins <sup>(7)</sup>	
DI60c	∑lict	Total Input Injection Current (sum of all I/O and control pins)	-20 <sup>(9)</sup>		+20(9)	mA	Absolute instantaneous sum of all $\pm$ input injection currents from all I/O pins (   IICL +   IICH   ) $\leq \Sigma$ IICT	

- **Note 1:** Data in "Typ" column is at 3.3V, 25°C unless otherwise stated.
  - 2: The leakage current on the MCLR pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.
  - 3: Negative current is defined as current sourced by the pin.
  - 4: See "Pin Diagrams" for a list of 5V tolerant pins.
  - **5:** VIL source < (VSS 0.3). Characterized but not tested.
  - **6:** Non-5V tolerant pins VIH source > (VDD + 0.3), 5V tolerant pins VIH source > 5.5V. Characterized but not tested.
  - 7: Digital 5V tolerant pins cannot tolerate any "positive" input injection current from input sources > 5.5V.
  - 8: Injection currents > | 0 | can affect the ADC results by approximately 4-6 counts.
  - **9:** Any number and/or combination of I/O pins not excluded under IICL or IICH conditions are permitted provided the mathematical "absolute instantaneous" sum of the input injection currents from all pins do not exceed the specified limit. Characterized but not tested.

TABLE 26-10: DC CHARACTERISTICS: I/O PIN OUTPUT SPECIFICATIONS

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V 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	
		Output Low Voltage I/O Pins: 2x Sink Driver Pins - All pins not defined by 4x or 8x driver pins	_	_	0.4	V	IOL ≤ 3 mA, VDD = 3.3V	
DO10	VoL	Output Low Voltage I/O Pins: 4x Sink Driver Pins - RA2, RA3, RA9, RA10, RA14, RA15, RB0, RB1, RB11, RF4, RF5, RG2, RG3	_	_	0.4	V	IOL ≤ 6 mA, VDD = 3.3V	
		Output Low Voltage I/O Pins: 8x Sink Driver Pins - OSC2, CLKO, RC15	_	_	0.4	V	IOL ≤ 10 mA, VDD = 3.3V	
	Vон	Output High Voltage I/O Pins: 2x Source Driver Pins - All pins not defined by 4x or 8x driver pins	2.4	_	_	V	IOL ≥ -3 mA, VDD = 3.3V	
DO20		Output High Voltage I/O Pins: 4x Source Driver Pins - RA2, RA3, RA9, RA10, RA14, RA15, RB0, RB1, RB11, RF4, RF5, RG2, RG3	2.4	_	_	V	IOL ≥ -6 mA, VDD = 3.3V	
		Output High Voltage I/O Pins: 8x Source Driver Pins - OSC2, CLKO, RC15	2.4	_	_	V	IOL ≥ -10 mA, VDD = 3.3V	
		Output High Voltage I/O Pins:	1.5	_	_	V	IOH ≥ -6 mA, VDD = 3.3V See <b>Note 1</b>	
		2x Source Driver Pins - All pins not defined by 4x or 8x driver pins	2.0	_	_		IOH ≥ -5 mA, VDD = 3.3V See <b>Note 1</b>	
			3.0	_	_		IOH ≥ -2 mA, VDD = 3.3V See <b>Note 1</b>	
		Output High Voltage 4x Source Driver Pins - RA2, RA3, RA9, RA10, RA14, RA15, RB0, RB1, RB11, RF4, RF5, RG2, RG3	1.5	_	_	V	IOH ≥ -12 mA, VDD = 3.3V See <b>Note 1</b>	
DO20A			2.0	_	_		IOH ≥ -11 mA, VDD = 3.3V See <b>Note 1</b>	
			3.0	_	_		IOH ≥ -3 mA, VDD = 3.3V See <b>Note 1</b>	
		Output High Voltage 8x Source Driver Pins - OSC2, CLKO, RC15	1.5	_	_		IOH ≥ -16 mA, VDD = 3.3V See <b>Note 1</b>	
			2.0	_	_	V	IOH ≥ -12 mA, VDD = 3.3V See <b>Note 1</b>	
			3.0	_	_		IOH ≥ -4 mA, VDD = 3.3V See <b>Note 1</b>	

Note 1: Parameters are characterized, but not tested.

#### TABLE 26-11: ELECTRICAL CHARACTERISTICS: BOR

DC CHARACTERISTICS			Standard Operation (unless otherwise Operating temperating temperations)	se stated ature -	<b>l)</b> 40°C ≤	TA ≤ +8		
Param.	Symbol	Characteris	stic <sup>(1)</sup>	Min <sup>(1)</sup>	Тур	Max <sup>(1)</sup>	Units	Conditions
BO10	VBOR	BOR Event on VDD Trans	2.40	_	2.55	V	VDD	

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

#### TABLE 26-12: DC CHARACTERISTICS: PROGRAM MEMORY

DC CHAI	RACTER	STICS	(unless		ise state	enditions: 3.0V to 3.6V ed) $-40^{\circ}C \le TA \le +85^{\circ}C$ for Industrial $-40^{\circ}C \le TA \le +125^{\circ}C$ for Extended			
Param No.	Symbol	Characteristic	Min	Min Typ <sup>(1)</sup> Max		Units	Conditions		
		Program Flash Memory							
D130	EP	Cell Endurance	10,000	_	_	E/W	_		
D131	VPR	VDD for Read	VMIN	_	3.6	V	Vмін = Minimum operating voltage		
D132b	VPEW	VDD for Self-Timed Write	VMIN	_	3.6	V	VміN = Minimum operating voltage		
D134	TRETD	Characteristic Retention	20	_	_	Year	Provided no other specifications are violated		
D135	IDDP	Supply Current during Programming	_	10	_	mA	_		
D136a	Trw	Row Write Time	1.32	_	1.74	ms	TRW = 11064 FRC cycles, TA = +85°C, see <b>Note 2</b>		
D136b	Trw	Row Write Time	1.28	_	1.79	ms	TRW = 11064 FRC cycles, TA = +150°C, see <b>Note 2</b>		
D137a	TPE	Page Erase Time	20.1	_	26.5	ms	TPE = 168517 FRC cycles, TA = +85°C, see <b>Note 2</b>		
D137b	TPE	Page Erase Time	19.5	_	27.3	ms	TPE = 168517 FRC cycles, TA = +150°C, see <b>Note 2</b>		
D138a	Tww	Word Write Cycle Time	42.3	_	55.9	μs	Tww = 355 FRC cycles, TA = +85°C, see <b>Note 2</b>		
D138b	Tww	Word Write Cycle Time	41.1	_	57.6	μs	Tww = 355 FRC cycles, TA = +150°C, see <b>Note 2</b>		
D136a D136b D137a D137b D138a D138b	TRW TRE TPE TWW TWW	Programming Row Write Time Row Write Time Page Erase Time Page Erase Time Word Write Cycle Time	1.28 20.1 19.5 42.3 41.1	- - - -	1.79 26.5 27.3 55.9 57.6	ms ms ms ms	TRW = 11064 FRC cycles, TA = +85°C, see Note 2 TRW = 11064 FRC cycles, TA = +150°C, see Note 2 TPE = 168517 FRC cycles TA = +85°C, see Note 2 TPE = 168517 FRC cycles TA = +150°C, see Note 2 TWW = 355 FRC cycles, TA = +85°C, see Note 2 TWW = 355 FRC cycles, TA = +85°C, see Note 2		

**Note 1:** Data in "Typ" column is at 3.3V, 25°C unless otherwise stated.

#### **TABLE 26-13: INTERNAL VOLTAGE REGULATOR SPECIFICATIONS**

IADLL	ABEL 20-13. INTERNAL VOLTAGE REGULATOR OF EGILIGATIONS											
	Standard Operating Conditions: 3.0V 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.	Param. Symbol Characteristics Min Typ Max Units Comments											
_	CEFC	External Filter Capacitor Value	4.7	10	_	μF	Capacitor must be low series resistance (< 5 ohms)					

<sup>2:</sup> Other conditions: FRC = 7.37 MHz, TUN<5:0> = b'0111111 (for Min), TUN<5:0> = b'100000 (for Max). This parameter depends on the FRC accuracy (see Table 26-19) and the value of the FRC Oscillator Tuning register (see Register 9-4). For complete details on calculating the Minimum and Maximum time, see Section 5.3 "Programming Operations".

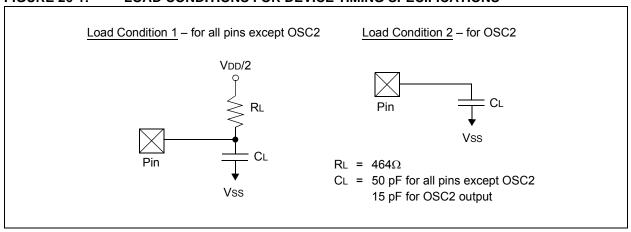
# 26.2 AC Characteristics and Timing Parameters

The information contained in this section defines dsPIC33FJXXXMCX06A/X08A/X10A AC characteristics and timing parameters.

TABLE 26-14: TEMPERATURE AND VOLTAGE SPECIFICATIONS - AC

	Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)
AC CHARACTERISTICS	Operating temperature -40°C ≤ TA ≤ +85°C for Industrial
	-40°C ≤ Ta ≤ +125°C for Extended
	Operating voltage VDD range as described in Table 26-1.

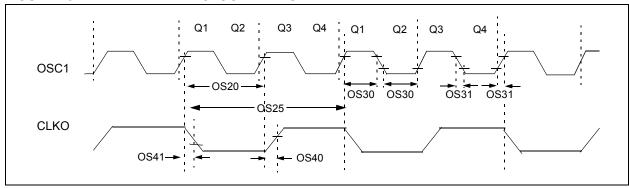
#### FIGURE 26-1: LOAD CONDITIONS FOR DEVICE TIMING SPECIFICATIONS



#### TABLE 26-15: CAPACITIVE LOADING REQUIREMENTS ON OUTPUT PINS

Param No.	Symbol	Characteristic	Min	Тур	Max	Units	Conditions
DO50	Cosc2	OSC2/SOSC2 Pin	_	_	15	pF	In XT and HS modes when external clock is used to drive OSC1
DO56	Сю	All I/O Pins and OSC2	_	_	50	pF	EC mode
DO58	Св	SCLx, SDAx	_	_	400	pF	In I <sup>2</sup> C™ mode

FIGURE 26-2: EXTERNAL CLOCK TIMING



**TABLE 26-16: EXTERNAL CLOCK TIMING REQUIREMENTS** 

AC CHA	RACTE	RISTICS	Standard Operating Conditions: 3.0V 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.	Symb	Characteristic	Min Typ <sup>(1)</sup> Max Units Condition							
OS10	Fin	External CLKI Frequency (External clocks allowed only in EC and ECPLL modes)	DC		40	MHz	EC			
		Oscillator Crystal Frequency	3.5 10 —	_ _ _	10 40 33	MHz MHz kHz	XT HS SOSC			
OS20	Tosc	Tosc = 1/Fosc	12.5	_	DC	ns	_			
OS25	TCY	Instruction Cycle Time <sup>(2)</sup>	25	_	DC	ns	_			
OS30	TosL, TosH	External Clock in (OSC1) High or Low Time	0.375 x Tosc	_	0.625 x Tosc	ns	EC			
OS31	TosR, TosF	External Clock in (OSC1) Rise or Fall Time	_	_	20	ns	EC			
OS40	TckR	CLKO Rise Time <sup>(3)</sup>	_	5.2	_	ns	_			
OS41	TckF	CLKO Fall Time <sup>(3)</sup>	_	5.2	_	ns	_			
OS42	Gм	External Oscillator Transconductance <sup>(4)</sup>	14	16	18	mA/V	VDD = 3.3V, TA = +25°C			

- Note 1: Data in "Typ" column is at 3.3V, 25°C unless otherwise stated.
  - 2: 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 OSC1/CLKI pin. When an external clock input is used, the "max." cycle time limit is "DC" (no clock) for all devices.
  - **3:** Measurements are taken in EC mode. The CLKO signal is measured on the OSC2 pin.
  - 4: Data for this parameter is preliminary. This parameter is characterized, but not tested in manufacturing.

#### TABLE 26-17: PLL CLOCK TIMING SPECIFICATIONS (VDD = 3.0V TO 3.6V)

AC CHARACTERISTICS			(unless of	Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) 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	Characteris	stic	Min	Typ <sup>(1)</sup>	Max	Units	Conditions			
OS50	FPLLI	PLL Voltage Controlled Oscillator (VCO) Input Frequency Range		0.8	_	8.0	MHz	ECPLL, HSPLL, XTPLL modes			
OS51	Fsys	On-Chip VCO System Frequency		100	_	200	MHz	_			
OS52	TLOCK	PLL Start-up Time (Lock Time)		0.9	1.5	3.1	ms	_			
OS53	DCLK	CLKO Stability (Jitter	-3.0	0.5	3.0	%	Measured over 100 ms period				

Note 1: Data in "Typ" column is at 3.3V, 25°C unless otherwise stated.

2: These parameters are characterized by similarity but are not tested in manufacturing. This specification is based on clock cycle by clock cycle measurements. To calculate the effective jitter for individual time base or communication clocks used by peripherals use the formula:

Peripheral Clock Jitter = DCLK / √(Fosc/Peripheral bit rate clock)

Example Only: Fosc = 80 MHz, DCLK = 3%, SPI bit rate clock, (i.e. SCK), is 5 MHz

SPI SCK Jitter = [ DCLK /  $\sqrt{(80 \text{ MHz}/5 \text{ MHz})}$ ] = [3%/ $\sqrt{16}$ ] = [3% / 4] = 0.75%

#### TABLE 26-18: AC CHARACTERISTICS: INTERNAL FRC ACCURACY

AC CHA	RACTERISTICS		Standard Operating Conditions: 3.0V 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	ions					
	Internal FRC Accuracy @	FRC Fr	equency	= 7.37 N	IHz <sup>(1)</sup>						
F20a	FRC	-2	_	+2	% -40°C ≤ TA ≤ +85°C VDD = 3.0-3.6V						
F20b	FRC	-5 — +5 % -40°C ≤ TA ≤ +125°C VDD = 3.0-									

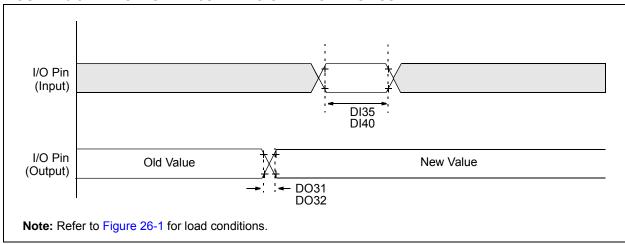
**Note 1:** Frequency calibrated at 25°C and 3.3V. TUN bits can be used to compensate for temperature drift.

#### **TABLE 26-19: INTERNAL LPRC ACCURACY**

AC CH	ARACTERISTICS	Standard Operating Conditions: 3.0V 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					
	LPRC @ 32.768 kHz <sup>(1)</sup>	)									
F21a	LPRC	-30	_	+30	%	-40°C ≤ TA ≤ +85°C —					
F21b	LPRC	-35	_	+35	%	-40°C ≤ TA ≤ +125°C —					

**Note 1:** Change of LPRC frequency as VDD changes.

#### FIGURE 26-3: CLKO AND I/O TIMING CHARACTERISTICS



#### **TABLE 26-20: I/O TIMING REQUIREMENTS**

AC CHAR	ACTERISTI	cs	Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)   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	Character	Min	Typ <sup>(1)</sup>	Max	Units	Conditions		
DO31	TioR	Port Output Rise Tim	е	_	10	25	ns	_	
DO32	TioF	Port Output Fall Time	9	_	10	25	ns		
DI35	TINP	INTx Pin High or Low	20			ns			
DI40	TRBP	CNx High or Low Tim	2	_	_	Tcy	_		

Note 1: Data in "Typ" column is at 3.3V, 25°C unless otherwise stated.

FIGURE 26-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING CHARACTERISTICS

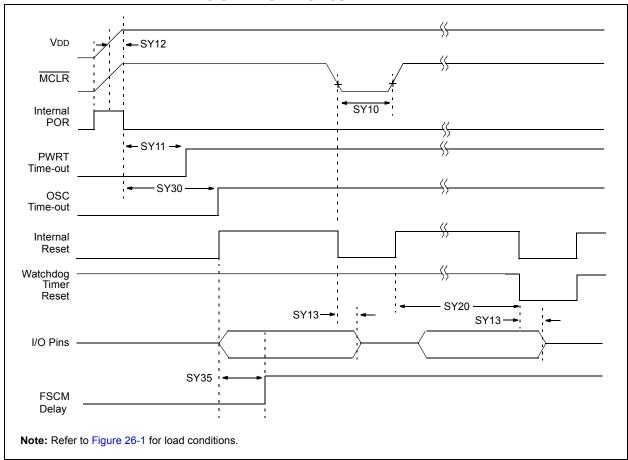


TABLE 26-21: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER, POWER-UP TIMER TIMING REQUIREMENTS

AC CHA	RACTER	ISTICS	Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) 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 <sup>(1)</sup>	Min	Typ <sup>(2)</sup>	Max	Units	Conditions			
SY10	ТмсL	MCLR Pulse Width (low)	2	_	_	μS	-40°C to +85°C			
SY11 SY12	TPWRT TPOR	Power-up Timer Period  Power-on Reset Delay		2 4 8 16 32 64 128		ms μs	-40°C to +85°C User programmable			
SY13	Tioz	I/O High-Impedance from MCLR Low or Watchdog Timer Reset	0.68	0.72	1.2	μS	_			
SY20	TWDT1	Watchdog Timer Time-out Period	_	_	_	_	See Section 23.4 "Watchdog Timer (WDT)" and LPRC specification F21 (Table 26-19)			
SY30	Tost	Oscillator Start-up Timer Period	_	1024 Tosc	_	_	Tosc = OSC1 period			
SY35	TFSCM	Fail-Safe Clock Monitor Delay	_	500	900	μS	-40°C to +85°C			

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

<sup>2:</sup> Data in "Typ" column is at 3.3V, 25°C unless otherwise stated.

