



# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## 16-bit Digital Signal Controllers (up to 64 KB Flash and 9 KB SRAM) with High-Speed PWM, ADC, and Comparators

### Operating Conditions

- 3.0V to 3.6V, -40°C to +85°C, DC to 50 MIPS
- 3.0V to 3.6V, -40°C to +125°C, DC to 40 MIPS

### Core: 16-bit dsPIC33F

- 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
- 32-bit multiply support

### Clock Management

- ±1% 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
- 1.7 mA/MHz dynamic current (typical)
- 50 µA IPD current (typical)

### High-Speed PWM

- Up to nine PWM pairs with independent timing
- Dead time for rising and falling edges
- 1.04 ns PWM resolution
- PWM support for:
  - DC/DC, AC/DC, Inverters, PFC, Lighting
  - BLDC, PMSM, ACIM, SRM
- Programmable Fault inputs
- Flexible trigger configurations for ADC conversions

### Advanced Analog Features

- High-Speed ADC module:
  - 10-bit resolution with up to two Successive Approximation Register (SAR) converters (up to 4 Msps)
  - Up to 24 input channels grouped into 12 conversion pairs plus two voltage reference monitoring inputs
  - Dedicated result buffer for each analog channel
- Flexible and independent ADC trigger sources

### Advanced Analog Features (Continued)

- Up to four High-Speed Comparators with direct connection to the PWM module:
  - 10-bit Digital-to-Analog Converter (DAC) for each comparator
  - DAC reference output
  - Programmable references with 1024 voltage points

### Timers/Output Compare/Input Capture

- Six general purpose timers:
  - Five 16-bit and up to two 32-bit timers/counters
- Four Output Compare (OC) modules configurable as timers/counters
- Quadrature Encoder Interface (QEI) module configurable as timer/counter
- Four Input Capture (IC) modules

### Communication Interfaces

- Two UART modules (12.5 Mbps)
  - With support for LIN 2.0 protocols and IrDA®
- Two 4-wire SPI modules (15 Mbps)
- ECAN™ module (1 Mbaud) CAN 2.0B support
- Two I<sup>2</sup>C™ modules (up to 1 Mbaud) with SMBus support

### Direct Memory Access (DMA)

- 4-channel DMA with user-selectable priority arbitration
- UART, SPI, ECAN™, IC, OC, and Timers

### Input/Output

- Sink/Source 18 mA on 18 pins, 10 mA on 1 pin, or 6 mA on 66 pins
- 5V-tolerant pins
- Selectable open drain and pull-ups
- 29 external interrupts

### Qualification and Class B Support

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

### 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

Type	QFN		TQFP		
Pin Count	64	64	80	100	100
I/O Pins	58	58	74	85	85
Contact Lead/Pitch	0.50	0.50	0.50	0.40	0.50
Dimensions	9x9x0.9	10x10x1	12x12x1	12x12x1	14x14x1