FIGURE 26-5: TIMER1, 2, 3, 4, 5, 6, 7, 8 AND 9 EXTERNAL CLOCK TIMING CHARACTERISTICS

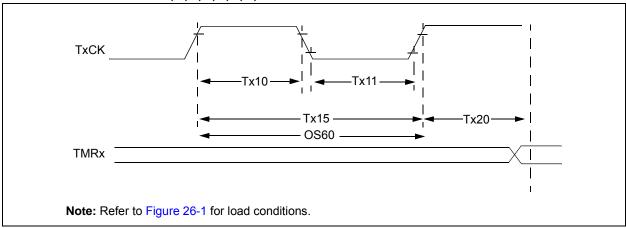


TABLE 26-22: TIMER1 EXTERNAL CLOCK TIMING REQUIREMENTS<sup>(1)</sup>

AC CHA	RACTERIST	rics		Standard Operating Conditions: 3.0V 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	Тур	Max	Units	Conditions		
TA10	ТтхН	TxCK High Time	Synchro no preso		Tcy + 20	_	_	ns	Must also meet parameter TA15		
			Synchronous, with prescaler		(Tcy + 20)/N	_	_	ns	N = prescale value (1, 8, 64,		
			Asynchronous		20	_	_	ns	256)		
TA11	TTXL	TxCK Low Time	Synchronous, no prescaler		(Tcy + 20)/N	_	_	ns	Must also meet parameter TA15		
			Synchronous, with prescaler		20	_	_	ns	N = prescale value (1, 8, 64,		
			Asynchro	onous	20	_	_	ns	256)		
TA15	ТтхР	TxCK Input Period	Synchro no preso		2Tcy + 40	_	_	ns	_		
			Synchronous, with prescaler		Greater of: 40 ns or (2Tcy + 40)/N	_	_	_	N = prescale value (1, 8, 64, 256)		
			Asynchro	onous	40	_	_	ns	_		
OS60	Ft1	SOSC1/T1CK Osci Frequency Range (oby setting bit, TCS)	oscillator enabled		DC	_	50	kHz	_		
TA20	TCKEXTMRL	Delay from Externa Edge to Timer Incre	TxCK Clock		0.75 Tcy + 40		1.75 Tcy + 40	ns	_		

Note 1: Timer1 is a Type A.

TABLE 26-23: TIMER2, TIMER4, TIMER6 AND TIMER8 EXTERNAL CLOCK TIMING REQUIREMENTS

Standard Operating Conditions: 3.0V 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		Тур	Max	Units	Conditions
TB10	TtxH	TxCK High Time	Synchronous	Greater of		_	ns	Must also meet
			mode	20 or (Tcy + 20)/N	_	_	ns	parameter TB15 N = prescale value (1, 8, 64, 256)
TB11	TtxL	TxCK Low Time	Synchronous	Greater of		_	ns	Must also meet
			mode	20 or (Tcy + 20)/N	_	_	ns	parameter TB15 N = prescale value (1, 8, 64, 256)
TB15	TtxP	TxCK Input Period	Synchronous mode	Greater of 40 or (2Tcy + 40)/N	_	_	ns	N = prescale value (1, 8, 64, 256)
TB20	TCKEXT- MRL	Delay from Externa Edge to Timer Incr		0.75 Tcy + 40	_	1.75 Tcy + 40	ns	_

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

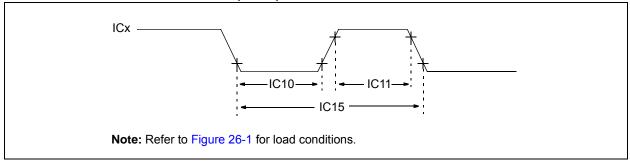
TABLE 26-24: TIMER3, TIMER5, TIMER7 AND TIMER9 EXTERNAL CLOCK TIMING REQUIREMENTS

AC CHARACTERISTICS				Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) 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	Characte	eristic		Min	Тур	Max	Units	Conditions
TC10	TtxH	TxCK High Time	Synchro	nous	Tcy + 20	_	_	ns	Must also meet parameter TC15
TC11	TtxL	TxCK Low Time	Synchro	nous	Tcy + 20	_	_	ns	Must also meet parameter TC15
TC15	2 Tcy + 40	_	_	ns	N = prescale value (1, 8, 64, 256)				
TC20	TCKEXTMRL	Delay from Externa Edge to Timer Incre	ment		0.75 Tcy + 40	_	1.75 Tcy + 40	_	_

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

**AC CHARACTERISTICS** 

#### FIGURE 26-6: INPUT CAPTURE (CAPx) TIMING CHARACTERISTICS

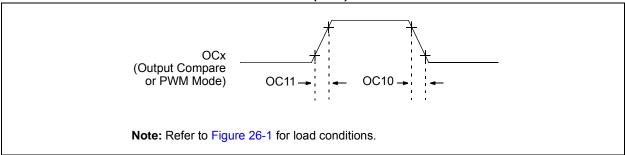


#### **TABLE 26-25: INPUT CAPTURE TIMING REQUIREMENTS**

AC CHA	RACTERI	STICS	Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq \text{TA} \leq +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \leq \text{TA} \leq +125^{\circ}\text{C}$ for Extended						
Param No.	Symbol	Characte	ristic <sup>(1)</sup>	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 Time	No prescaler	0.5 Tcy + 20	_	ns	_		
			With prescaler 10 — ns						
IC15	TccP	ICx Input Period	(Tcy + 40)/N						

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

#### FIGURE 26-7: OUTPUT COMPARE MODULE (OCx) TIMING CHARACTERISTICS



#### TABLE 26-26: OUTPUT COMPARE MODULE TIMING REQUIREMENTS

AC CHA	ARACTER	ISTICS	Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) 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 <sup>(1)</sup>	Min Typ Max Units Conditions					
OC10	TccF	OCx Output Fall Time	_	_	_	ns	See parameter D032	
OC11	TccR	OCx Output Rise Time	ns See parameter D031					

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

FIGURE 26-8: OC/PWM MODULE TIMING CHARACTERISTICS

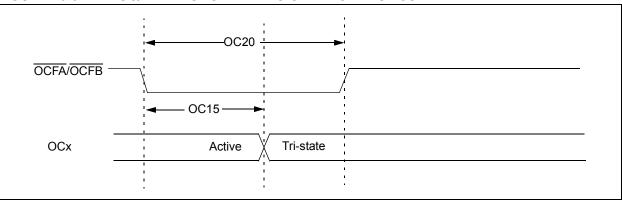


TABLE 26-27: SIMPLE OC/PWM MODE TIMING REQUIREMENTS

AC CHAP	AC CHARACTERISTICS			Standard Operating Conditions: 3.0V 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 <sup>(1)</sup>	Min	Тур	Max	Units	Conditions		
OC15	TFD	Fault Input to PWM I/O Change	Tcy + 20 ns						
OC20	TFLT	Fault Input Pulse Width	Tcy + 20 — ns —						

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

FIGURE 26-9: MOTOR CONTROL PWM MODULE FAULT TIMING CHARACTERISTICS

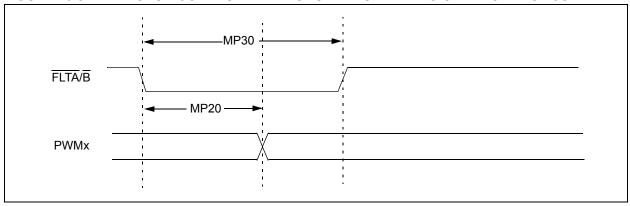


FIGURE 26-10: MOTOR CONTROL PWM MODULE TIMING CHARACTERISTICS

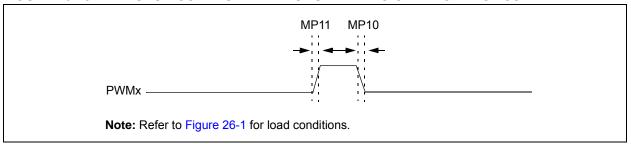


TABLE 26-28: MOTOR CONTROL PWM MODULE TIMING REQUIREMENTS

AC CHARACTERISTICS				Standard Operating Conditions: 3.0V 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 <sup>(1)</sup>	Min Typ Max Units Conditions						
MP10	TFPWM	PWM Output Fall Time	_	_	_	ns	See parameter D032		
MP11	TRPWM	PWM Output Rise Time	_	_	_	ns	See parameter D031		
MP20	TFD	Fault Input ↓ to PWM I/O Change	_		50	ns	_		
MP30	TFH	Minimum Pulse Width	50	_	_	ns	_		

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

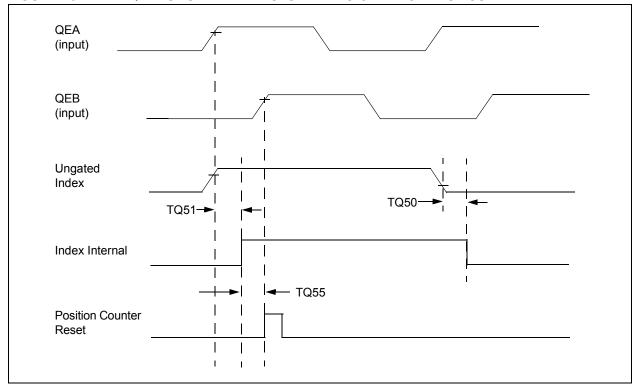
FIGURE 26-11: **QEA/QEB INPUT CHARACTERISTICS QEA** (input) TQ30 TQ35 **QEB** (input) TQ40 ► -TQ31 **TQ35 QEB** Internal

**TABLE 26-29: QUADRATURE DECODER TIMING REQUIREMENTS** 

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V 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 <sup>(1)</sup>		Typ <sup>(2)</sup>	Max	Units	Conditions	
TQ30	TquL	Quadrature Input Low Time		6 Tcy	_	ns	_	
TQ31	TquH	Quadrature Input High Time		6 Tcy	_	ns	_	
TQ35	TquIN	Quadrature Input Period		12 TcY	_	ns	_	
TQ36	TQUP	Quadrature Phase Period		3 Tcy	_	ns	_	
TQ40	TQUFL	Filter Time to Recognize Lov with Digital Filter	/	3 * N * Tcy	_	ns	N = 1, 2, 4, 16, 32, 64, 128 and 256 (Note 3)	
TQ41	TQUFH	Filter Time to Recognize Hig with Digital Filter	h	3 * N * Tcy	_	ns	N = 1, 2, 4, 16, 32, 64, 128 and 256 (Note 3)	
Note 1	Those per	ramators are characterized by	t not to	l satad in manufa	oturina	[	120 and 230 (140te	

- **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.
  - N = Index Channel Digital Filter Clock Divide Select bits. Refer to Section 15. "Quadrature Encoder Interface (QEI)" (DS70208) in the "dsPIC33F/PIC24H Family Reference Manual".

FIGURE 26-12: QEI MODULE INDEX PULSE TIMING CHARACTERISTICS



**TABLE 26-30: QEI INDEX PULSE TIMING REQUIREMENTS** 

AC CHA	RACTERIS	STICS	(unless other	Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq \text{TA} \leq +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \leq \text{TA} \leq +125^{\circ}\text{C}$ for Extended					
Param No.	Symbol	Characteristic	; <sup>(1)</sup>	Min	Max	Units	Conditions		
TQ50	TqiL	Filter Time to Recognize with Digital Filter	Low	3 * N * Tcy	_	ns	N = 1, 2, 4, 16, 32, 64, 128 and 256 <b>(Note 2)</b>		
TQ51 TqiH Filter Time to Recognize with Digital Filter			High	3 * N * Tcy		ns	N = 1, 2, 4, 16, 32, 64, 128 and 256 <b>(Note 2)</b>		
TQ55	Tqidxr	Index Pulse Recognized Counter Reset (ungated		3 Tcy		ns	_		

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

2: Alignment of index pulses to QEA and QEB is shown for position counter Reset timing only. Shown for forward direction only (QEA leads QEB). Same timing applies for reverse direction (QEA lags QEB) but index pulse recognition occurs on falling edge.

FIGURE 26-13: TIMERQ (QEI MODULE) EXTERNAL CLOCK TIMING CHARACTERISTICS

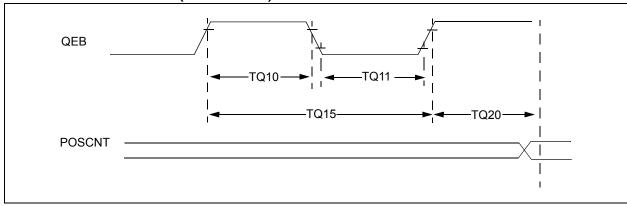


TABLE 26-31: QEI MODULE EXTERNAL CLOCK TIMING REQUIREMENTS

AC CHARACTERISTICS (U			(unles	Standard Operating Conditions: 3.0V 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	Characteri	stic <sup>(1)</sup>		Min	Тур	Max	Units	Conditions	
TQ10	TtQH		Synchro with pre	,	Tcy + 20	_	_	ns	Must also meet parameter TQ15	
TQ11	TtQL	· ·	Synchrowith pre	,	Tcy + 20		_	ns	Must also meet parameter TQ15	
TQ15	TtQP		Synchrowith pre		2 * Tcy + 40	-	_	ns	_	
TQ20	TCKEXTMRL	Delay from External Edge to Timer Increr		lock	0.5 TcY	_	1.5 Tcy	_	_	

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

TABLE 26-32: SPIX MAXIMUM DATA/CLOCK RATE SUMMARY

AC CHARAC	CTERISTICS		ag Conditions: 3.0V to 3.6V stated) ture $-40^{\circ}C \le TA \le +85^{\circ}C$ for Industrial $-40^{\circ}C \le TA \le +125^{\circ}C$ for Extended					
Maximum Data Rate	Master Transmit Only (Half-Duplex)	Master Transmit/Receive (Full-Duplex)	Slave Transmit/Receive (Full-Duplex)	CKE	СКР	SMP		
15 MHz	Table 26-33	_	_	0,1	0,1	0,1		
10 MHz	_	Table 26-34		1	0,1	1		
10 MHz	_	Table 26-35	_	0	0,1	1		
15 MHz	_	_	Table 26-36	1	0	0		
11 MHz	<u> </u>	_	Table 26-37	1	1	0		
15 MHz	_	_	Table 26-38	0	1	0		
11 MHz	_	_	Table 26-39	0	0	0		

FIGURE 26-14: SPIX MASTER MODE (HALF-DUPLEX, TRANSMIT ONLY CKE = 0) TIMING CHARACTERISTICS

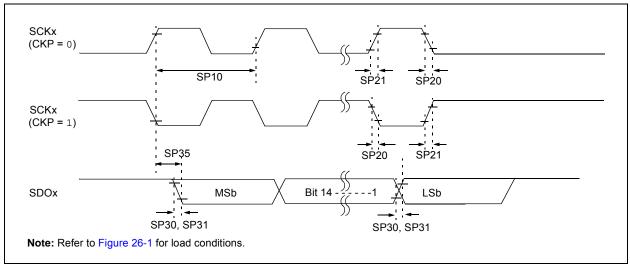


FIGURE 26-15: SPIX MASTER MODE (HALF-DUPLEX, TRANSMIT ONLY CKE = 1) TIMING CHARACTERISTICS

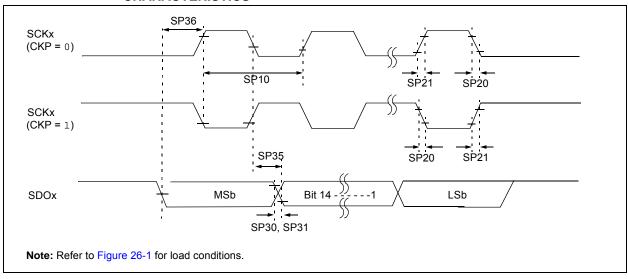


TABLE 26-33: SPIX MASTER MODE (HALF-DUPLEX, TRANSMIT ONLY) TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V 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 <sup>(1)</sup>	Min	Typ <sup>(2)</sup>	Max	Units	Conditions	
SP10	TscP	Maximum SCK Frequency	_	_	15	MHz	See Note 3	
SP20	TscF	SCKx Output Fall Time	_	_	1	ns	See parameter DO32 and <b>Note 4</b>	
SP21	TscR	SCKx Output Rise Time	_	_	_	ns	See parameter DO31 and <b>Note 4</b>	
SP30	TdoF	SDOx Data Output Fall Time	_	_	_	ns	See parameter DO32 and <b>Note 4</b>	
SP31	TdoR	SDOx Data Output Rise Time	_	_	_	ns	See parameter DO31 and Note 4	
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	_	6	20	ns	_	
SP36	TdiV2scH, TdiV2scL	SDOx Data Output Setup to First SCKx Edge	30	_	_	ns	_	

- Note 1: These parameters are characterized, but are 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 66.7 ns. Therefore, the clock generated in Master mode must not violate this specification.
  - 4: Assumes 50 pF load on all SPIx pins.

FIGURE 26-16: SPIX MASTER MODE (FULL-DUPLEX, CKE = 1, CKP = x, SMP = 1) TIMING CHARACTERISTICS

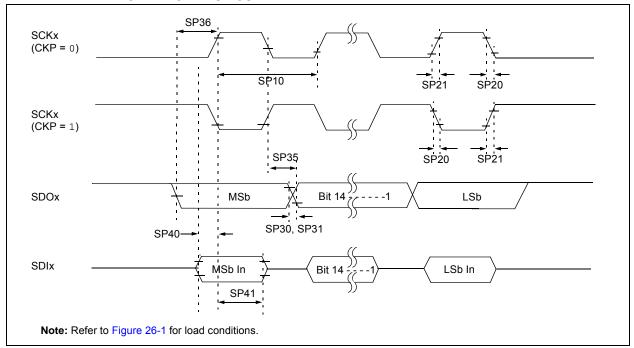


TABLE 26-34: SPIX MASTER MODE (FULL-DUPLEX, CKE = 1, CKP = x, SMP = 1) TIMING REQUIREMENTS

AC CHA	RACTERIST	Standard Operating Conditions: 3.0V 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 <sup>(1)</sup>	Min	Typ <sup>(2)</sup>	Max	Units	Conditions
SP10	TscP	Maximum SCK Frequency	_	_	10	MHz	See Note 3
SP20	TscF	SCKx Output Fall Time	_	_	_	ns	See parameter DO32 and <b>Note 4</b>
SP21	TscR	SCKx Output Rise Time	_	_	_	ns	See parameter DO31 and <b>Note 4</b>
SP30	TdoF	SDOx Data Output Fall Time	_	_	_	ns	See parameter DO32 and <b>Note 4</b>
SP31	TdoR	SDOx Data Output Rise Time	_	_	_	ns	See parameter DO31 and <b>Note 4</b>
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	30	_	_	ns	_
SP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	30	_	_	ns	_

- Note 1: These parameters are characterized, but are 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. The clock generated in Master mode must not violate this specification.
  - 4: Assumes 50 pF load on all SPIx pins.

FIGURE 26-17: SPIX MASTER MODE (FULL-DUPLEX, CKE = 0, CKP = x, SMP = 1) TIMING CHARACTERISTICS

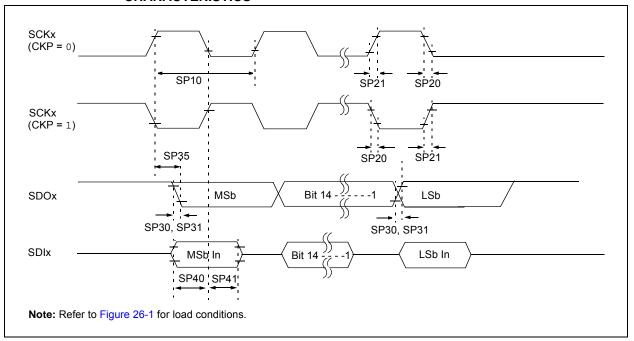


TABLE 26-35: SPIX MASTER MODE (FULL-DUPLEX, CKE = 0, CKP = x, SMP = 1) TIMING REQUIREMENTS

AC CHA	AC CHARACTERISTICS			Standard Operating Conditions: 2.4V 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 <sup>(1)</sup>	Min	Typ <sup>(2)</sup>	Max	Units	Conditions	
SP10	TscP	Maximum SCK Frequency	_	_	10	MHz	-40°C to +125°C and see <b>Note 3</b>	
SP20	TscF	SCKx Output Fall Time	_	_	_	ns	See parameter DO32 and <b>Note 4</b>	
SP21	TscR	SCKx Output Rise Time	_	_	_	ns	See parameter DO31 and <b>Note 4</b>	
SP30	TdoF	SDOx Data Output Fall Time	_	_	_	ns	See parameter DO32 and <b>Note 4</b>	
SP31	TdoR	SDOx Data Output Rise Time	_	_	_	ns	See parameter DO31 and <b>Note 4</b>	
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	_	6	20	ns	_	
SP36	TdoV2scH, TdoV2scL	SDOx Data Output Setup to First SCKx Edge	30	_	_	ns	_	
SP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	30	_	_	ns	_	
SP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	30	_	_	ns	_	

- Note 1: These parameters are characterized, but are 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. The clock generated in Master mode must not violate this specification.
  - 4: Assumes 50 pF load on all SPIx pins.

FIGURE 26-18: SPIX SLAVE MODE (FULL-DUPLEX, CKE = 1, CKP = 0, SMP = 0) TIMING CHARACTERISTICS

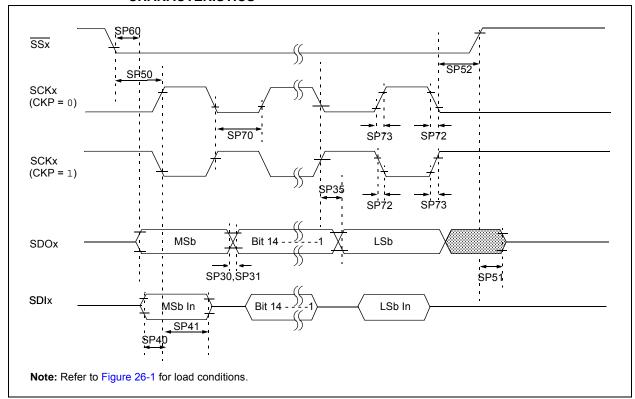


TABLE 26-36: SPIX SLAVE MODE (FULL-DUPLEX, CKE = 1, CKP = 0, SMP = 0) TIMING REQUIREMENTS

AC CHA	AC CHARACTERISTICS			Standard Operating Conditions: 2.4V 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 <sup>(1)</sup>	Min	Typ <sup>(2)</sup>	Max	Units	Conditions	
SP70	TscP	Maximum SCK Input Frequency	_	_	15	MHz	See Note 3	
SP72	TscF	SCKx Input Fall Time	_		_	ns	See parameter DO32 and Note 4	
SP73	TscR	SCKx Input Rise Time	_	_	_	ns	See parameter DO31 and Note 4	
SP30	TdoF	SDOx Data Output Fall Time	_	_	_	ns	See parameter DO32 and Note 4	
SP31	TdoR	SDOx Data Output Rise Time	_		_	ns	See parameter DO31 and Note 4	
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	_	6	20	ns	_	
SP36	TdoV2scH, TdoV2scL	SDOx Data Output Setup to First SCKx Edge	30	_	_	ns	_	
SP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	30	_	_	ns	_	
SP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	30		_	ns	_	
SP50	TssL2scH, TssL2scL	SSx ↓ to SCKx ↑ or SCKx Input	120		_	ns	_	
SP51	TssH2doZ	SSx ↑to SDOx Output High-Impedance <sup>(4)</sup>	10		50	ns	_	
SP52	TscH2ssH TscL2ssH	SSx after SCKx Edge	1.5 Tcy + 40	_	_	ns	See Note 4	
SP60	TssL2doV	SDOx Data Output Valid after SSx Edge	_	_	50	ns	_	

- **Note 1:** These parameters are characterized, but are 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 66.7 ns. Therefore, the SCK clock generated by the Master must not violate this specificiation.
  - 4: Assumes 50 pF load on all SPIx pins.

FIGURE 26-19: SPIX SLAVE MODE (FULL-DUPLEX, CKE = 1, CKP = 1, SMP = 0) TIMING CHARACTERISTICS

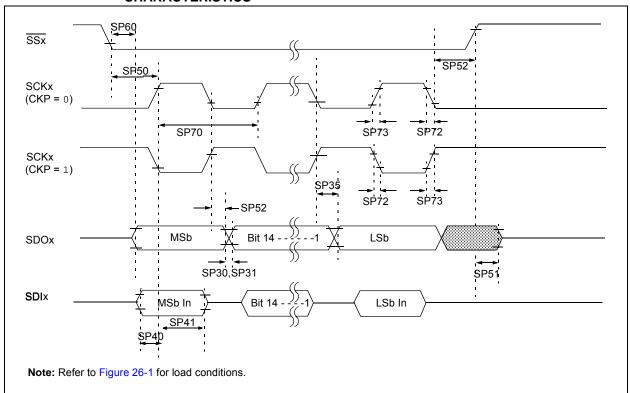


TABLE 26-37: SPIX SLAVE MODE (FULL-DUPLEX, CKE = 1, CKP = 1, SMP = 0) TIMING REQUIREMENTS

AC CHA	AC CHARACTERISTICS			Standard Operating Conditions: 2.4V 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 <sup>(1)</sup>	Min	Typ <sup>(2)</sup>	Max	Units	Conditions	
SP70	TscP	Maximum SCK Input Frequency	_	_	11	MHz	See Note 3	
SP72	TscF	SCKx Input Fall Time	_	_	_	ns	See parameter DO32 and Note 4	
SP73	TscR	SCKx Input Rise Time	_	_	_	ns	See parameter DO31 and Note 4	
SP30	TdoF	SDOx Data Output Fall Time	_	_	_	ns	See parameter DO32 and Note 4	
SP31	TdoR	SDOx Data Output Rise Time	_		_	ns	See parameter DO31 and Note 4	
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	_	6	20	ns	_	
SP36	TdoV2scH, TdoV2scL	SDOx Data Output Setup to First SCKx Edge	30	_	_	ns	_	
SP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	30		_	ns	_	
SP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	30		_	ns	_	
SP50	TssL2scH, TssL2scL	$\overline{SSx} \downarrow \text{ to SCKx} \uparrow \text{ or SCKx Input}$	120			ns	_	
SP51	TssH2doZ	SSx ↑to SDOx Output High-Impedance <sup>(4)</sup>	10		50	ns	_	
SP52	TscH2ssH TscL2ssH	SSx after SCKx Edge	1.5 Tcy + 40	_	_	ns	See Note 4	
SP60	TssL2doV	SDOx Data Output Valid after SSx Edge	_		50	ns		

- **Note 1:** These parameters are characterized, but are 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 91 ns. Therefore, the SCK clock generated by the Master must not violate this specificiation.
  - 4: Assumes 50 pF load on all SPIx pins.

FIGURE 26-20: SPIX SLAVE MODE (FULL-DUPLEX CKE = 0, CKP = 1, SMP = 0) TIMING CHARACTERISTICS

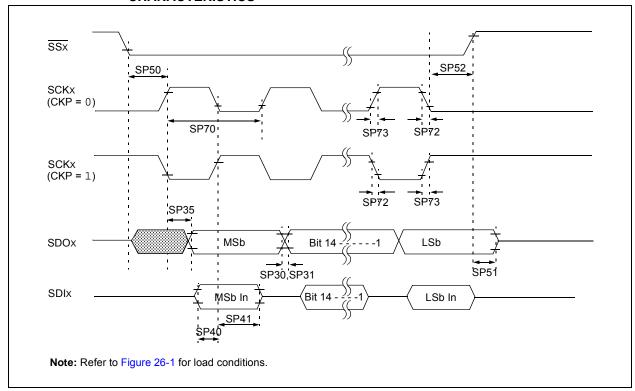


TABLE 26-38: SPIX SLAVE MODE (FULL-DUPLEX, CKE = 0, CKP = 1, SMP = 0) TIMING REQUIREMENTS

AC CHA	AC CHARACTERISTICS			Standard Operating Conditions: 2.4V 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 <sup>(1)</sup>	Min	Typ <sup>(2)</sup>	Max	Units	Conditions	
SP70	TscP	Maximum SCK Input Frequency	_	_	15	MHz	See Note 3	
SP72	TscF	SCKx Input Fall Time	_	1	1	ns	See parameter DO32 and Note 4	
SP73	TscR	SCKx Input Rise Time	_		1	ns	See parameter DO31 and Note 4	
SP30	TdoF	SDOx Data Output Fall Time	_			ns	See parameter DO32 and Note 4	
SP31	TdoR	SDOx Data Output Rise Time	_	1	1	ns	See parameter DO31 and Note 4	
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	_	6	20	ns	_	
SP36	TdoV2scH, TdoV2scL	SDOx Data Output Setup to First SCKx Edge	30	_	_	ns	_	
SP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	30		_	ns	_	
SP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	30	1	1	ns	_	
SP50	TssL2scH, TssL2scL	$\overline{SSx} \downarrow \text{ to SCKx} \uparrow \text{ or SCKx Input}$	120	1	1	ns	_	
SP51	TssH2doZ	SSx ↑to SDOx Output High-Impedance <sup>(4)</sup>	10	_	50	ns	_	
SP52	TscH2ssH TscL2ssH	SSx after SCKx Edge	1.5 Tcy + 40	_	_	ns	See Note 4	

Note 1: These parameters are characterized, but are 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 66.7 ns. Therefore, the SCK clock generated by the Master must not violate this specificiation.
- 4: Assumes 50 pF load on all SPIx pins.

FIGURE 26-21: SPIX SLAVE MODE (FULL-DUPLEX, CKE = 0, CKP = 0, SMP = 0) TIMING CHARACTERISTICS

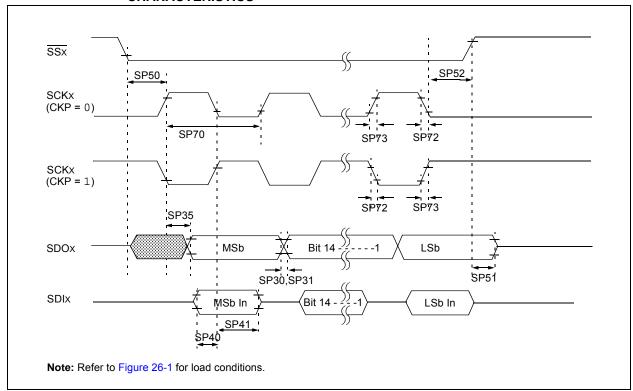


TABLE 26-39: SPIX SLAVE MODE (FULL-DUPLEX, CKE = 0, CKP = 0, SMP = 0) TIMING REQUIREMENTS

AC CHA	AC CHARACTERISTICS			Standard Operating Conditions: 2.4V 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 <sup>(1)</sup>	Min	Typ <sup>(2)</sup>	Max	Units	Conditions	
SP70	TscP	Maximum SCK Input Frequency	_	_	11	MHz	See Note 3	
SP72	TscF	SCKx Input Fall Time	_	_	_	ns	See parameter DO32 and Note 4	
SP73	TscR	SCKx Input Rise Time	_	1	1	ns	See parameter DO31 and Note 4	
SP30	TdoF	SDOx Data Output Fall Time	_		I	ns	See parameter DO32 and Note 4	
SP31	TdoR	SDOx Data Output Rise Time	_		I	ns	See parameter DO31 and Note 4	
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	_	6	20	ns	_	
SP36	TdoV2scH, TdoV2scL	SDOx Data Output Setup to First SCKx Edge	30	_	_	ns	_	
SP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	30	_	_	ns	_	
SP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	30	_	_	ns	_	
SP50	TssL2scH, TssL2scL	$\overline{SSx} \downarrow \text{ to SCKx} \uparrow \text{ or SCKx Input}$	120		_	ns	_	
SP51	TssH2doZ	SSx ↑to SDOx Output High-Impedance <sup>(4)</sup>	10	_	50	ns	_	
SP52	TscH2ssH TscL2ssH	SSx after SCKx Edge	1.5 Tcy + 40	_	_	ns	See Note 4	

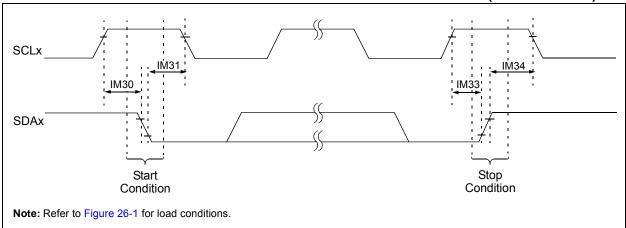
Note 1: These parameters are characterized, but are not tested in manufacturing.

<sup>2:</sup> Data in "Typ" column is at 3.3V, 25°C unless otherwise stated.

**<sup>3:</sup>** The minimum clock period for SCKx is 91 ns. Therefore, the SCK clock generated by the Master must not violate this specificiation.

<sup>4:</sup> Assumes 50 pF load on all SPIx pins.

#### FIGURE 26-22: I2Cx BUS START/STOP BITS TIMING CHARACTERISTICS (MASTER MODE)



#### FIGURE 26-23: I2Cx BUS DATA TIMING CHARACTERISTICS (MASTER MODE)

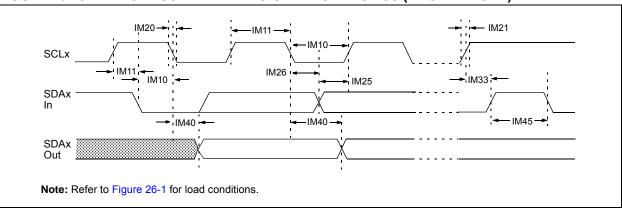


TABLE 26-40: I2Cx BUS DATA TIMING REQUIREMENTS (MASTER MODE)

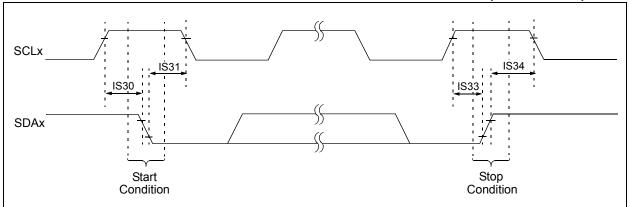
				Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)					
AC CHA	RACTER	ISTICS		Operating tempera		)°C < Ta <	≤ +85°C for Industrial		
				operating tempera			+125°C for Extended		
Param No.	Symbol	Charact	teristic	Min <sup>(1)</sup>	Max	Units	Conditions		
IM10	TLO:SCL	Clock Low Time	100 kHz mode	Tcy/2 (BRG + 1)	_	μS	_		
			400 kHz mode	Tcy/2 (BRG + 1)	_	μS	_		
			1 MHz mode <sup>(2)</sup>	Tcy/2 (BRG + 1)	_	μS	_		
IM11	THI:SCL	Clock High Time	100 kHz mode	Tcy/2 (BRG + 1)	_	μS	_		
			400 kHz mode	Tcy/2 (BRG + 1)	_	μS	_		
			1 MHz mode <sup>(2)</sup>	Tcy/2 (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 <sup>(2)</sup>	_	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 <sup>(2)</sup>	_	300	ns			
IM25	TSU:DAT	Data Input	100 kHz mode	250	_	ns	_		
		Setup Time	400 kHz mode	100	_	ns			
			1 MHz mode <sup>(2)</sup>	40	_	ns			
IM26 THD:DA	THD:DAT	Data Input	100 kHz mode	0	_	μS	_		
		Hold Time	400 kHz mode	0	0.9	μS			
			1 MHz mode <sup>(2)</sup>	0.2	_	μS			
IM30	Tsu:sta	Start Condition	100 kHz mode	Tcy/2 (BRG + 1)	_	μS	Only relevant for		
		Setup Time	400 kHz mode	Tcy/2 (BRG + 1)	_	μS	Repeated Start		
			1 MHz mode <sup>(2)</sup>	Tcy/2 (BRG + 1)	_	μS	condition		
IM31	THD:STA	Start Condition Hold Time	100 kHz mode	Tcy/2 (BRG + 1)	_	μS	After this period the		
			400 kHz mode	Tcy/2 (BRG + 1)	_	μS	first clock pulse is		
			1 MHz mode <sup>(2)</sup>	Tcy/2 (BRG + 1)	_	μS	generated		
IM33	Tsu:sto	Stop Condition	100 kHz mode	Tcy/2 (BRG + 1)	_	μS	_		
		Setup Time	400 kHz mode	Tcy/2 (BRG + 1)	_	μS			
			1 MHz mode <sup>(2)</sup>	Tcy/2 (BRG + 1)	_	μS			
IM34	THD:STO	Stop Condition	100 kHz mode	Tcy/2 (BRG + 1)	_	ns	_		
		Hold Time	400 kHz mode	Tcy/2 (BRG + 1)	_	ns			
			1 MHz mode <sup>(2)</sup>	Tcy/2 (BRG + 1)	_	ns			
IM40	TAA:SCL	Output Valid	100 kHz mode	_	3500	μS	_		
		From Clock	400 kHz mode	_	1000	μS	_		
			1 MHz mode <sup>(2)</sup>	_	400	μS	_		
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 <sup>(2)</sup>	0.5	_	μS	transmission can start		
IM50	Св	Bus Capacitive Lo	oading	_	400	pF	_		
IM51	TPGD	Pulse Gobbler De	elay	65	390	ns	See Note 3		

Note 1: BRG is the value of the I<sup>2</sup>C™ Baud Rate Generator. Refer to Section 19. "Inter-Integrated Circuit (I<sup>2</sup>C™)" (DS70195) in the "dsPIC33F/PIC24H Family Reference Manual".