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

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 PRODUCT FAMILIES

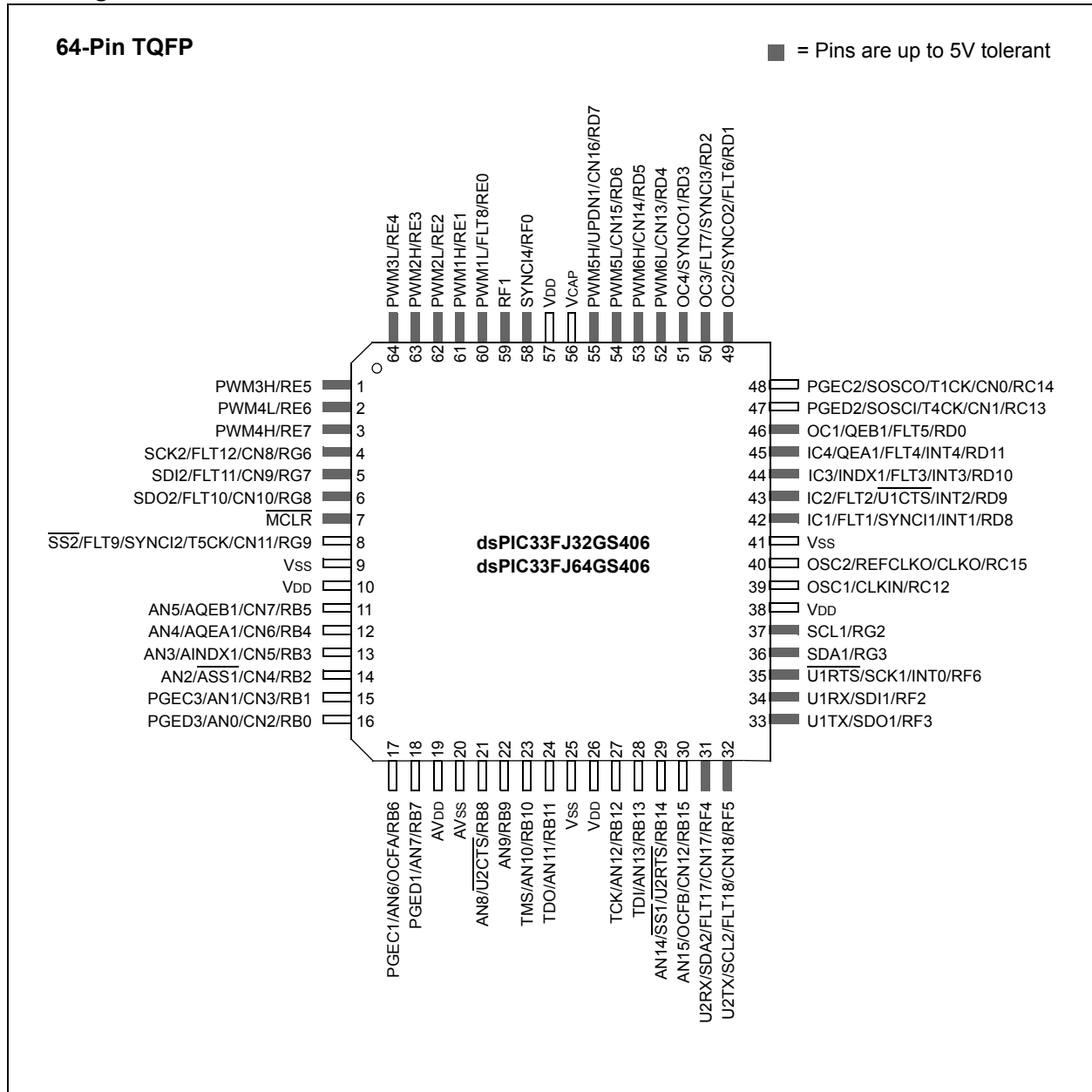
The device names, pin counts, memory sizes, and peripheral availability of each device are listed in [Table 1](#). The following pages show their pinout diagrams.

**TABLE 1: dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 CONTROLLER FAMILIES**

Device	Pins	Program Flash Memory (Kbytes)	RAM (Bytes)	16-bit Timer	Input Capture	Output Compare	UART	Quadrature Encoder Interface	SPI	ECAN™	DMA Channels	PWM	Analog Comparator	External Interrupts	DAC Output	I <sup>2</sup> C™	ADC			I/O Pins	Packages
																	SARs	Sample and Hold (S&H) Circuit	Analog-to-Digital Inputs		
dsPIC33FJ32GS406	64	32	4K	5	4	4	2	1	2	0	0	6x2	0	5	0	2	1	5	16	58	PT, MR
dsPIC33FJ32GS606	64	32	4K	5	4	4	2	2	2	0	0	6x2	4	5	1	2	2	6	16	58	PT, MR
dsPIC33FJ32GS608	80	32	4K	5	4	4	2	2	2	0	0	8x2	4	5	1	2	2	6	18	74	PT
dsPIC33FJ32GS610	100	32	4K	5	4	4	2	2	2	0	0	9x2	4	5	1	2	2	6	24	85	PT, PF
dsPIC33FJ64GS406	64	64	8K	5	4	4	2	1	2	0	0	6x2	0	5	0	2	1	5	16	58	PT, MR
dsPIC33FJ64GS606	64	64	9K <sup>(1)</sup>	5	4	4	2	2	2	1	4	6x2	4	5	1	2	2	6	16	58	PT, MR
dsPIC33FJ64GS608	80	64	9K <sup>(1)</sup>	5	4	4	2	2	2	1	4	8x2	4	5	1	2	2	6	18	74	PT
dsPIC33FJ64GS610	100	64	9K <sup>(1)</sup>	5	4	4	2	2	2	1	4	9x2	4	5	1	2	2	6	24	85	PT, PF