<sup>2:</sup> Maximum pin capacitance = 10 pF for all I2Cx pins (for 1 MHz mode only).

**<sup>3:</sup>** Typical value for this parameter is 130 ns.

FIGURE 26-24: I2Cx BUS START/STOP BITS TIMING CHARACTERISTICS (SLAVE MODE)



#### FIGURE 26-25: I2Cx BUS DATA TIMING CHARACTERISTICS (SLAVE MODE)

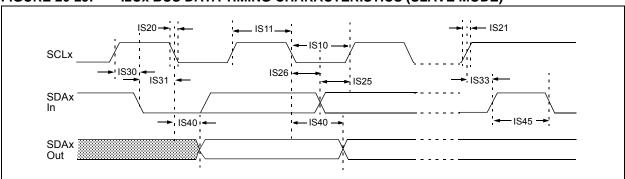


TABLE 26-41: I2Cx BUS DATA TIMING REQUIREMENTS (SLAVE MODE)

AC CHA	RACTERIS	STICS		Standard Op (unless othe Operating ten	rwise st	<b>ated)</b> e -40°	ons: 3.0V to 3.6V $C \le TA \le +85^{\circ}C$ for Industrial $C \le TA \le +125^{\circ}C$ for Extended
Param No.	Symbol	Charac	teristic	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 <sup>(1)</sup>	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 <sup>(1)</sup>	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 <sup>(1)</sup>	_	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 <sup>(1)</sup>	_	300	ns	
IS25	Tsu:dat	Data Input	100 kHz mode	250	_	ns	_
		Setup Time	400 kHz mode	100	_	ns	
			1 MHz mode <sup>(1)</sup>	100	_	ns	
IS26	THD:DAT	Data Input	100 kHz mode	0	_	μS	_
		Hold Time	400 kHz mode	0	0.9	μS	
			1 MHz mode <sup>(1)</sup>	0	0.3	μS	
IS30	Tsu:sta	Start Condition	100 kHz mode	4.7	_	μS	Only relevant for Repeated
		Setup Time	400 kHz mode	0.6	_	μS	Start condition
			1 MHz mode <sup>(1)</sup>	0.25	_	μS	
IS31	THD:STA	Start Condition	100 kHz mode	4.0	_	μS	After this period, the first
		Hold Time	400 kHz mode	0.6	_	μS	clock pulse is generated
			1 MHz mode <sup>(1)</sup>	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 <sup>(1)</sup>	0.6	_	μS	
IS34	THD:STO	Stop Condition	100 kHz mode	4000	_	ns	_
		Hold Time	400 kHz mode	600	_	ns	
			1 MHz mode <sup>(1)</sup>	250		ns	
IS40	TAA:SCL	Output Valid	100 kHz mode	0	3500	ns	_
		From Clock	400 kHz mode	0	1000	ns	1
			1 MHz mode <sup>(1)</sup>	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
			1 MHz mode <sup>(1)</sup>	0.5	_	μS	can start
IS50	Св	Bus Capacitive Lo			400	pF	

**Note 1:** Maximum pin capacitance = 10 pF for all I2Cx pins (for 1 MHz mode only).

#### FIGURE 26-26: CAN MODULE I/O TIMING CHARACTERISTICS

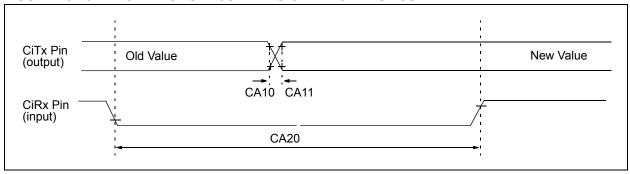


TABLE 26-42: ECAN™ TECHNOLOGY MODULE I/O TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V 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 <sup>(1)</sup>	Min	Тур	Max	Units	Conditions
CA10	TioF	Port Output Fall Time	_	_	_	ns	See parameter D032
CA11	TioR	Port Output Rise Time	_	_	_	ns	See parameter D031
CA20	Tcwf	Pulse Width to Trigger CAN Wake-up Filter	120	_	_	ns	_

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

**TABLE 26-43: ADC MODULE SPECIFICATIONS** 

AC CHA	ARACTER	RISTICS	Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)  Operating temperature $-40^{\circ}\text{C} \leq \text{TA} \leq +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \leq \text{TA} \leq +125^{\circ}\text{C}$ for Extended								
Param No.	Symbol	Characteristic	Min.	Тур	Max.	Units	Conditions				
Device Supply											
AD01	AVDD	Module VDD Supply	Greater of VDD – 0.3 or 3.0	_	Lesser of VDD + 0.3 or 3.6	V	_				
AD02	AVss	Module Vss Supply	Vss - 0.3	ı	Vss + 0.3	>	_				
			Reference	ce Inpu	ts						
AD05	VREFH	Reference Voltage High	AVss + 2.5	I	AVDD	>	_				
AD05a			3.0	_	3.6	V	VREFH = AVDD VREFL = AVSS = 0				
AD06	VREFL	Reference Voltage Low	AVss	_	AVDD - 2.5	V	_				
AD06a			0	_	0	V	VREFH = AVDD VREFL = AVSS = 0				
AD07	VREF	Absolute Reference Voltage	2.5	_	3.6	V	VREF = VREFH - VREFL				
AD08	IREF	Current Drain	_	_	10	μΑ	ADC off				
AD08a	IAD	Operating Current	_	7.0 2.7	9.0 3.2	mA mA	10-bit ADC mode, see <b>Note 1</b> 12-bit ADC mode, see <b>Note 1</b>				
			Analog	g Input							
AD12	VINH	Input Voltage Range Vілн	VINL	_	VREFH	٧	This voltage reflects Sample and Hold Channels 0, 1, 2 and 3 (CH0-CH3), positive input				
AD13	VINL	Input Voltage Range VINL	VREFL	_	AVss + 1V	٧	This voltage reflects Sample and Hold Channels 0, 1, 2 and 3 (CH0-CH3), negative input				
AD17	RIN	Recommended Impedance of Analog Voltage Source	_	_	200 200	Ω Ω	10-bit ADC 12-bit ADC				

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

#### TABLE 26-44: ADC MODULE SPECIFICATIONS (12-BIT MODE)<sup>(1)</sup>

AC CHA	AC CHARACTERISTICS			Standard Operating Conditions: 3.0V 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				
ADC Accuracy (12-Bit Mode) – Measurements with External VREF+/VREF-											
AD20a	Nr	Resolution	12	2 data bit	ts	bits	_				
AD21a	INL	Integral Nonlinearity	-2	1	+2	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V				
AD22a	DNL	Differential Nonlinearity	>-1	1	<1	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V				
AD23a	GERR	Gain Error	_	3.4	10	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V				
AD24a	EOFF	Offset Error	Q	0.9	5	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V				
AD25a	_	Monotonicity	_			_	Guaranteed				
		ADC Accuracy (12-Bit Mo	de) – Mea	asureme	nts with	Interna	I VREF+/VREF-				
AD20b	Nr	Resolution	12	2 data bit	ts	bits	_				
AD21b	INL	Integral Nonlinearity	-2	_	+2	LSb	VINL = AVSS = 0V, AVDD = 3.6V				
AD22b	DNL	Differential Nonlinearity	>-1	_	<1	LSb	VINL = AVSS = 0V, AVDD = 3.6V				
AD23b	GERR	Gain Error	_	10.5	20	LSb	VINL = AVSS = 0V, AVDD = 3.6V				
AD24b	Eoff	Offset Error	_	3.8	10	LSb	VINL = AVSS = 0V, AVDD = 3.6V				
AD25b	_	Monotonicity	_	_	_	_	Guaranteed				
	•	Dynamic	Perform	nance (1	2-Bit Mo	de)					
AD30a	THD	Total Harmonic Distortion	_	_	-75	dB	_				
AD31a	SINAD	Signal to Noise and Distortion	68.5	69.5	_	dB	_				
AD32a	SFDR	Spurious Free Dynamic Range	80		_	dB	_				
AD33a	FNYQ	Input Signal Bandwidth			250	kHz					
AD34a	ENOB	Effective Number of Bits	11.09	11.3	_	bits	_				

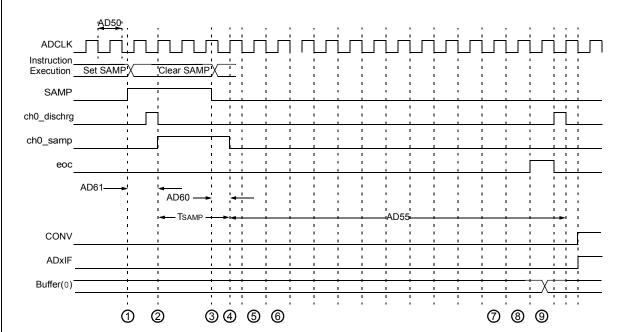
Note 1: Injection currents > | 0 | can affect the ADC results by approximately 4-6 counts.

TABLE 26-45: ADC MODULE SPECIFICATIONS (10-BIT MODE)<sup>(1)</sup>

AC CHA	RACTERIS	TICS	Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq \text{TA} \leq +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \leq \text{TA} \leq +125^{\circ}\text{C}$ for Extended									
Param No.	Symbol	Characteristic	Min.	Тур	Max.	Units	Conditions					
	ADC Accuracy (10-Bit Mode) – Measurements with External VREF+/VREF-											
AD20c	Nr	Resolution	10	) data bi	ts	bits	_					
AD21c	INL	Integral Nonlinearity	-1.5	_	+1.5	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V					
AD22c	DNL	Differential Nonlinearity	>-1	_	<1	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V					
AD23c	GERR	Gain Error	_	3	6	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V					
AD24c	EOFF	Offset Error	_	2	5	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V					
AD25c	_	Monotonicity	_	_	_	_	Guaranteed					
	ı	ADC Accuracy (10-Bit Mode	e) – Meas	uremen	ts with l	nternal '	VREF+/VREF-					
AD20d	Nr	Resolution	10	data bi	ts	bits	_					
AD21d	INL	Integral Nonlinearity	-1		+1	LSb	VINL = AVSS = 0V, AVDD = 3.6V					
AD22d	DNL	Differential Nonlinearity	>-1		<1	LSb	VINL = AVSS = 0V, AVDD = 3.6V					
AD23d	GERR	Gain Error	_	7	15	LSb	VINL = AVSS = 0V, AVDD = 3.6V					
AD24d	Eoff	Offset Error	_	3	7	LSb	VINL = AVSS = 0V, AVDD = 3.6V					
AD25d	_	Monotonicity	_	_	_	_	Guaranteed					
		Dynamic I	Performa	nce (10-	Bit Mod	e)						
AD30b	THD	Total Harmonic Distortion	_	_	-64	dB	_					
AD31b	SINAD	Signal to Noise and Distortion	57	58.5	_	dB	_					
AD32b	SFDR	Spurious Free Dynamic Range	72	_	_	dB	_					
AD33b	FNYQ	Input Signal Bandwidth			550	kHz	_					
AD34b	ENOB	Effective Number of Bits	9.16	9.4	_	bits	_					

**Note 1:** Injection currents > | 0 | can affect the ADC results by approximately 4-6 counts.

FIGURE 26-27: ADC CONVERSION (12-BIT MODE) TIMING CHARACTERISTICS (ASAM = 0, SSRC<2:0> = 000)



- (1) Software sets ADxCON. SAMP to start sampling.
- 2 Sampling starts after discharge period. TSAMP is described in Section 16. "10/12-bit ADC with DMA" in the "dsPIC33F Family Reference Manual".
- 3 Software clears ADxCON. SAMP to start conversion.
- 4 Sampling ends, conversion sequence starts.
- (5) Convert bit 11.
- (6) Convert bit 10.
- 7 Convert bit 1.
- (8) Convert bit 0.
- (9) One TAD for end of conversion.

# TABLE 26-46: ADC CONVERSION (12-BIT MODE) TIMING REQUIREMENTS

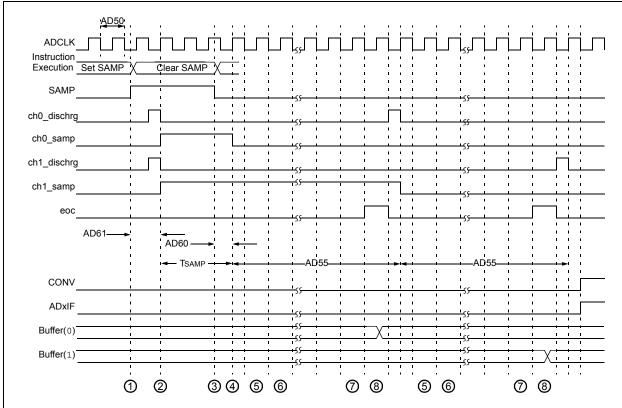
AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{Ta} \le +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \le \text{Ta} \le +125^{\circ}\text{C}$ for Extended				
Param No.	Symbol	Characteristic	Min. Typ Max. Units Conditions				
		Clock	k Paramet	ers			
AD50a	TAD	ADC Clock Period	117.6		_	ns	_
AD51a	trc	ADC Internal RC Oscillator Period	_	250	_	ns	_
Conversion Rate							
AD55a	tconv	Conversion Time	_	14 TAD		_	_
AD56a	FCNV	Throughput Rate	_	_	500	ksps	_
AD57a	TSAMP	Sample Time	3.0 TAD	_	_	_	_
		Timin	g Parame	ters			
AD60a	tpcs	Conversion Start from Sample Trigger <sup>(1,2)</sup>	2.0 TAD	_	3.0 TAD		_
AD61a	tpss	Sample Start from Setting Sample (SAMP) bit (1,2)	2.0 TAD	_	3.0 TAD	_	_
AD62a	tcss	Conversion Completion to Sample Start (ASAM = $1$ ) <sup>(1,2)</sup>	_	0.5 TAD	_	_	_
AD63a	tDPU	Time to Stabilize Analog Stage from ADC Off to ADC On <sup>(1,2,3)</sup>	_	_	20	μS	_

**Note 1:** Because the sample caps will eventually lose charge, clock rates below 10 kHz can affect linearity performance, especially at elevated temperatures.

<sup>2:</sup> These parameters are characterized but not tested in manufacturing.

**<sup>3:</sup>** tDPU is the time required for the ADC module to stabilize when it is turned on (AD1CON1<ADON> = 1). During this time, the ADC result is indeterminate.

FIGURE 26-28: ADC CONVERSION (10-BIT MODE) TIMING CHARACTERISTICS (CHPS<1:0> = 01, SIMSAM = 0, ASAM = 0, SSRC<2:0> = 000)



- 1 Software sets ADxCON. SAMP to start sampling.
- 2 Sampling starts after discharge period. TSAMP is described in Section 16. "10/12-bit ADC with DMA" in the "dsPIC33F Family Reference Manual".
- 3 Software clears ADxCON. SAMP to start conversion.
- 4 Sampling ends, conversion sequence starts.
- (5) Convert bit 9.
- 6 Convert bit 8.
- (7) Convert bit 0.
- (8) One TAD for end of conversion.