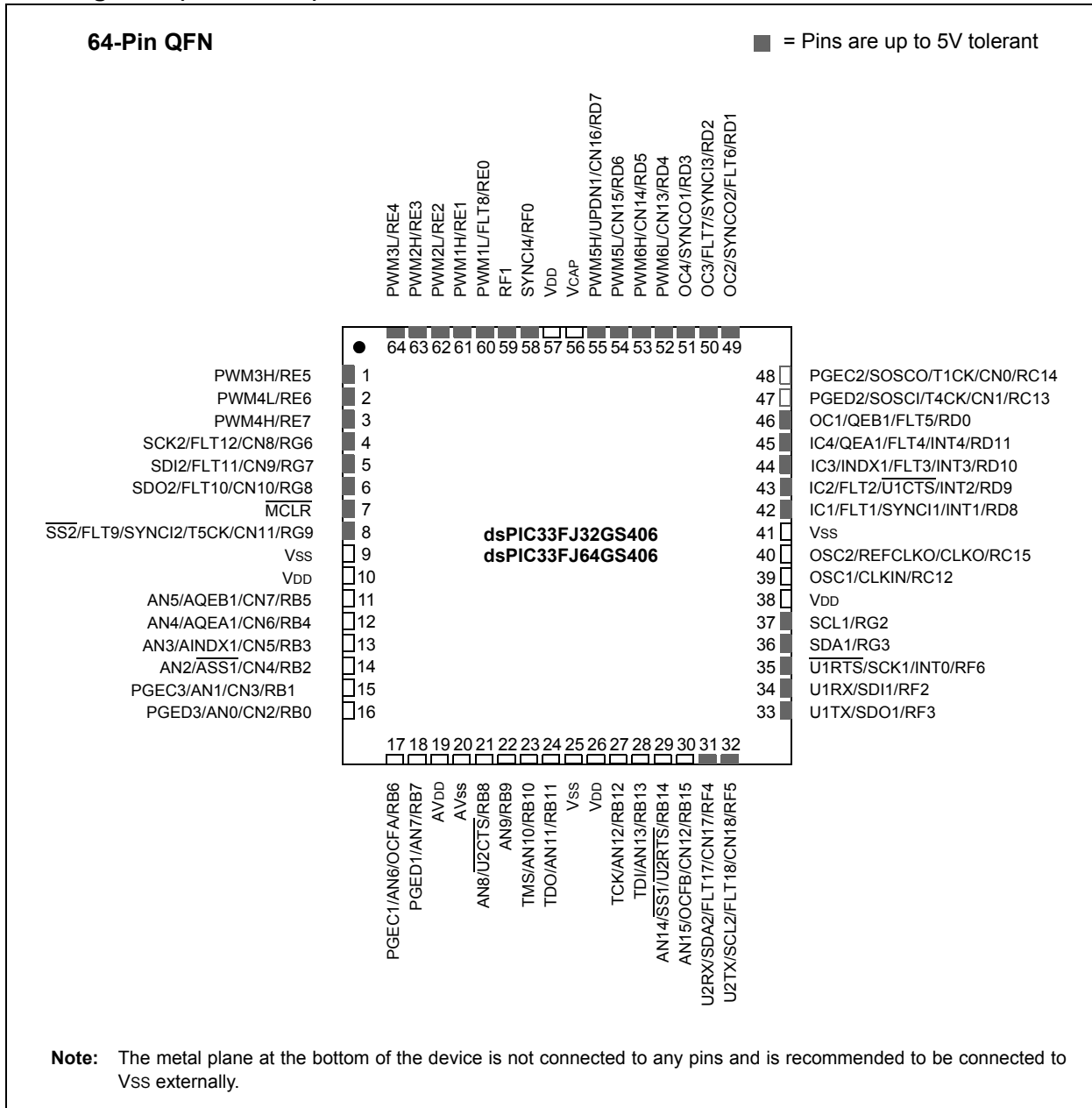
**Note 1:** RAM size is inclusive of 1 Kbyte DMA RAM.

Pin Diagrams

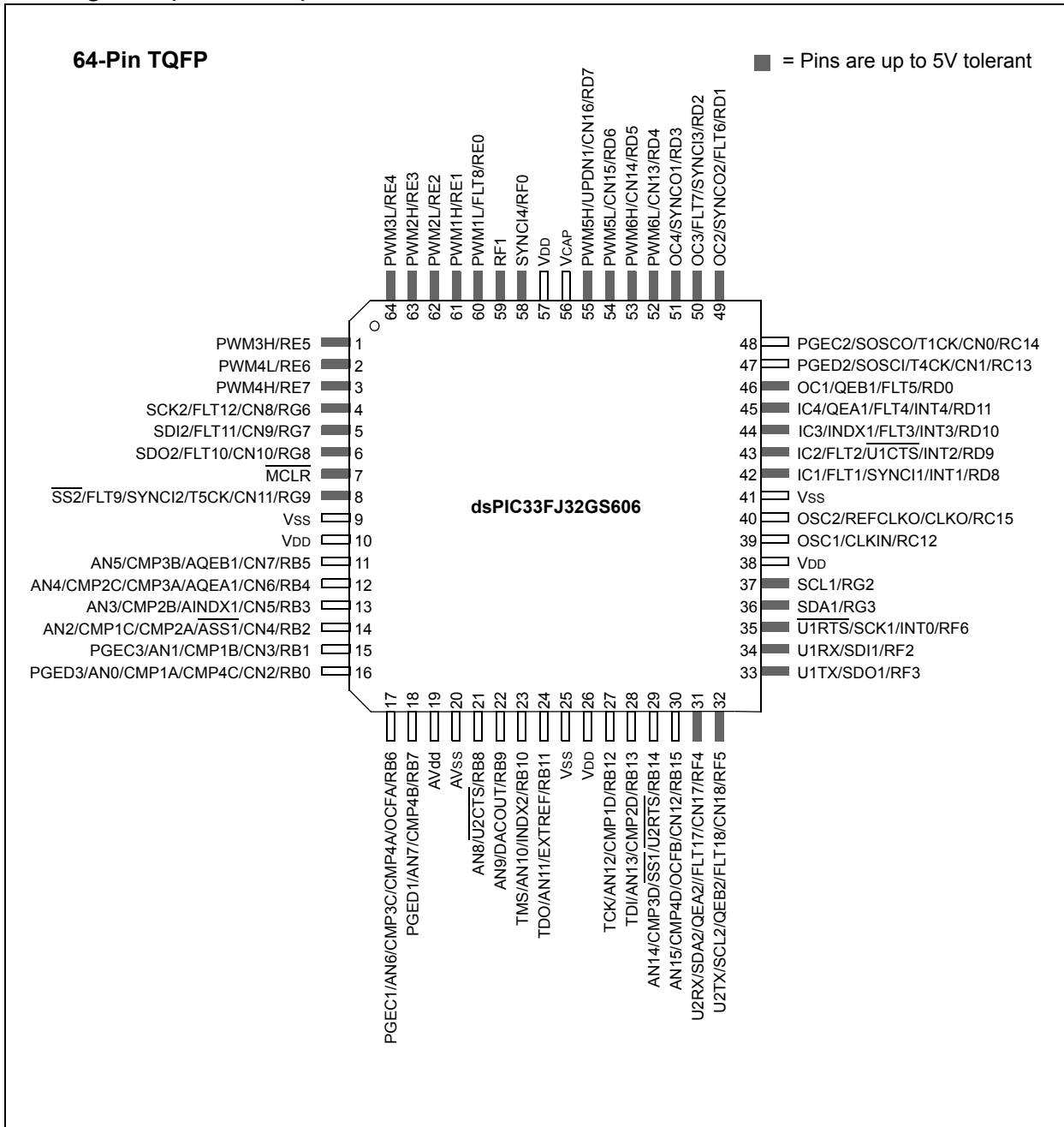


# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