FIGURE 26-29: ADC CONVERSION (10-BIT MODE) TIMING CHARACTERISTICS (CHPS<1:0> = 01, SIMSAM = 0, ASAM = 1, SSRC<2:0> = 111, SAMC<4:0> = 00001)

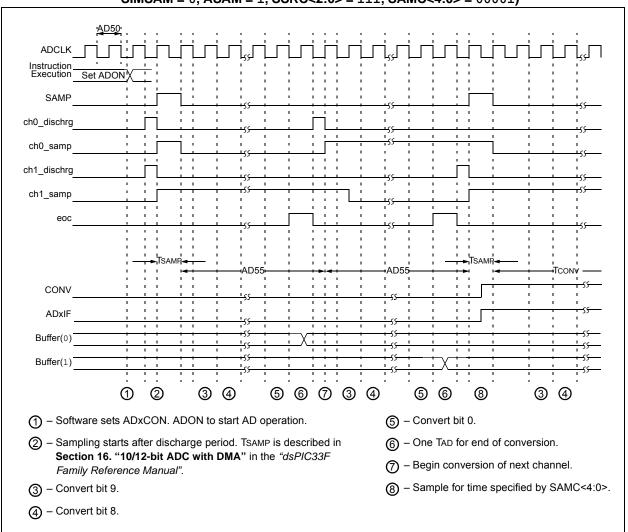


TABLE 26-47: ADC CONVERSION (10-BIT MODE) TIMING REQUIREMENTS

AC CH	AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{TA} \le +85^{\circ}\text{C}$				
Param No.	Symbol	Characteristic	Min.	Typ <sup>(1)</sup>	Max.	Units	Conditions	
		Cloc	k Parame	ters				
AD50b	TAD	ADC Clock Period	76	_	_	ns	_	
AD51b	trc	ADC Internal RC Oscillator Period	_	250		ns	_	
		Con	version R	ate				
AD55b	tconv	Conversion Time	_	12 TAD			_	
AD56b	FCNV	Throughput Rate	_	_	1.1	Msps	_	
AD57b	TSAMP	Sample Time	2 TAD	_	_	_	_	
		Timin	g Parame	eters				
AD60b	tpcs	Conversion Start from Sample Trigger <sup>(1,2)</sup>	2.0 TAD	_	3.0 TAD	_	Auto-Convert Trigger (SSRC<2:0> = 111) not selected	
AD61b	tpss	Sample Start from Setting Sample (SAMP) bit <sup>(1,2)</sup>	2.0 TAD	_	3.0 TAD	_	_	
AD62b	tcss	Conversion Completion to Sample Start (ASAM = $1$ ) <sup>(1,2)</sup>	_	0.5 TAD	_	_	_	
AD63b	tDPU	Time to Stabilize Analog Stage from ADC Off to ADC On <sup>(1,3)</sup>	_	_	20	μS	_	

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

- **2:** Because the sample caps will eventually lose charge, clock rates below 10 kHz can affect linearity performance, especially at elevated temperatures.
- **3:** tDPU is the time required for the ADC module to stabilize when it is turned on (AD1CON1<ADON> = 1). During this time, the ADC result is indeterminate.

### TABLE 26-48: DMA READ/WRITE TIMING REQUIREMENTS

AC CHA	ARACTERISTICS	(unless o	Standard Operating Conditions: 3.0V 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. Typ Max. Units Conditions						
DM1a	DMA Read/Write Cycle Time	_	_	2 Tcy	ns	This characteristic applies to dsPIC33FJ256MCX06A/X08A/X10A devices only.		
DM1b	DMA Read/Write Cycle Time	1 Tcy ns T			ns	This characteristic applies to all devices with the exception of the dsPIC33FJ256MCX06A/X08A/X10A.		

### 27.0 HIGH TEMPERATURE ELECTRICAL CHARACTERISTICS

This section provides an overview of dsPIC33FJXXXMCX06A/X08A/X10A electrical characteristics for devices operating in an ambient temperature range of -40°C to +150°C.

The specifications between -40°C to +150°C are identical to those shown in **Section 26.0 "Electrical Characteristics"** for operation between -40°C to +125°C, with the exception of the parameters listed in this section.

Parameters in this section begin with an H, which denotes High temperature. For example, parameter DC10 in **Section 26.0 "Electrical Characteristics"** is the Industrial and Extended temperature equivalent of HDC10.

Absolute maximum ratings for the dsPIC33FJXXXMCX06A/X08A/X10A high temperature devices are listed below. Exposure to these maximum rating conditions for extended periods can 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<sup>(1)</sup>

Ambient temperature under bias <sup>(4)</sup>	40°C to +150°C
Storage temperature	65°C to +160°C
Voltage on VDD with respect to Vss	0.3V to +4.0V
Voltage on any pin that is not 5V tolerant with respect to Vss <sup>(5)</sup>	0.3V to (VDD + 0.3V)
Voltage on any 5V tolerant pin with respect to Vss when VDD < 3.0V <sup>(5)</sup>	0.3V to (VDD + 0.3V)
Voltage on any 5V tolerant pin with respect to Vss when $VDD \ge 3.0V^{(5)}$	0.3V to 5.6V
Voltage on VCAP with respect to Vss	
Maximum current out of Vss pin	60 mA
Maximum current into VDD pin <sup>(2)</sup>	
Maximum junction temperature	+155°C
Maximum current sourced/sunk by any 2x I/O pin <sup>(3)</sup>	2 mA
Maximum current sourced/sunk by any 4x I/O pin <sup>(3)</sup>	4 mA
Maximum current sourced/sunk by any 8x I/O pin <sup>(3)</sup>	8 mA
Maximum current sunk by all ports combined	10 mA
Maximum current sourced by all ports combined <sup>(2)</sup>	10 mA

- **Note 1:** Stresses above those listed under "Absolute Maximum Ratings" can 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 can affect device reliability.
  - 2: Maximum allowable current is a function of device maximum power dissipation (see Table 27-2).
  - **3:** Unlike devices at 125°C and below, the specifications in this section also apply to the CLKOUT, VREF+, VREF-, SCLx, SDAx, PGECx, and PGEDx pins.
  - **4:** AEC-Q100 reliability testing for devices intended to operate at 150°C is 1,000 hours. Any design in which the total operating time from 125°C to 150°C will be greater than 1,000 hours is not warranted without prior written approval from Microchip Technology Inc.
  - 5: Refer to the "Pin Diagrams" section for 5V tolerant pins.

### 27.1 High Temperature DC Characteristics

# TABLE 27-1: OPERATING MIPS VS. VOLTAGE

Characteristic	V <sub>DD</sub> Range	Temperature Range	Max MIPS
Characteristic	(in Volts)	(in °C)	dsPIC33FJXXXMCX06A/X08A/X10A
HDC5	VBOR to 3.6V <sup>(1)</sup>	-40°C to +150°C	20

**Note 1:** Device is functional at VBORMIN < VDD < VDDMIN. Analog modules such as the ADC will have degraded performance. Device functionality is tested but not characterized. Refer to parameter BO10 in Table 26-11 for the minimum and maximum BOR values.

### **TABLE 27-2: THERMAL OPERATING CONDITIONS**

Rating	Symbol	Min	Тур	Max	Unit
High Temperature Devices					
Operating Junction Temperature Range	TJ	-40	_	+155	°C
Operating Ambient Temperature Range	TA	-40	_	+150	°C
Power Dissipation: Internal chip power dissipation: $PINT = VDD \ x \ (IDD - \Sigma \ IOH)$ I/O Pin Power Dissipation: $I/O = \Sigma \ (\{VDD - VOH\} \ x \ IOH) + \Sigma \ (VOL \ x \ IOL)$	Pb	ı	W		
Maximum Allowed Power Dissipation	PDMAX	(	TJ - TA)/θJ	A	W

# TABLE 27-3: DC TEMPERATURE AND VOLTAGE SPECIFICATIONS

DC CHARACTERISTICS Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{Ta} \le +150^{\circ}\text{C}$ for High									
Parameter No.	Symbol	Characteristic	Min Typ Max Units Conditions						
Operating \	/oltage								
HDC10	Supply Vo	Supply Voltage							
	VDD	— 3.0 3.3 3.6 V -40°C to +150°C							

### TABLE 27-4: DC CHARACTERISTICS: POWER-DOWN CURRENT (IPD)

DC CHARACT	ERISTICS		(unless oth	Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{TA} \le +150^{\circ}\text{C}$ for High Temperature					
Parameter No.	Typical	Max	Units	Units Conditions					
Power-Down (	Current (IPD)								
HDC60e	250	2000	μΑ	+150°C 3.3V Base Power-Down Current <sup>(1,3)</sup>					

- **Note 1:** 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, and VREGS (RCON<8>) = 1.
  - 2: The  $\Delta$  current is the additional current consumed when the module is enabled. This current should be added to the base IPD current.
  - 3: These currents are measured on the device containing the most memory in this family.
  - 4: These parameters are characterized, but are not tested in manufacturing.

DC CHARACT	ERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{TA} \le +150^{\circ}\text{C}$ for High Temperature				
Parameter No.	Typical	Max	Units Conditions				
Power-Down Current (IPD)							
HDC61c	3	5	μА	+150°C 3.3V Watchdog Timer Current: ΔIWDT <sup>(2,4)</sup>			

- **Note 1:** 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, and VREGS (RCON<8>) = 1.
  - 2: The  $\Delta$  current is the additional current consumed when the module is enabled. This current should be added to the base IPD current.
  - 3: These currents are measured on the device containing the most memory in this family.
  - 4: These parameters are characterized, but are not tested in manufacturing.

# TABLE 27-5: DC CHARACTERISTICS: DOZE CURRENT (IDOZE)

DC CHARA	Standard Operating (unless otherwise st Operating temperature						<b>V</b> C for High Temperature	
Parameter No.	Typical <sup>(1)</sup>	Max	Doze Ratio	Units	Conditions			
HDC72a	39	45	1:2	mA				
HDC72f	18	25	1:64	mA	+150°C 3.3V 20 MIPS		20 MIPS	
HDC72g	18	25	1:128	mA				

Note 1: Parameters with Doze ratios of 1:2 and 1:64 are characterized, but are not tested in manufacturing.

TABLE 27-6: DC CHARACTERISTICS: I/O PIN OUTPUT SPECIFICATIONS

DC CHAI		ISTICS	Standa (unles	ard Ope	rating C	Condition (ted)	ns: 3.0V to 3.6V $C \le TA \le +85^{\circ}C$ for High perature
Param.	Symbol	I Characteristic Min. Typ. Max. Ur		Units	Conditions		
		Output Low Voltage I/O Pins: 2x Sink Driver Pins - All pins not defined by 4x or 8x driver pins	_	_	0.4	V	IoL ≤ 1.8 mA, V <sub>DD</sub> = 3.3V See <b>Note 1</b>
HDO10	Vol	Output Low Voltage I/O Pins: 4x Sink Driver Pins - RA2, RA3, RA9, RA10, RA14, RA15, RB0, RB1, RB11, RF4, RF5, RG2, RG3		_	0.4	V	IOL ≤ 3.6 mA, VDD = 3.3V See <b>Note 1</b>
		Output Low Voltage I/O Pins: 8x Sink Driver Pins - OSC2, CLKO, RC15	ı		0.4	٧	IOL ≤ 6 mA, VDD = 3.3V See <b>Note 1</b>
		Output High Voltage I/O Pins: 2x Source Driver Pins - All pins not defined by 4x or 8x driver pins	2.4	_	_	V	IOL ≥ -1.8 mA, VDD = 3.3V See <b>Note 1</b>
HDO20	Vон	Output High Voltage I/O Pins: 4x Source Driver Pins - RA2, RA3, RA9, RA10, RA14, RA15, RB0, RB1, RB11, RF4, RF5, RG2, RG3	2.4	_	_	V	IOL ≥ -3 mA, VDD = 3.3V See <b>Note 1</b>
		Output High Voltage I/O Pins: 8x Source Driver Pins - OSC2, CLKO, RC15	2.4	_	_	V	IOL ≥ -6 mA, VDD = 3.3V See <b>Note 1</b>
		Output High Voltage I/O Pins:	1.5	_	_		IOH ≥ -1.9 mA, VDD = 3.3V See <b>Note 1</b>
		2x Source Driver Pins - All pins not defined by 4x or 8x driver pins	2.0	_	_	V	IOH ≥ -1.85 mA, VDD = 3.3V See <b>Note 1</b>
			3.0	_	_		IOH ≥ -1.4 mA, VDD = 3.3V See <b>Note 1</b>
		Output High Voltage 4x Source Driver Pins - RA2, RA3,	1.5	_	_		IOH ≥ -3.9 mA, VDD = 3.3V See <b>Note 1</b>
HDO20A	HDO20A Voh1	RA9, RA10, RA14, RA15, RB0, RB1, RB11, RF4, RF5, RG2, RG3	2.0	_	_	V	IOH ≥ -3.7 mA, VDD = 3.3V See <b>Note 1</b>
			3.0	_	_		IOH ≥ -2 mA, VDD = 3.3V See <b>Note 1</b>
		Output High Voltage 8x Source Driver Pins - OSC2, CLKO,	1.5	_	_		IOH ≥ -7.5 mA, VDD = 3.3V See <b>Note 1</b>
		RC15	2.0	_	_	V	IOH ≥ -6.8 mA, VDD = 3.3V See <b>Note 1</b>
		tors are characterized, but not tosted	3.0	_	_		IOH ≥ -3 mA, VDD = 3.3V See <b>Note 1</b>

Note 1: Parameters are characterized, but not tested.

# 27.2 AC Characteristics and Timing Parameters

The information contained in this section defines dsPIC33FJXXXMCX06A/X08A/X10A AC characteristics and timing parameters for high temperature devices. However, all AC timing specifications in this section are the same as those in Section 26.2 "AC Characteristics and Timing Parameters", with the exception of the parameters listed in this section.

Parameters in this section begin with an H, which denotes High temperature. For example, parameter OS53 in Section 26.2 "AC Characteristics and Timing Parameters" is the Industrial and Extended temperature equivalent of HOS53.

TABLE 27-7: TEMPERATURE AND VOLTAGE SPECIFICATIONS - AC

AC CHARACTERISTICS	Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)
AC CHARACTERISTICS	Operating temperature $-40^{\circ}\text{C} \le \text{Ta} \le +150^{\circ}\text{C}$ for High Temperature Operating voltage VDD range as described in Table 27-1.

### FIGURE 27-1: LOAD CONDITIONS FOR DEVICE TIMING SPECIFICATIONS

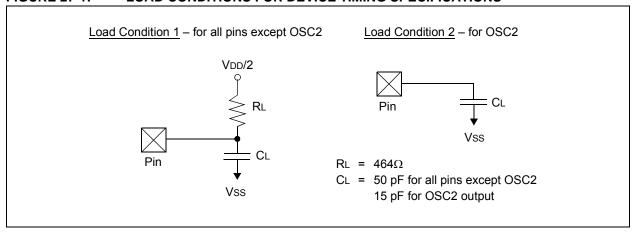


TABLE 27-8: PLL CLOCK TIMING SPECIFICATIONS

_	C TERISTICS	Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{Ta} \le +150^{\circ}\text{C}$ for High Temperature							
Param No.	Symbol	Characteristic	Characteristic Min Typ Max Units Conditions						
HOS53	DCLK	CLKO Stability (Jitter) <sup>(1)</sup> -5 0.5 5 Measured over 100 m period							

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

### TABLE 27-9: INTERNAL LPRC ACCURACY

AC CHARACTERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)  Operating temperature -40°C ≤ TA ≤ +150°C for High Temperature						
Param No.	Characteristic	Characteristic Min Typ Max Units Conditions						
	LPRC @ 32.768 kHz (1)	LPRC @ 32.768 kHz <sup>(1)</sup>						
HF21	LPRC	-70 <sup>(2)</sup>	_	+70(2)	%	-40°C ≤ Ta ≤ +150°C —		

Note 1: Change of LPRC frequency as VDD changes.

# TABLE 27-10: SPIX MASTER MODE (CKE = 0) TIMING REQUIREMENTS

_	AC TERISTICS	Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{Ta} \le +150^{\circ}\text{C}$ for High Temperature								
Param No.	Symbol	nbol Characteristic <sup>(1)</sup> Min Typ Max Un				Units	Conditions			
HSP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	_	10	25	ns	_			
HSP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input 28 — ns to SCKx Edge		_						
HSP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	35	_	_	ns	_			

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

# TABLE 27-11: SPIX MODULE MASTER MODE (CKE = 1) TIMING REQUIREMENTS

	AC CTERISTICS	Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{Ta} \le +150^{\circ}\text{C}$ for High Temperature								
Param No.	Symbol	Characteristic <sup>(1)</sup>	Min	Тур	Max	Units	Conditions			
HSP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	_	10	25	ns	_			
HSP36	TdoV2sc, TdoV2scL	SDOx Data Output Setup to First SCKx Edge	35	_	_	ns	_			
HSP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	28	_	_	ns	_			
HSP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	35	_	_	ns	_			

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

<sup>2:</sup> Characterized but not tested.

TABLE 27-12: SPIx MODULE SLAVE MODE (CKE = 0) TIMING REQUIREMENTS

CHARA	AC CTERISTICS	Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{Ta} \le +150^{\circ}\text{C}$ for High Temperature								
Param No. Symbol		Characteristic <sup>(1)</sup>	Min	Тур	Max	Units	Conditions			
HSP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	_	_	35	ns	_			
HSP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	25	_	ns		_			
HSP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	25	_	_	ns	_			
HSP51	TssH2doZ	SSx ↑ to SDOx Output High-Impedance	15	_	55	ns	See Note 2			

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

TABLE 27-13: SPIX MODULE SLAVE MODE (CKE = 1) TIMING REQUIREMENTS

AC CHARACTERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{Ta} \le +150^{\circ}\text{C}$ for High Temperature								
Param No.	Symbol	Characteristic <sup>(1)</sup>	Min	Тур	Max	Units	Conditions			
HSP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	_	_	35	ns	_			
HSP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	25	_	_	ns	_			
HSP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	25	-	-	ns	_			
HSP51	TssH2doZ	SSx ↑ to SDOx Output 15 — 55 ns See No High-Impedance		See Note 2						
HSP60	TssL2doV	SDOx Data Output Valid after SSx Edge	_	_	55	ns	_			

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

<sup>2:</sup> Assumes 50 pF load on all SPIx pins.

<sup>2:</sup> Assumes 50 pF load on all SPIx pins.

### **TABLE 27-14: ADC MODULE SPECIFICATIONS**

_	AC TERISTICS	Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{Ta} \le +150^{\circ}\text{C}$ for High Temperature						
Param No. Symbol		Characteristic	Min	Тур	Max	Units	Conditions	
		ı	Referenc	e Input	S			
HAD08	IREF	Current Drain	600 50	μ <b>Α</b> μ <b>Α</b>	ADC operating, See Note 1 ADC off, See Note 1			

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

# TABLE 27-15: ADC MODULE SPECIFICATIONS (12-BIT MODE)(3)

		1		l .					
-	AC TERISTICS	Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{TA} \le +150^{\circ}\text{C}$ for High Temperature							
Param No.	Symbol	Characteristic	Min	Тур	Max	Units	Conditions		
ADC Accuracy (12-bit Mode) – Measurements with external VREF+/VREF-(1)									
AD23a	GERR	Gain Error	_	5	10	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V		
AD24a	EOFF	Offset Error		2	5	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V		
	AD	C Accuracy (12-bit Mode	e) – Meas	uremen	ts with in	ternal V	/REF+/VREF- <sup>(1)</sup>		
AD23a	GERR	Gain Error	2	10	20	LSb	VINL = AVSS = 0V, AVDD = 3.6V		
AD24a	Eoff	Offset Error	2	5	10	LSb	VINL = AVSS = 0V, AVDD = 3.6V		
	•	Dynamic	Performa	nce (12	-bit Mode	e) <sup>(2)</sup>			
HAD33a	FNYQ	Input Signal Bandwidth	_	_	200	kHz	_		

- Note 1: These parameters are characterized, but are tested at 20 ksps only.
  - 2: These parameters are characterized by similarity, but are not tested in manufacturing.
  - 3: Injection currents > | 0 | can affect the ADC results by approximately 4-6 counts.