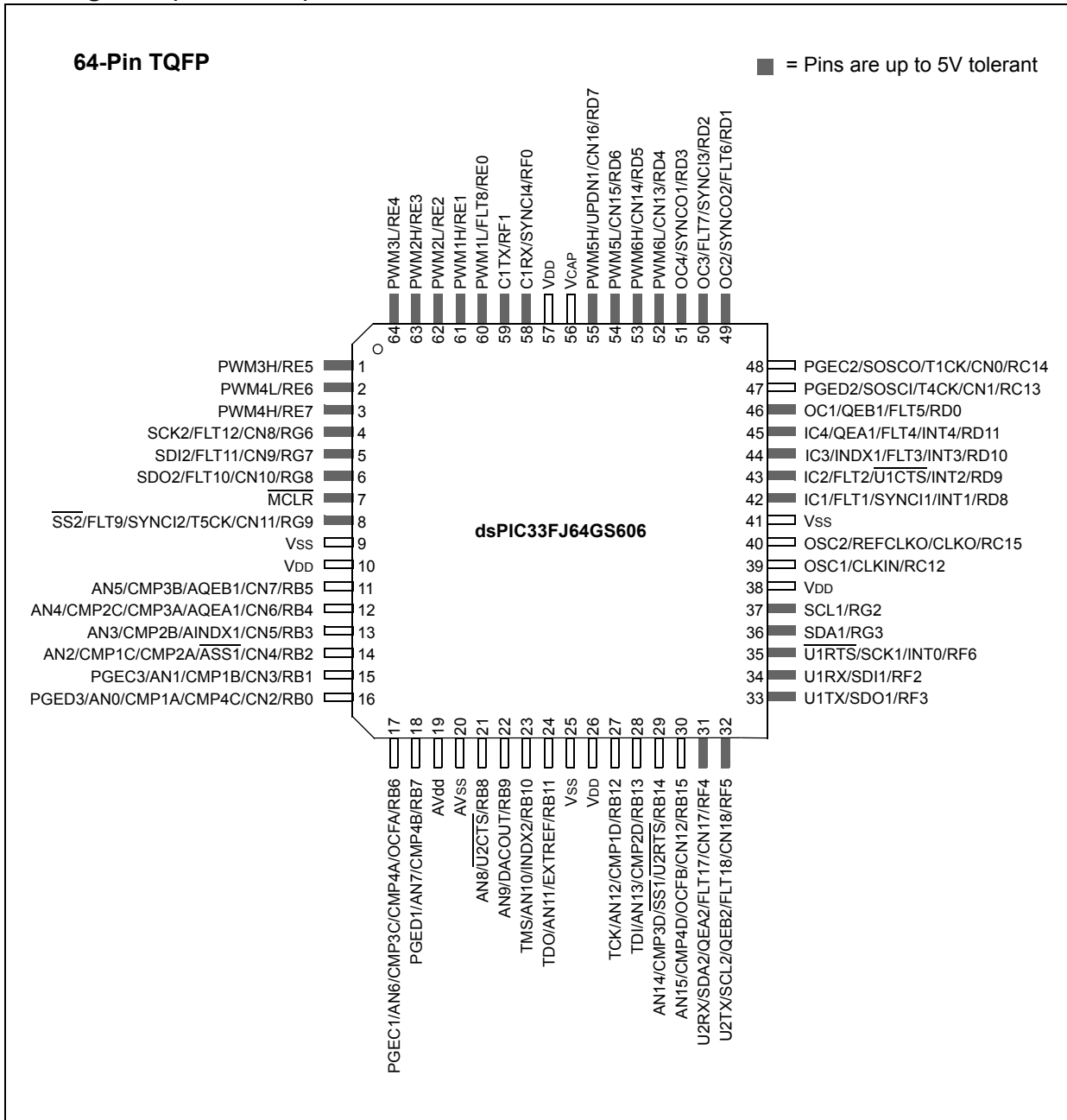
## Pin Diagrams (Continued)



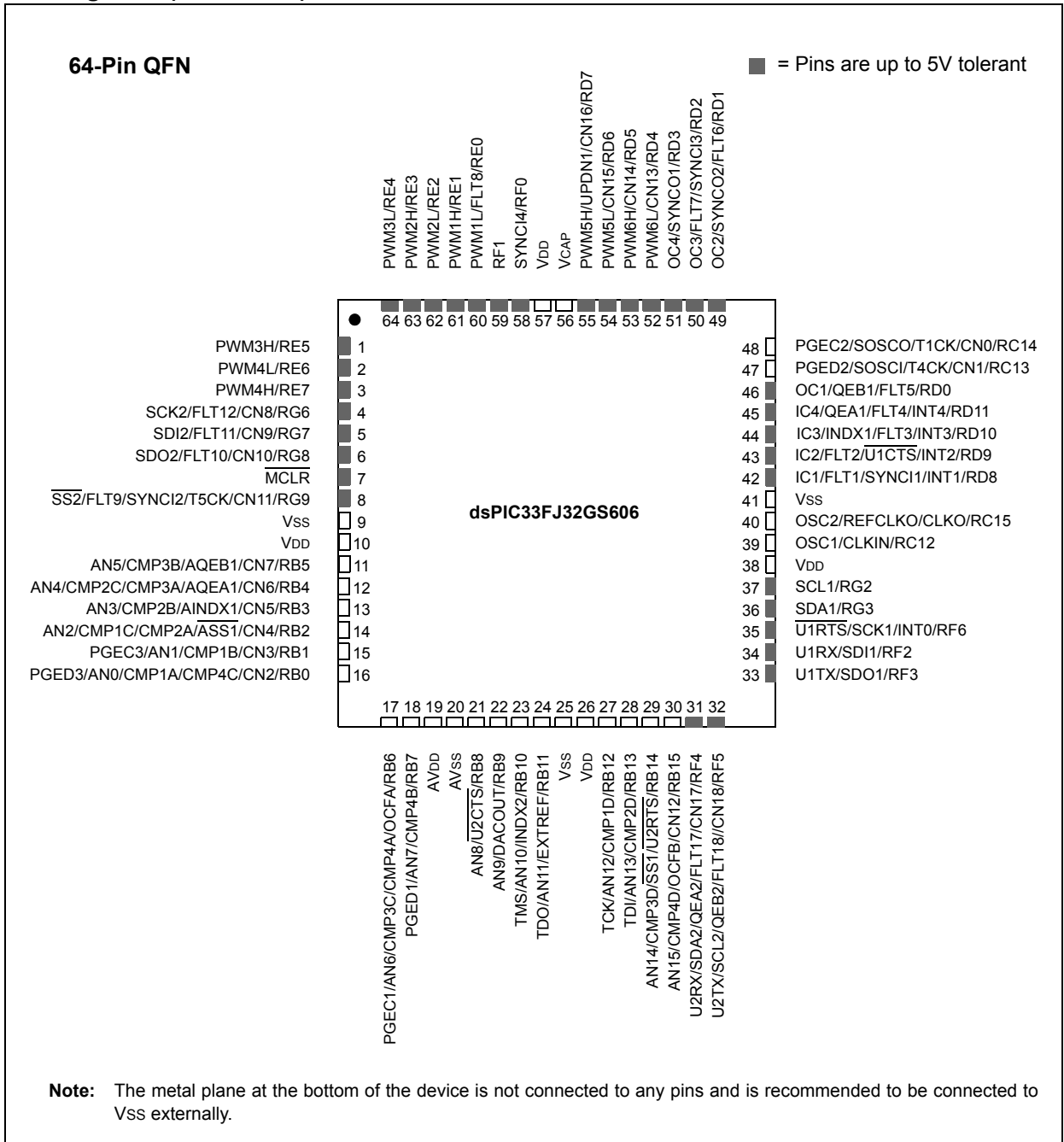
Pin Diagrams (Continued)



Pin Diagrams (Continued)

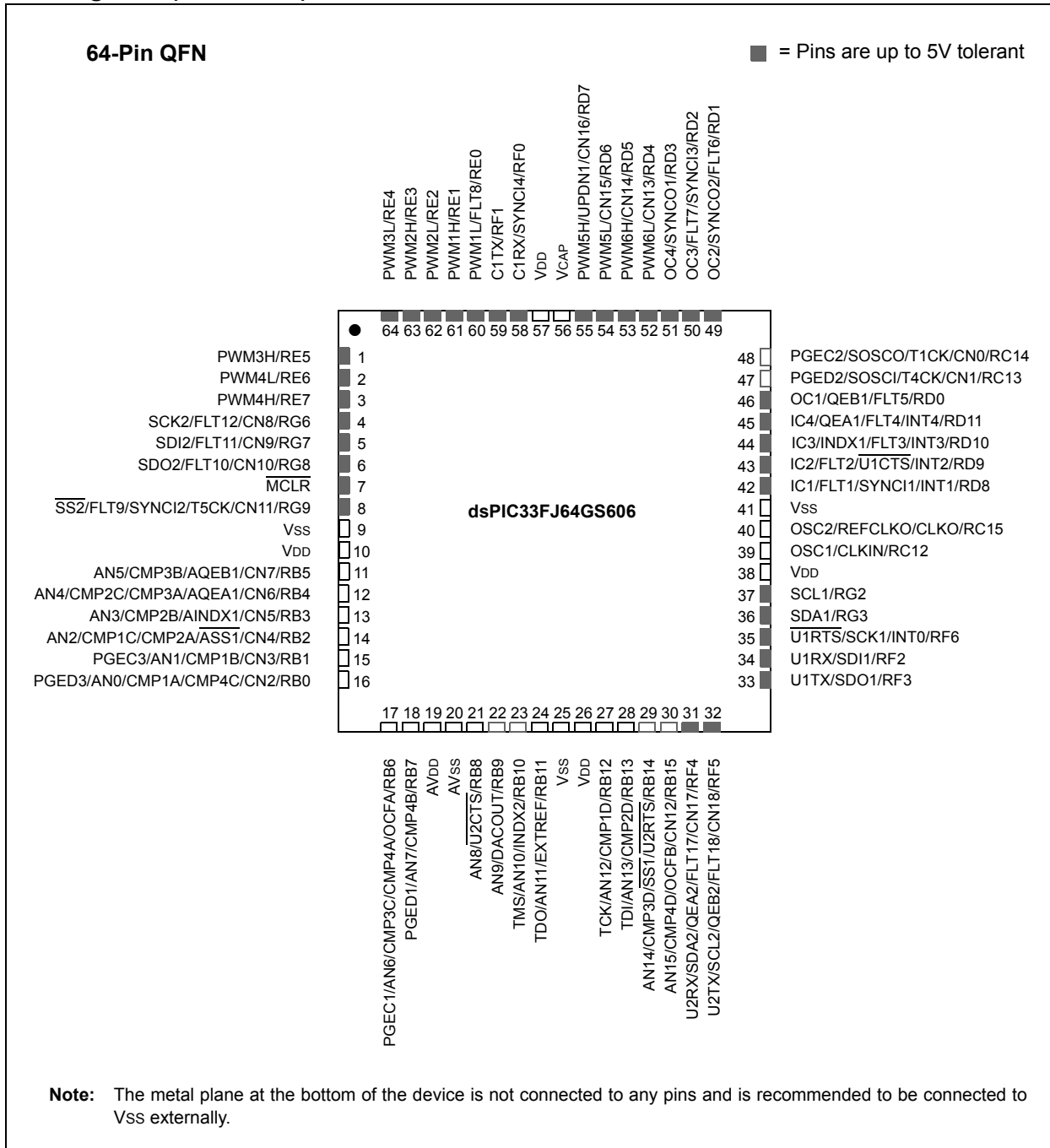


Pin Diagrams (Continued)



# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

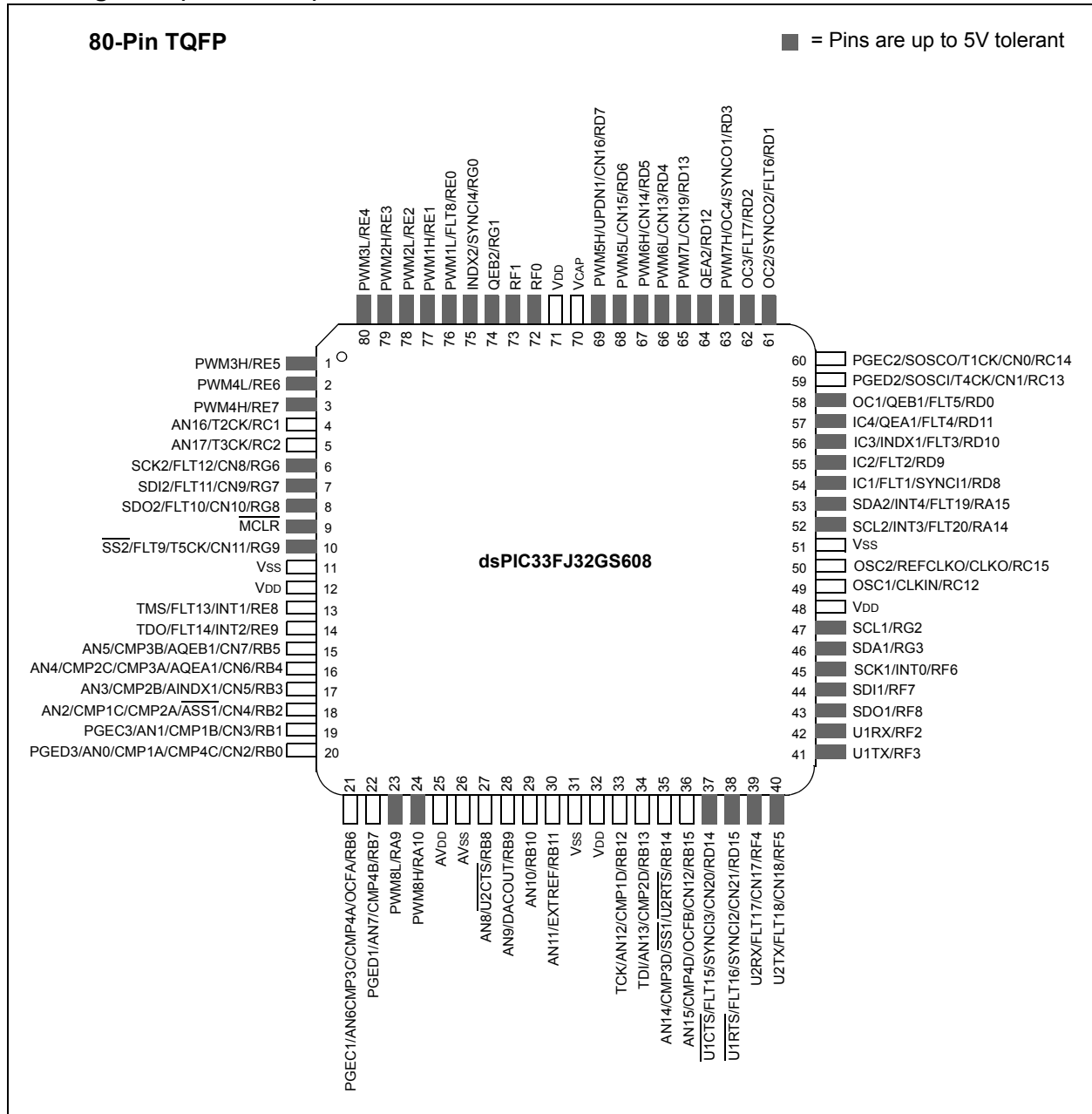
## Pin Diagrams (Continued)