# TABLE 27-16: ADC MODULE SPECIFICATIONS (10-BIT MODE)(3)

-	AC TERISTICS	Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{TA} \le +150^{\circ}\text{C}$ for High Temperature							
Param No.	Symbol	Characteristic	Min	Тур	Max	Units	Conditions		
ADC Accuracy (12-bit Mode) – Measurements with external VREF+/VREF- <sup>(1)</sup>									
AD23b	GERR	Gain Error	_	3	6	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V		
AD24b	EOFF	Offset Error	_	2	5	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V		
	AD	C Accuracy (12-bit Mode)	- Measu	rement	s with int	ernal V	REF+/VREF- <sup>(1)</sup>		
AD23b	GERR	Gain Error	_	7	15	LSb	VINL = AVSS = 0V, AVDD = 3.6V		
AD24b	Eoff	Offset Error	_	3	7	LSb	VINL = AVSS = 0V, AVDD = 3.6V		
		Dynamic Po	erforman	ice (10-l	oit Mode)	(2)			
HAD33b	FNYQ	Input Signal Bandwidth	_	_	400	kHz	_		

- Note 1: These parameters are characterized, but are tested at 20 ksps only.
  - 2: These parameters are characterized by similarity, but are not tested in manufacturing.
  - 3: Injection currents > | 0 | can affect the ADC results by approximately 4-6 counts.

# TABLE 27-17: ADC CONVERSION (12-BIT MODE) TIMING REQUIREMENTS

AC CHARACTERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{Ta} \le +150^{\circ}\text{C}$ for High Temperature							
Param No.	Symbol   Characteristic   Min   Ivn   Max   Units   Conditions						Conditions		
		Clock	( Parame	ters					
HAD50	TAD	ADC Clock Period <sup>(1)</sup> 147 — ns —							
			ersion R	ate					

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

# TABLE 27-18: ADC CONVERSION (10-BIT MODE) TIMING REQUIREMENTS

AC CHARACTERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{Ta} \le +150^{\circ}\text{C}$ for High Temperature							
Param No.	Symbol	Characteristic	Min	Тур	Max	Units	Conditions		
		Cloc	k Parame	ters					
HAD50	TAD	ADC Clock Period <sup>(1)</sup>	104	_	_	ns	_		
	Conversion Rate								
HAD56	FCNV	Throughput Rate <sup>(1)</sup>	_	_	800	Ksps	_		

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

dsPIC3	3FJXX	06A/X	08A/X	10A	
NOTES:					

0.000

0.00

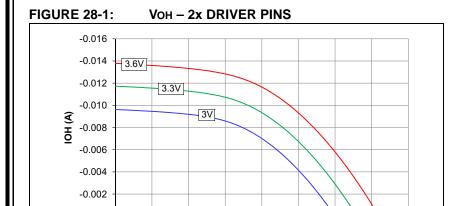
0.50

1.00

# dsPIC33FJXXXMCX06A/X08A/X10A

### 28.0 DC AND AC DEVICE CHARACTERISTICS GRAPHS

**Note:** The graphs provided following this note are a statistical summary based on a limited number of samples and are provided for design guidance purposes only. The performance characteristics listed herein are not tested or guaranteed. In some graphs, the data presented may be outside the specified operating range (e.g., outside specified power supply range) and therefore, outside the warranted range.



1.50

2.00

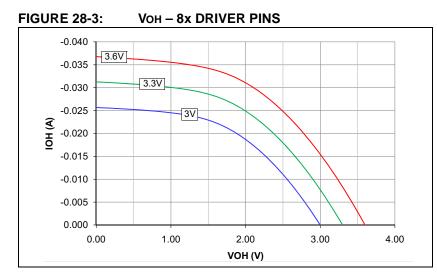
VOH (V)

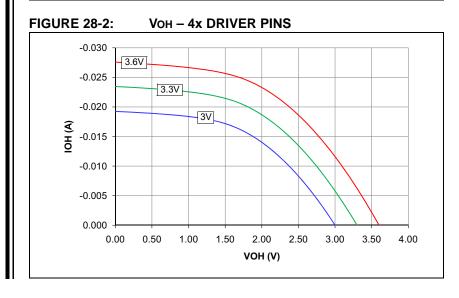
2.50

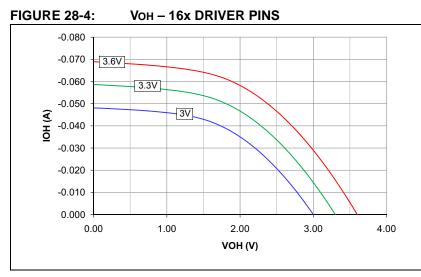
3.00

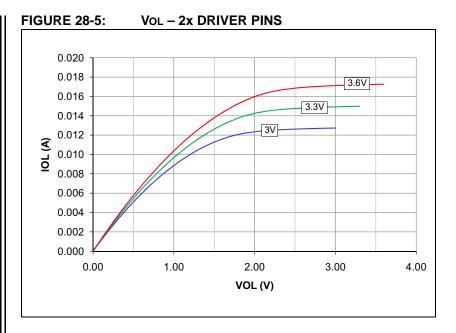
3.50

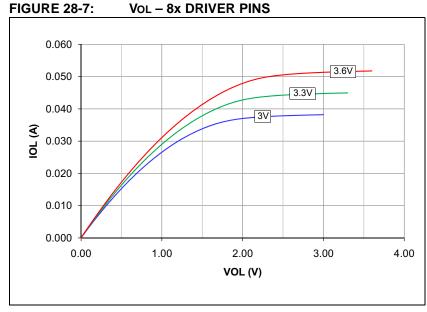
4.00

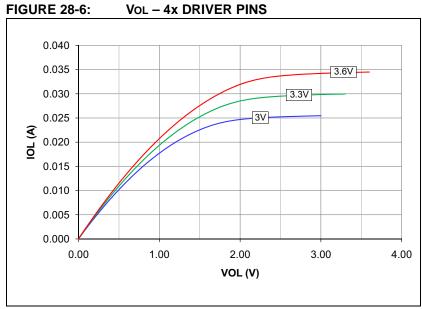


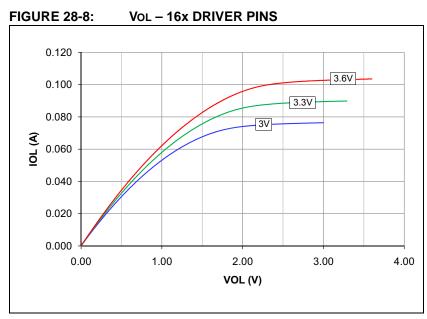




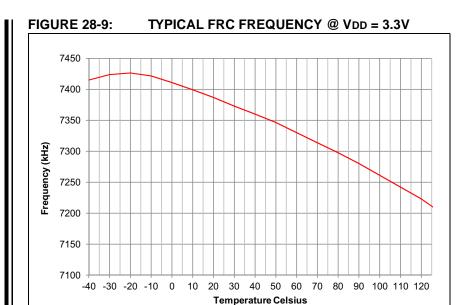


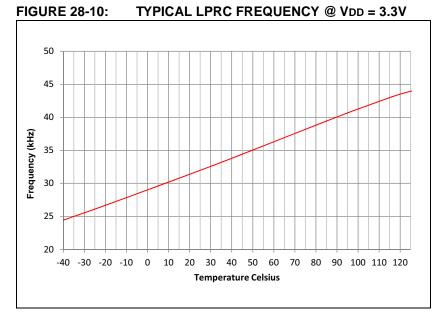












dsPIC33	dsPIC33FJXXXMCX06A/X08A/X10A					
NOTES:						

### 29.0 PACKAGING INFORMATION

# 29.1 Package Marking Information

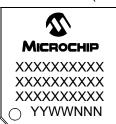
64-Lead QFN (9x9x0.9mm)



### Example



64-Lead TQFP (10x10x1 mm)



### Example



80-Lead TQFP (12x12x1 mm)



### 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
B-b-free JEDEC designator for Matte Tin (Sn)
This package is Pb-free. The Pb-free JEDEC designator (e3)
can be found on the outer packaging for this package.

be carried over to the next line, thus limiting the number of available characters for customer-specific information.

# 29.1 Package Marking Information (Continued)

100-Lead TQFP (12x12x1 mm)



### Example



100-Lead TQFP (14x14x1mm)



### Example



Legend: XX...X Customer-specific information
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

(e3) Pb-free JEDEC designator for Matte Tin (Sn)

This package is Pb-free. The Pb-free JEDEC designator (e3)

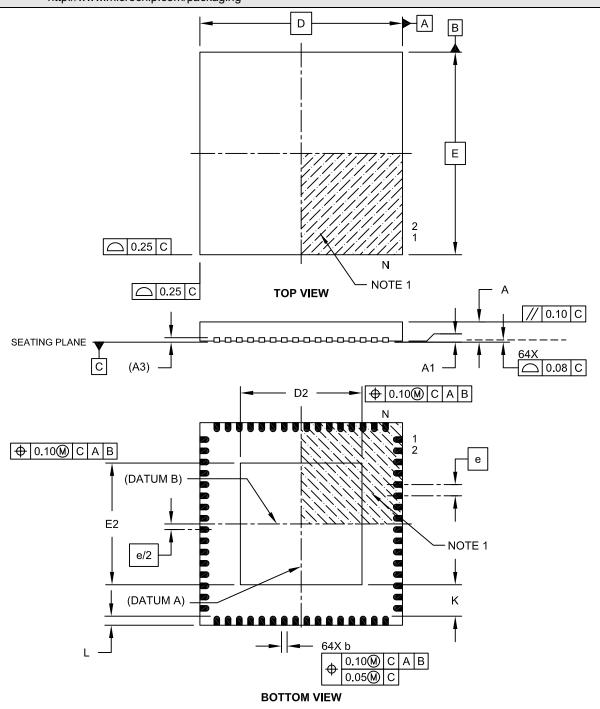
can be found on the outer packaging for this package.

**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.

# 29.2 Package Details

64-Lead Plastic Quad Flat, No Lead Package (MR) – 9x9x0.9 mm Body with 5.40 x 5.40 Exposed Pad [QFN]

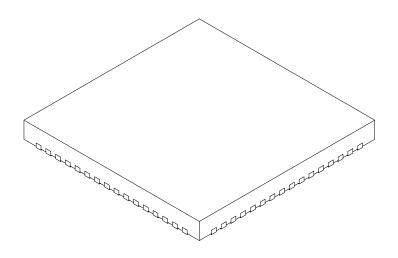
**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Microchip Technology Drawing C04-154A Sheet 1 of 2

# 64-Lead Plastic Quad Flat, No Lead Package (MR) – 9x9x0.9 mm Body with 5.40 x 5.40 Exposed Pad [QFN]

**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 Pins	N		64		
Pitch	е		0.50 BSC		
Overall Height	Α	0.80 0.90 1.00			
Standoff	A1	0.00	0.02	0.05	
Contact Thickness A3		0.20 REF			
Overall Width	Е	9.00 BSC			
Exposed Pad Width	E2	5.30	5.40	5.50	
Overall Length	D		9.00 BSC		
Exposed Pad Length	D2	5.30	5.40	5.50	
Contact Width	b	0.20 0.25 0.30			
Contact Length	L	0.30 0.40 0.50			
Contact-to-Exposed Pad	K	0.20	-	-	

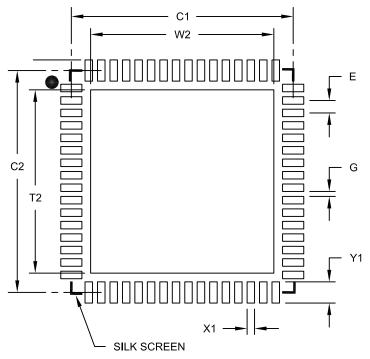
### Notes:

- 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 C04-154A Sheet 2 of 2

64-Lead Plastic Quad Flat, No Lead Package (MR) – 9x9x0.9 mm Body [QFN] 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	MOM	MAX	
Contact Pitch	Е		0.50 BSC		
Optional Center Pad Width	W2			7.35	
Optional Center Pad Length T2				7.35	
Contact Pad Spacing	C1		8.90		
Contact Pad Spacing	C2		8.90		
Contact Pad Width (X64)	X1			0.30	
Contact Pad Length (X64) Y1				0.85	
Distance Between Pads	G	0.20			

### Notes:

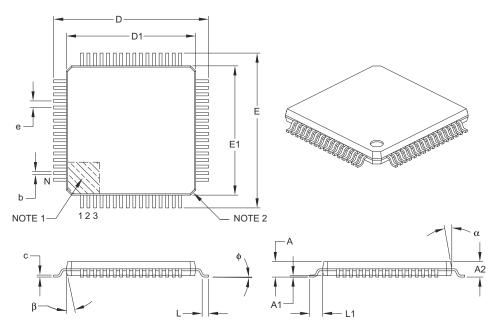
1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2149A

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

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



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

### Notes:

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

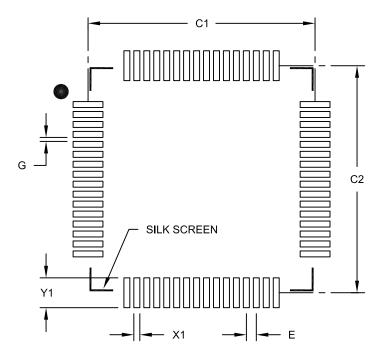
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

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

Microchip Technology Drawing C04-085B

64-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

	MILLIMETERS			
Dimension Limits		MIN	NOM	MAX
Contact Pitch	Е		0.50 BSC	
Contact Pad Spacing	C1	11.40		
Contact Pad Spacing	C2	2 11.40		
Contact Pad Width (X64)	X1			0.30
Contact Pad Length (X64)	Y1			1.50
Distance Between Pads	G	0.20		

### Notes:

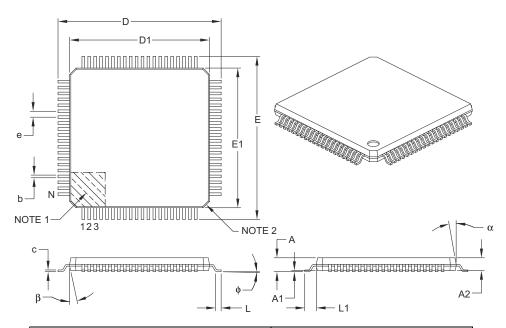
1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2085B

# 80-Lead Plastic Thin Quad Flatpack (PT) – 12x12x1 mm Body, 2.00 mm [TQFP]

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



	Units	MILLIMETERS		
	Dimension Limits	MIN	NOM	MAX
Number of Leads	N		80	
Lead Pitch	е		0.50 BSC	
Overall Height	A	ı	_	1.20
Molded Package Thickness	A2	0.95	1.00	1.05
Standoff	A1	0.05	_	0.15
Foot Length	L	0.45	0.60	0.75
Footprint	L1	1.00 REF		
Foot Angle	ф	0°	3.5°	7°
Overall Width	E		14.00 BSC	
Overall Length	D		14.00 BSC	
Molded Package Width	E1		12.00 BSC	
Molded Package Length	D1	12.00 BSC		
Lead Thickness	С	0.09	_	0.20
Lead Width	b	0.17	0.22	0.27
Mold Draft Angle Top	α	11°	12°	13°
Mold Draft Angle Bottom	β	11°	12°	13°

### Notes:

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

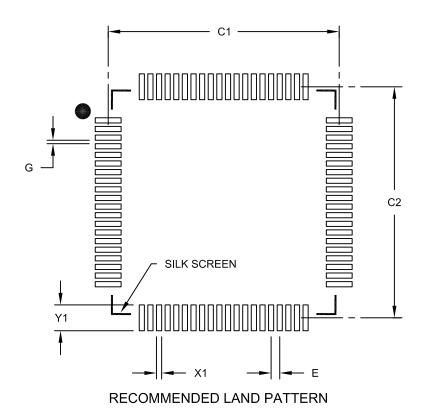
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

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

Microchip Technology Drawing C04-092B

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

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



	MILLIMETERS			
Dimension Limits		MIN	NOM	MAX
Contact Pitch	Е	0.50 BSC		
Contact Pad Spacing	C1		13.40	
Contact Pad Spacing	C2		13.40	
Contact Pad Width (X80)	X1			0.30
Contact Pad Length (X80)	Y1	·		1.50
Distance Between Pads	G	0.20		

### Notes

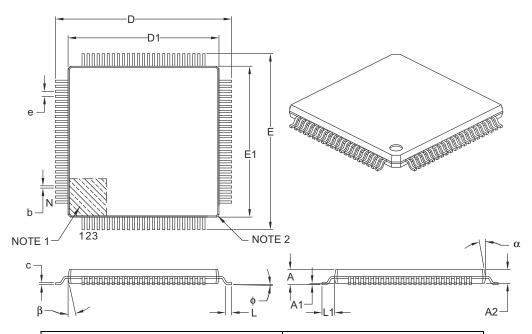
1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2092B

# 100-Lead Plastic Thin Quad Flatpack (PT) - 12x12x1 mm Body, 2.00 mm [TQFP]

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



Units		MILLIMETERS		
Dimension	Dimension Limits		NOM	MAX
Number of Leads	N		100	
Lead Pitch	е		0.40 BSC	
Overall Height	Α	_	_	1.20
Molded Package Thickness	A2	0.95	1.00	1.05
Standoff	A1	0.05	_	0.15
Foot Length	L	0.45	0.60	0.75
Footprint	L1	1.00 REF		
Foot Angle	ф	0°	3.5°	7°
Overall Width	Е		14.00 BSC	
Overall Length	D		14.00 BSC	
Molded Package Width	E1		12.00 BSC	
Molded Package Length	D1	12.00 BSC		
Lead Thickness	С	0.09	-	0.20
Lead Width	b	0.13	0.18	0.23
Mold Draft Angle Top	α	11°	12°	13°
Mold Draft Angle Bottom	β	11°	12°	13°

### Notes:

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

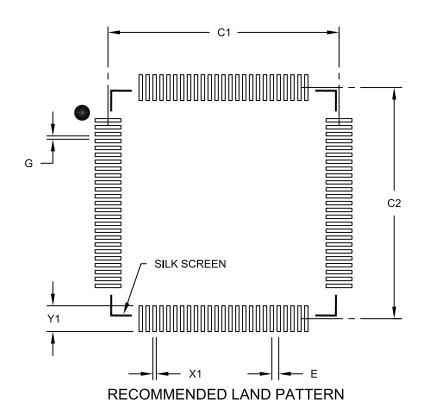
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

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

Microchip Technology Drawing C04-100B

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

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



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

### Notes:

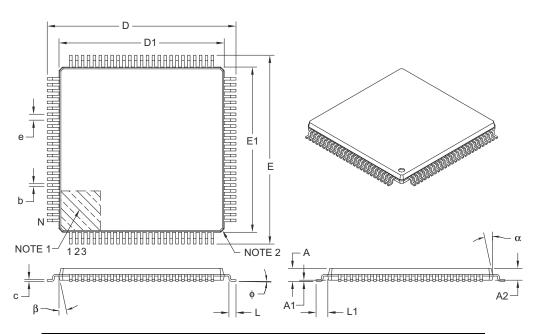
1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2100B

# 100-Lead Plastic Thin Quad Flatpack (PF) - 14x14x1 mm Body, 2.00 mm [TQFP]

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



	Units	MILLIMETERS		
	Dimension Limits	MIN	NOM	MAX
Number of Leads	N		100	
Lead Pitch	е		0.50 BSC	
Overall Height	A	ı	_	1.20
Molded Package Thickness	A2	0.95	1.00	1.05
Standoff	A1	0.05	_	0.15
Foot Length	L	0.45	0.60	0.75
Footprint	L1		1.00 REF	
Foot Angle	ф	0°	3.5°	7°
Overall Width	E		16.00 BSC	
Overall Length	D		16.00 BSC	
Molded Package Width	E1		14.00 BSC	
Molded Package Length	D1		14.00 BSC	
Lead Thickness	С	0.09	_	0.20
Lead Width	b	0.17	0.22	0.27
Mold Draft Angle Top	α	11°	12°	13°
Mold Draft Angle Bottom	β	11°	12°	13°

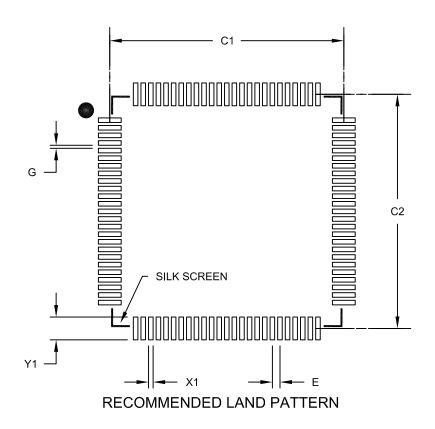
### Notes:

- 1. Pin 1 visual index feature may vary, but must be located within the hatched area.
- 2. Chamfers at corners are optional; size may vary.
- 3. Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions 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.

Microchip Technology Drawing C04-110B

100-Lead Plastic Thin Quad Flatpack (PF) - 14x14x1 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



	MILLIMETERS			
Dimension Limits		MIN	NOM	MAX
Contact Pitch	Е	0.50 BSC		
Contact Pad Spacing	C1	15.40		
Contact Pad Spacing	C2	2 15.40		
Contact Pad Width (X100)	X1			0.30
Contact Pad Length (X100)	Y1	1.50		
Distance Between Pads	G	0.20		

### Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2110B

dsPIC33FJXXXWCXU6A/XU8A/X1UA						
NOTES:						

APPENDIX A: MIGRATING FROM dsPIC33FJXXXMCX06/ X08/X10 DEVICES TO dsPIC33FJXXXMCX06A/ X08A/X10A DEVICES

The dsPIC33FJXXXMCX06A/X08A/X10A devices were designed to enhance the dsPIC33FJXXXMCX06/X08/X10 families of devices.

In general, the dsPIC33FJXXXMCX06A/X08A/X10A devices are backward-compatible with dsPIC33FJXXXMCX06/X08/X10 devices; however, manufacturing differences may cause dsPIC33FJXXXMCX06A/X08A/X10A devices to behave differently from dsPIC33FJXXXMCX06/X08/X10 devices. Therefore, complete system test and characterization is recommended if dsPIC33FJXXXMCX06A/X08A/X10A devices are used to replace dsPIC33FJXXXMCX06/X08/X10 devices.

The following enhancements were introduced:

- Extended temperature support of up to +125°C
- Enhanced Flash module with higher endurance and retention
- · New PLL Lock Enable Configuration bit
- Added Timer5 trigger for ADC1 and Timer3 trigger for ADC2

### APPENDIX B: REVISION HISTORY

# Revision A (May 2009)

This is the initial released version of the document.

# **Revision B (October 2009)**

The revision includes the following global update:

 Added Note 2 to the shaded table that appears at the beginning of each chapter. This new note provides information regarding the availability of registers and their associated bits.

This revision also includes minor typographical and formatting changes throughout the data sheet text.

All other major changes are referenced by their respective section in the following table.

### TABLE B-1: MAJOR SECTION UPDATES

Ocadien Neura	
Section Name	Update Description
"16-bit Digital Signal Controllers (up to 256 KB Flash and 30 KB SRAM) with Motor	Added information on high temperature operation (see "Operating Range:").
Control and Advanced Analog"	
Section 11.0 "I/O Ports"	Changed the reference to digital-only pins to 5V tolerant pins in the second paragraph of <b>Section 11.2 "Open-Drain Configuration"</b> .
Section 20.0 "Universal Asynchronous Receiver Transmitter (UART)"	Updated the two baud rate range features to: 10 Mbps to 38 bps at 40 MIPS.
Section 22.0 "10-bit/12-bit Analog-to-Digital Converter (ADC)"	Updated the ADCx block diagram (see Figure 22-1).
Section 23.0 "Special Features"	Updated the second paragraph and removed the fourth paragraph in <b>Section 23.1 "Configuration Bits"</b> .
	Updated the Device Configuration Register Map (see Table 23-1).
Section 26.0 "Electrical Characteristics"	Updated the Absolute Maximum Ratings for high temperature and added Note 4.
	Updated Power-Down Current parameters DC60d, DC60a, DC60b, and DC60d (see Table 26-7).
	Added I2Cx Bus Data Timing Requirements (Master Mode) parameter IM51 (see Table 26-40).
	Updated the SPIx Module Slave Mode (CKE = 1) Timing Characteristics (see Figure 26-17).
	Updated the Internal LPRC Accuracy parameters (see Table 26-19).
	Updated the ADC Module Specifications (12-bit Mode) parameters AD23a, AD24a, AD23b, and AD24b (see Table 26-46).
	Updated the ADC Module Specifications (10-bit Mode) parameters AD23c, AD24c, AD23d, and AD24d (see Table 26-46).
Section 27.0 "High Temperature Electrical Characteristics"	Added new chapter with high temperature specifications.
"Product Identification System"	Added the "H" definition for high temperature.

# Revision C (March 2011)

This revision includes typographical and formatting changes throughout the data sheet text. In addition, all instances of VDDCORE have been removed.

All other major changes are referenced by their respective section in the following table.

TABLE B-2: MAJOR SECTION UPDATES

Section Name	Update Description
Section 2.0 "Guidelines for Getting Started with 16-bit Digital Signal Controllers"	Updated the title of Section 2.3 "CPU Logic Filter Capacitor Connection (VCAP)".
	The frequency limitation for device PLL start-up conditions was updated in Section 2.7 "Oscillator Value Conditions on Device Start-up".
	The second paragraph in Section 2.9 "Unused I/Os" was updated.
Section 4.0 "Memory Organization"	The All Resets values for the following SFRs in the Timer Register Map were changed (see Table 4-6):  • TMR1  • TMR2  • TMR3  • TMR4  • TMR5  • TMR6  • TMR7  • TMR8
Section 9.0 "Oscillator Configuration"	Added Note 3 to the OSCCON: Oscillator Control Register (see Register 9-1).
	Added Note 2 to the CLKDIV: Clock Divisor Register (see Register 9-2).  Added Note 1 to the PLLFBD: PLL Feedback Divisor Register (see
	Register 9-3).
	Added Note 2 to the OSCTUN: FRC Oscillator Tuning Register (see Register 9-4).
Section 22.0 "10-bit/12-bit Analog-to-Digital Converter (ADC)"	Updated the VREFL references in the ADC1 module block diagram (see Figure 22-1).
Section 23.0 "Special Features"	Added a new paragraph and removed the third paragraph in Section 23.1 "Configuration Bits".
	Added the column "RTSP Effects" to the Configuration Bits Descriptions (see Table 23-2).

TABLE B-2: MAJOR SECTION UPDATES (CONTINUED)

Section Name	Update Description
Section 26.0 "Electrical Characteristics"	Removed Note 4 from the DC Temperature and Voltage Specifications (see Table 26-4).
	Updated the maximum value for parameter DI19 and added parameters DI28, DI29, DI60a, DI60b, and DI60c to the I/O Pin Input Specifications (see Table 26-9).
	Removed Note 2 from the AC Characteristics: Internal RC Accuracy (see Table 26-18).
	Updated the characteristic description for parameter DI35 in the I/O Timing Requirements (see Table 26-20).
	Updated the ADC Module Specification minimum values for parameters AD05 and AD07, and updated the maximum value for parameter AD06 (see Table 26-43).
	Added Note 1 to the ADC Module Specifications (12-bit Mode) (see Table 26-44).
	Added Note 1 to the ADC Module Specifications (10-bit Mode) (see Table 26-45).
	Added DMA Read/Write Timing Requirements (see Table 26-48).
Section 27.0 "High Temperature Electrical Characteristics"	Updated all ambient temperature end range values to +150°C throughout the chapter.
	Updated the storage temperature end range to +160°C.
	Updated the maximum junction temperature from +145°C to +155°C.
	Updated the maximum values for High Temperature Devices in the Thermal Operating Conditions (see Table 27-2).
	Updated the ADC Module Specifications (12-bit Mode), removing all parameters with the exception of HAD33a (see Table 27-14).
	Updated the ADC Module Specifications (10-bit Mode), removing all parameters with the exception of HAD33b (see Table 27-16).

### **Revision D (June 2012)**

This revision includes typographical and formatting changes throughout the data sheet text.

All other major changes are referenced by their respective section in the following table.

TABLE B-3: MAJOR SECTION UPDATES

Section Name	Update Description
Section 2.0 "Guidelines for Getting Started with 16-bit Digital Signal Controllers"	Updated the Recommended Minimum Connection (see Figure 2-1).
Section 9.0 "Oscillator Configuration"	Updated the COSC<2:0> and NOSC<2:0> bit value definitions for '001' (see Register 9-1).
Section 22.0 "10-bit/12-bit Analog-to-Digital Converter (ADC)"	Updated the Analog-to-Digital Conversion Clock Period Block Diagram (see Figure 22-2).
Section 23.0 "Special Features"	Added Note 3 to the On-chip Voltage Regulator Connections (see Figure 23-1).
Section 26.0 "Electrical Characteristics"	Updated "Absolute Maximum Ratings".
	Updated Operating MIPS vs. Voltage (see Table 26-1).
	Removed parameter DC18 from the DC Temperature and Voltage Specifications (see Table 26-4).
	Updated the notes in the following tables:
	• Table 26-5
	• Table 26-6
	• Table 26-7
	• Table 26-8
	Updated the I/O Pin Output Specifications (see Table 26-10).
	Updated the Conditions for parameter BO10 (see Table 26-11).
	Updated the Conditions for parameters D136b, D137b and D138b (TA = 150°C) (see Table 26-12).
Section 27.0 "High Temperature Electrical	Updated "Absolute Maximum Ratings(1)".
Characteristics"	Updated the I/O Pin Output Specifications (see Table 27-6).
	Removed Table 26-7: DC Characteristics: Program Memory.

dsPIC33FJXXXMCX06A/X08A/X10A					
NOTES:					

#### **INDEX**

A
A/D Converter
DMA
Initialization245
Key Features245
AC Characteristics
ADC Module336
ADC Module (10-bit Mode)
ADC Module (12-bit Mode)
Internal RC Accuracy292
Load Conditions
ADC Module
ADC1 Register Map52
ADC2 Register Map52
Alternate Vector Interrupt Table (AIVT)85
Arithmetic Logic Unit (ALU)29
Assembler
MPASM Assembler276
В
_
Barrel Shifter
Bit-Reversed Addressing
Example
Implementation
Sequence Table (16-Entry)67
Block Diagrams
16-Bit Timer1 Module
A/D Module
Connections for On-Chip Voltage Regulator
Device Clock (Oscillator)
Device Clock (PLL)
DSP Engine
dsPIC33F
dsPIC33F CPU Core
ECAN Technology
I <sup>2</sup> C Module 204
Input Capture         173           Output Compare         175
Programmer's Model
PWM Module 180
Quadrature Encoder Interface
Reset System79
Shared Port Structure
SPI Module
Timer2 (16-Bit)
Top Level System Architecture Using Dedicated
Transaction Bus134
UART Module
Watchdog Timer (WDT)
Brown-out Reset (BOR)
,
C
C Compilers
MPLAB C18276
Clock Switching
Enabling151
Sequence151
Code Examples
Erasing a Program Memory Page76
Initiating a Programming Sequence77
Loading Write Buffers77
Port Write/Read162
PWRSAV Instruction Syntax153

Code Protection	259, 266
CodeGuard Security	259, 266
Configuration Bits	259
Configuration Register Map	259
Configuring Analog Port Pins	
CPU	
Control Register	26
CPU Clocking System	
PLL	
Selection	
Sources	
Customer Change Notification Service	
Customer Notification Service	
Customer Support	369
D	
_	
Data Accumulators and Adder/Subtracter	
Data Space Write Saturation	
Overflow and Saturation	
Round Logic	32
Write Back	32
Data Address Space	37
Alignment	
Memory Map for dsPIC33FJXXXMCX06A/X08	
Devices with 16-Kbyte RAM	
Memory Map for dsPIC33FJXXXMCX06A/X08	
Devices with 30-Kbyte RAM	
Memory Map for dsPIC33FJXXXMCX06A/X08	
Devices with 8-Kbyte RAM	
Near Data Space	
Software Stack	
Width	37
DC and AC Characteristics	
Graphs and Tables	339
DC Characteristics	
Doze Current (IDOZE)	285, 331
High Temperature	330
I/O Pin Input Specifications	286
I/O Pin Output Specifications	
Idle Current (IIDLE)	
Operating Current (IDD)	
Operating MIPS vs. Voltage	
Power-Down Current (IPD)	
Power-down Current (IPD)	
Program Memory	
Temperature and Voltage	
Temperature and Voltage Specifications	
Thermal Operating Conditions	
Development Support	275
DMA Module	
DMA Register Map	
DMAC Registers	135
DMAxCNT	135
DMAxCON	135
DMAxPAD	135
DMAxREQ	
DMAxSTA	
DMAxSTB	
DSP Engine	
Multiplier	20
	21
Multiplier	3′

E	Instruction-Based Power-Saving Modes	153
ECAN Module	ldle	
ECAN1 Register Map (C1CTRL1.WIN = 0 or 1)55	Sleep	153
ECAN1 Register Map (C1CTRL1.WIN = 0)55	Internal RC Oscillator	
ECAN1 Register Map (C1CTRL1.WIN = 1)56	Use with WDT	265
ECAN2 Register Map (C2CTRL1.WIN = 0 or 1)58	Internet Address	369
ECAN2 Register Map (C2CTRL1.WIN = 0) 58, 59	Interrupt Control and Status Registers	89
ECAN Technology	IECx	89
Frame Types	IFSx	89
Modes of Operation	INTCON1	89
Overview217	INTCON2	89
Electrical Characteristics	INTTREG	89
AC	IPCx	89
Enhanced CAN Module217	Interrupt Setup Procedures	131
Equations	Initialization	131
Device Operating Frequency144	Interrupt Disable	131
Fosc Calculation	Interrupt Service Routine (ISR)	131
Programming Time	Trap Service Routine	131
XT with PLL Mode145	Interrupt Vector Table (IVT)	85
Errata	Interrupts Coincident with Power Save Instructions	154
F	J	
Flash Program Memory73	JTAG Boundary Scan Interface	259
Control Registers74	M	
Operations74	<del></del>	
Programming Algorithm	Memory Organization	35
RTSP Operation74	Microchip Internet Web Site	
Table Instructions	Migration	357
Flexible Configuration259	Modes of Operation	
FSCM	Disable	
Delay for Crystal and PLL Clock Sources84	Initialization	
Device Resets	Listen All Messages	
	Listen Only	
G	Loopback Mode	
Getting Started with 16-Bit DSCs	Normal Operation	
-	Modulo Addressing	
н	Applicability	
High Temperature Electrical Characteristics329	Operation Example	
1	Start and End Address	
I	W Address Register Selection	
I/O Ports161	Motor Control PWM	179
Parallel I/O (PIO)161	Motor Control PWM Module	
Write/Read Timing162	8-Output Register Map	
I <sup>2</sup> C	MPLAB ASM30 Assembler, Linker, Librarian	276
Operating Modes203	MPLAB Integrated Development	
I <sup>2</sup> C Module	Environment Software	
I2C1 Register Map50	MPLAB PM3 Device Programmer	
I2C2 Register Map50	MPLAB REAL ICE In-Circuit Emulator System	
In-Circuit Debugger	MPLINK Object Linker/MPLIB Object Librarian	276
In-Circuit Emulation	N	
In-Circuit Serial Programming (ICSP)259, 266		
Input Capture	NVM Module	0.0
Registers174	Register Map	62
Input Change Notification Module	0	
Instruction Addressing Modes	Open-Drain Configuration	161
File Register Instructions63	Output Compare	
Fundamental Modes Supported64	Modes	
MAC Instructions64	IVIUUC5	176
MCU Instructions63	P	
Move and Accumulator Instructions64	Packaging	343
Other Instructions64	Details	
Instruction Set	Marking	
Overview	Peripheral Module Disable (PMD)	
Summary	. Suprioral modulo blouble (1 MD)	13-

Pinout I/O Descriptions (table)	15	CIFCTRL (ECAN FIFO Control)	. 223
PMD Module		CiFEN1 (ECAN Acceptance Filter Enable)	
Register Map	62	CiFIFO (ECAN FIFO Status)	
POR and Long Oscillator Start-up Times		CiFMSKSEL1 (ECAN Filter 7-0 Mask Selection)	
PORTA		CiFMSKSEL2 (ECAN Filter 15-8 Mask	
Register Map	60	Selection)	. 237
PORTB		CilNTE (ECAN Interrupt Enable)	
Register Map	60	CilNTF (ECAN Interrupt Flag)	
PORTC		CiRXFnEID (ECAN Acceptance Filter n	
Register Map	61	Extended Identifier)	. 235
PORTD		CiRXFnSID (ECAN Acceptance Filter n	
Register Map	61	Standard Identifier)	. 235
PORTE		CiRXFUL1 (ECAN Receive Buffer Full 1)	
Register Map	61	CiRXFUL2 (ECAN Receive Buffer Full 2)	
PORTF		CiRXMnEID (ECAN Acceptance Filter	
Register Map	61	Mask n Extended Identifier)	238
PORTG		CiRXMnSID (ECAN Acceptance Filter Mask n	
Register Map	62	Standard Identifier)	238
Power-Saving Features		CiRXOVF1 (ECAN Receive Buffer Overflow 1)	
Clock Frequency and Switching		CiRXOVF2 (ECAN Receive Buffer Overflow 2)	
Program Address Space		CiTRBnDLC (ECAN Buffer n Data	. 240
Construction		Length Control)	245
Data Access from Program Memory Using	00	CiTRBnDm (ECAN Buffer n Data Field Byte m)	
	71	CiTRBnEID (ECAN Buffer n Extended Identifier)	
Program Space Visibility	/ 1	,	
Data Access from Program Memory Using	70	CiTRBnSID (ECAN Buffer n Standard Identifier)	
Table Instructions		CiTRBnSTAT (ECAN Receive Buffer n Status)	
Data Access from, Address Generation		CiTRmnCON (ECAN TX/RX Buffer m Control)	
Memory Map	35	CiVEC (ECAN Interrupt Code)	
Table Read High Instructions		CLKDIV (Clock Divisor)	
TBLRDH	70	CORCON (Core Control)29	8, 90
Table Read Low Instructions		DFLTxCON (Digital Filter x Control)	. 196
TBLRDL	70	DMACS0 (DMA Controller Status 0)	. 139
Visibility Operation	71	DMACS1 (DMA Controller Status 1)	. 141
Program Memory		DMAxCNT (DMA Channel x Transfer Count)	. 138
Interrupt Vector	36	DMAxCON (DMA Channel x Control)	. 135
Organization	36	DMAxPAD (DMA Channel x	
Reset Vector		Peripheral Address)	. 138
		DMAxREQ (DMA Channel x IRQ Select)	
Q		DMAxSTA (DMA Channel x RAM Start	
Quadrature Encoder Interface (QEI)	193	Address Offset A)	. 137
Quadrature Encoder Interface (QEI) Module		DMAxSTB (DMA Channel x RAM Start	
Register Map	50	Address Offset B)	. 137
_		DSADR (Most Recent DMA RAM Address)	
R		I2CxCON (I2Cx Control)	
Reader Response	370	I2CxMSK (I2Cx Slave Mode Address Mask)	
Registers		I2CxSTAT (I2Cx Status)	
ADxCHS0 (ADCx Input Channel 0 Select)	256	ICxCON (Input Capture x Control)	
ADxCHS123 (ADCx Input		IEC0 (Interrupt Enable Control 0)	
Channel 1, 2, 3 Select)	255	IEC1 (Interrupt Enable Control 1)	
ADxCON1 (ADCx Control 1)		IEC2 (Interrupt Enable Control 2)	
ADxCON2 (ADCx Control 2)			
ADxCON3 (ADCx Control 3)		IEC3 (Interrupt Enable Control 3)	
ADxCON4 (ADCx Control 4)		IEC4 (Interrupt Enable Control 4)	
ADxCSSH (ADCx Input Scan Select High)		IFS0 (Interrupt Flag Status 0)	
		IFS1 (Interrupt Flag Status 1)	
ADXCSSL (ADCx Input Scan Select Low)		IFS2 (Interrupt Flag Status 2)	
ADXPCFGH (ADCx Port Configuration High)		IFS3 (Interrupt Flag Status 3)	
ADxPCFGL (ADCx Port Configuration Low)		IFS4 (Interrupt Flag Status 4)	
CiBUFPNT1 (ECAN Filter 0-3 Buffer Pointer)		INTCON1 (Interrupt Control 1)	
CiBUFPNT2 (ECAN Filter 4-7 Buffer Pointer)		INTCON2 (Interrupt Control 2)	
CiBUFPNT3 (ECAN Filter 8-11 Buffer Pointer)		INTTREG (Interrupt Control and Status)	. 130
CiBUFPNT4 (ECAN Filter 12-15 Buffer Pointer)		IPC0 (Interrupt Priority Control 0)	
CiCFG1 (ECAN Baud Rate Configuration 1)	228	IPC1 (Interrupt Priority Control 1)	
CiCFG2 (ECAN Baud Rate Configuration 2)	229	IPC10 (Interrupt Priority Control 10)	
CiCTRL1 (ECAN Control 1)	220	IPC11 (Interrupt Priority Control 11)	
CiCTRL2 (ECAN Control 2)	221	IPC12 (Interrupt Priority Control 12)	
CiEC (ECAN Transmit/Receive Error Count)	227	, , , , , , , , , , , , , , , , , , , ,	

IPC13 (Interrupt Priority Control 13)12	25
IPC14 (Interrupt Priority Control 14)12	26
IPC15 (Interrupt Priority Control 15)12	27
IPC16 (Interrupt Priority Control 16)12	28
IPC17 (Interrupt Priority Control 17)12	
IPC2 (Interrupt Priority Control 2)1	14
IPC3 (Interrupt Priority Control 3)1	
IPC4 (Interrupt Priority Control 4)1	
IPC5 (Interrupt Priority Control 5)1	17
IPC6 (Interrupt Priority Control 6)1	18
IPC7 (Interrupt Priority Control 7)1	19
IPC8 (Interrupt Priority Control 8)12	20
IPC9 (Interrupt Priority Control 9)12	21
NVMCOM (Flash Memory Control)	75
OCxCON (Output Compare x Control)1	77
OSCCON (Oscillator Control)14	
OSCTUN (FRC Oscillator Tuning)19	50
PLLFBD (PLL Feedback Divisor)14	49
PMD1 (Peripheral Module Disable Control 1)19	55
PMD2 (Peripheral Module Disable Control 2)19	57
PMD3 (Peripheral Module Disable Control 3)19	59
PWMxCON1 (PWMx Control 1)18	
PWMxCON2 (PWMx Control 2)18	85
PxDC1 (PWMx Duty Cycle 1)19	91
PxDC2 (PWMx Duty Cycle 2)19	91
PxDC3 (PWMx Duty Cycle 3)19	92
PxDC4 (PWMx Duty Cycle 4)19	92
PxDTCON1 (PWMx Dead-Time Control 1)18	
PxDTCON2 (PWMx Dead-Time Control 2)18	
PxFLTACON (PWMx Fault A Control)18	
PxFLTBCON (PWMx Fault B Control)18	
PxOVDCON (PWMx Override Control)19	
PxSECMP (PWMx Special Event Compare)18	
PxTCON (PWMx Time Base Control)18	
PxTMR (PWMx Timer Count Value)18	
PxTPER (PWMx Time Base Period)18	
QEIXCON (QEIX Control)	
RCON (Reset Control)	
SPIxCON1 (SPIx Control 1)	
SPIxCON2 (SPIx Control 2)	02
SPIxSTAT (SPIx Status and Control)	99
SR (CPU STATUS)	
SR (CPU Status)	
T1CON (Timer1 Control)	ю
TxCON (T2CON, T4CON, T6CON or	70
T8CON (T3CON T5CON T7CON or	70
TyCON (T3CON, T5CON, T7CON or	71
T9CON Control)1  UxMODE (UARTx Mode)	
UxSTA (UARTx Status and Control)	13
Clock Source Selection	ຂາ
Special Function Register States	
Times	
Reset Sequence	
Resets	
Revision History	

S	
Serial Peripheral Interface (SPI)	197
Software Simulator (MPLAB SIM)	
Software Stack Pointer, Frame Pointer	
CALL Stack Frame	63
Special Features of the CPU	259
SPI Module	
SPI1 Register Map	
SPI2 Register Map	
Symbols Used in Opcode Descriptions	268
System Control	
Register Map	62
T	
Temperature and Voltage Specifications	
AC290,	333
Timer1	
Timer2/3, Timer4/5, Timer6/7 and Timer8/9	
Timing Characteristics	
CLKO and I/O	293
Timing Diagrams	
10-Bit A/D Conversion (CHPS<1:0> = 01,	
SIMSAM = 0, $ASAM = 0$ ,	
SSRC<2:0> = 000)	326
10-Bit A/D Conversion (CHPS<1:0> = 01,	
SIMSAM = 0, ASAM = 1, SSRC<2:0> = 111,	
SAMC<4:0> = 00001)	
12-Bit A/D Conversion (ASAM = 0, SSRC = 000)	
CAN I/O	
External Clock	
I2Cx Bus Data (Master Mode)	
I2Cx Bus Data (Slave Mode) I2Cx Bus Start/Stop Bits (Master Mode)	
I2Cx Bus Start/Stop Bits (Master Mode)	
Input Capture (CAPx)	
Motor Control PWM	
Motor Control PWM Fault	
OC/PWM	
Output Compare (OCx)	298
QEA/QEB Input	
QEI Module Index Pulse	302
Reset, Watchdog Timer, Oscillator Start-up Timer	
and Power-up Timer	
Timer1, 2, 3, 4, 5, 6, 7, 8, 9 External Clock	
TimerQ (QEI Module) External Clock	303
Timing Requirements	
ADC Conversion (10-bit mode)	
ADC Conversion (12-bit Mode)	
CLKO and I/O	
External Clock	
Input Capture SPIx Master Mode (CKE = 0)	
SPIx Module Master Mode (CKE = 0)	
SPIx Module Slave Mode (CKE = 1)	
SPIx Module Slave Mode (CKE = 1)	
or ix Module older Mode (ONL - 1)	555

Timing Specifications	
10-Bit A/D Conversion Requirements	. 328
12-Bit A/D Conversion Requirements	. 325
CAN I/O Requirements	. 320
I2Cx Bus Data Requirements (Master Mode)	. 317
I2Cx Bus Data Requirements (Slave Mode)	. 319
Motor Control PWM Requirements	. 300
Output Compare Requirements	. 298
PLL Clock292	, 333
QEI External Clock Requirements	. 303
QEI Index Pulse Requirements	. 302
Quadrature Decoder Requirements	. 301
Reset, Watchdog Timer, Oscillator Start-up Timer,	
Power-up Timer and Brown-out	
Reset Requirements	. 295
Simple OC/PWM Mode Requirements	. 299
Timer1 External Clock Requirements	. 296
Timer2, Timer4, Timer6 and Timer8 External	
Clock Requirements	. 297
Timer3, Timer5, Timer7 and Timer9	
External Clock Requirements	297

U	
UART Module	
UART1 Register Map	51
UART2 Register Map	51
V	
Voltage Regulator (On-Chip)	264
W	
Watchdog Timer (WDT)	. 259, 265
Programming Considerations	265
WWW Address	369
WWW, On-Line Support	11

dsPIC3	dsPiC33FJXXXWCXU6A/XU8A/X1UA				
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 (FAQs), 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
- · Development Systems Information Line

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

#### READER RESPONSE

It is our intention to provide you with the best documentation possible to ensure successful use of your Microchip product. If you wish to provide your comments on organization, clarity, subject matter, and ways in which our documentation can better serve you, please FAX your comments to the Technical Publications Manager at (480) 792-4150.

Please list the following information, and use this outline to provide us with your comments about this document.

TO: RE:	Technical Publications Manager Reader Response	Total Pages Sent
	n: Name	
	Company	
	Address	_
	City / State / ZIP / Country	
	Telephone: ()	FAX: ()
Appl	lication (optional):	
Wou	ıld you like a reply? Y N	
Devi	ce: dsPIC33FJXXXMCX06A/X08A/X10A	Literature Number: DS70594D
Que	stions:	
1. \	What are the best features of this document?	
2. I	How does this document meet your hardware and s	oftware development needs?
-		
-		
3. I	Do you find the organization of this document easy	to follow? If not, why?
-		
4. \	What additions to the document do you think would	enhance the structure and subject?
-		
5.	What deletions from the document could be made w	vithout affecting the overall usefulness?
-		
6. I	Is there any incorrect or misleading information (what	at and where)?
7	Llow would you improve this document?	
7.	How would you improve this document?	

### PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.

Temperature Rang	nily ————————————————————————————————————		Examples:  a) dsPIC33FJ256MC Motor Control dsPI 64-Kbyte program 64-pin, Industrial te TQFP package.
Architecture:	33 =	16-bit Digital Signal Controller	
Flash Memory Family:	FJ =	Flash program memory, 3.3V	
Product Group:	MC5 = MC7 =	Motor Control family Motor Control family	
Pin Count:	06 = 08 = 10 =	80-pin	
Temperature Range:		-40°C to +85°C (Industrial) -40°C to +125°C (Extended) -40°C to +150°C (High)	
Package:	PT = PF = MR =	14x14 mm TQFP (Thin Quad Flatpack)	
Pattern	Three-dig (blank oth	it QTP, SQTP, Code or Special Requirements erwise)	

710ATI/PT: IC33, memory, emperature,

dsPiC33FJXXXIVICXU6A/XU8A/X1UA							
			JXXXIVIC XU6A/XU8A	JXXXIVICXU6A/XU8A/X1UA	JXXXIVICXU6A/XU8A/X1UA		

#### 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, KEELOQ, KEELOQ logo, MPLAB, PIC, PICmicro, PICSTART, PIC<sup>32</sup> logo, rfPIC and UNI/O are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

FilterLab, Hampshire, HI-TECH C, Linear Active Thermistor, MXDEV, MXLAB, SEEVAL and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Analog-for-the-Digital Age, Application Maestro, chipKIT, chipKIT logo, CodeGuard, dsPICDEM, dsPICDEM.net, dsPICworks, dsSPEAK, ECAN, ECONOMONITOR, FanSense, HI-TIDE, In-Circuit Serial Programming, ICSP, Mindi, MiWi, MPASM, MPLAB Certified logo, MPLIB, MPLINK, mTouch, Omniscient Code Generation, PICC, PICC-18, PICDEM, PICDEM.net, PICkit, PICtail, REAL ICE, rfLAB, Select Mode, Total Endurance, TSHARC, UniWinDriver, WiperLock and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

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

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

© 2009-2012, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

Printed on recycled paper.

ISBN: 978-1-62076-343-8

# 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

**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

Farmington Hills, MI Tel: 248-538-2250 Fax: 248-538-2260

Indianapolis Noblesville, IN Tel: 317-773-8323 Fax: 317-773-5453

Los Angeles

Mission Viejo, CA Tel: 949-462-9523 Fax: 949-462-9608

Santa Clara

Santa Clara, CA Tel: 408-961-6444 Fax: 408-961-6445

Toronto

Mississauga, Ontario,

Canada

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: 852-2401-1200 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 - Hangzhou** Tel: 86-571-2819-3187 Fax: 86-571-2819-3189

China - Hong Kong SAR Tel: 852-2401-1200 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 Fax: 86-21-5407-5066

**China - Shenyang** Tel: 86-24-2334-2829 Fax: 86-24-2334-2393

**China - Shenzhen** Tel: 86-755-8203-2660 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

**China - Xiamen**Tel: 86-592-2388138
Fax: 86-592-2388130 **China - Zhuhai** 

Tel: 86-756-3210040 Fax: 86-756-3210049

#### ASIA/PACIFIC

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: 91-20-2566-1512 Fax: 91-20-2566-1513

**Japan - Osaka** Tel: 81-66-152-7160 Fax: 81-66-152-9310

**Japan - Yokohama** Tel: 81-45-471- 6166 Fax: 81-45-471-6122

Tel: 82-53-744-4301 Fax: 82-53-744-4302 **Korea - Seoul** 

Korea - Daegu

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

**Taiwan - Kaohsiung** Tel: 886-7-536-4818 Fax: 886-7-330-9305

Fax: 886-3-5770-955

**Taiwan - Taipei** Tel: 886-2-2500-6610 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 - Munich** Tel: 49-89-627-144-0 Fax: 49-89-627-144-44

Italy - Milan Tel: 39-0331-742611 Fax: 39-0331-466781

**Netherlands - Drunen** Tel: 31-416-690399 Fax: 31-416-690340

**Spain - Madrid** Tel: 34-91-708-08-90 Fax: 34-91-708-08-91

**UK - Wokingham** Tel: 44-118-921-5869 Fax: 44-118-921-5820

11/29/11



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