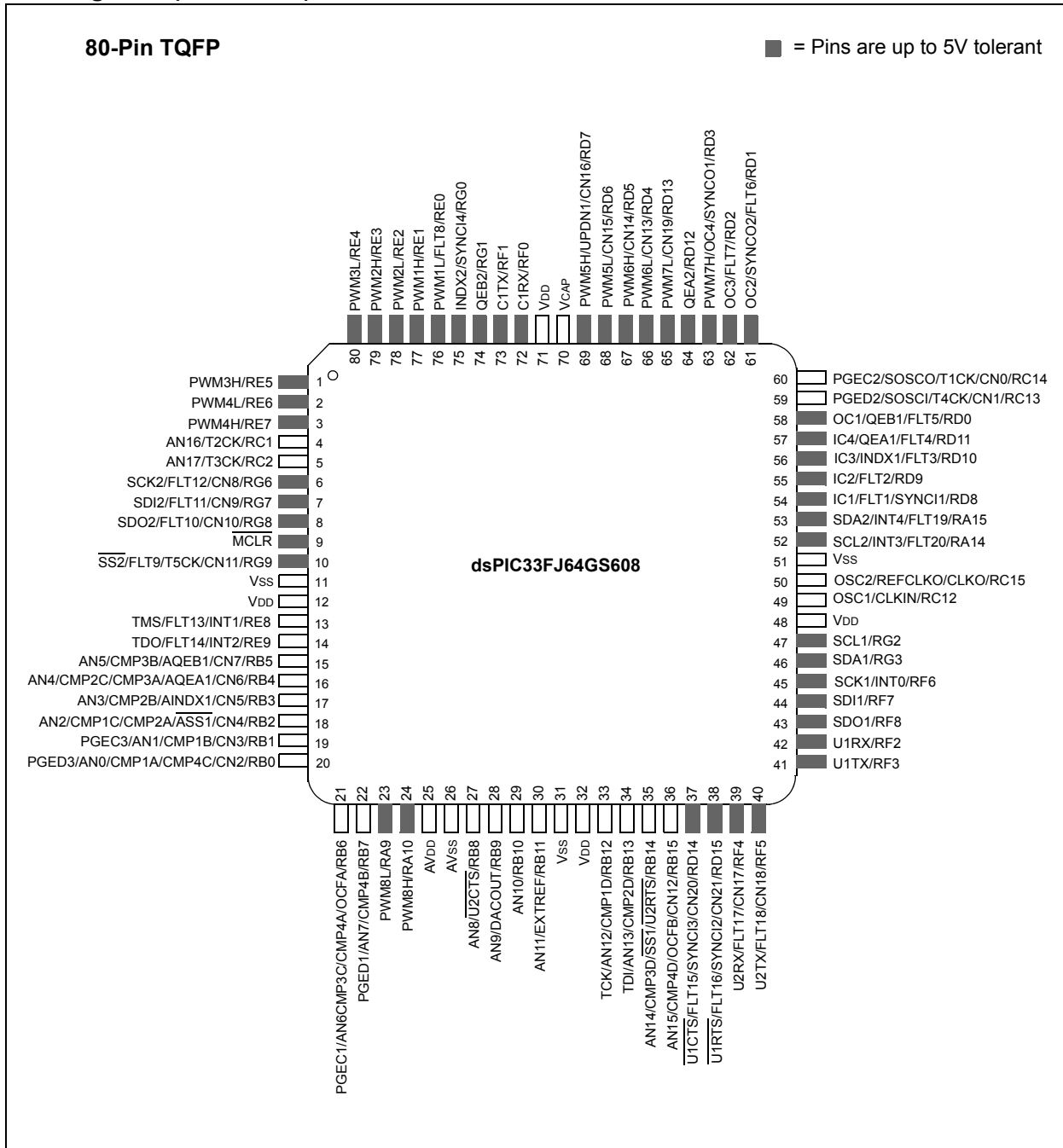
**Note:** The metal plane at the bottom of the device is not connected to any pins and is recommended to be connected to Vss externally.



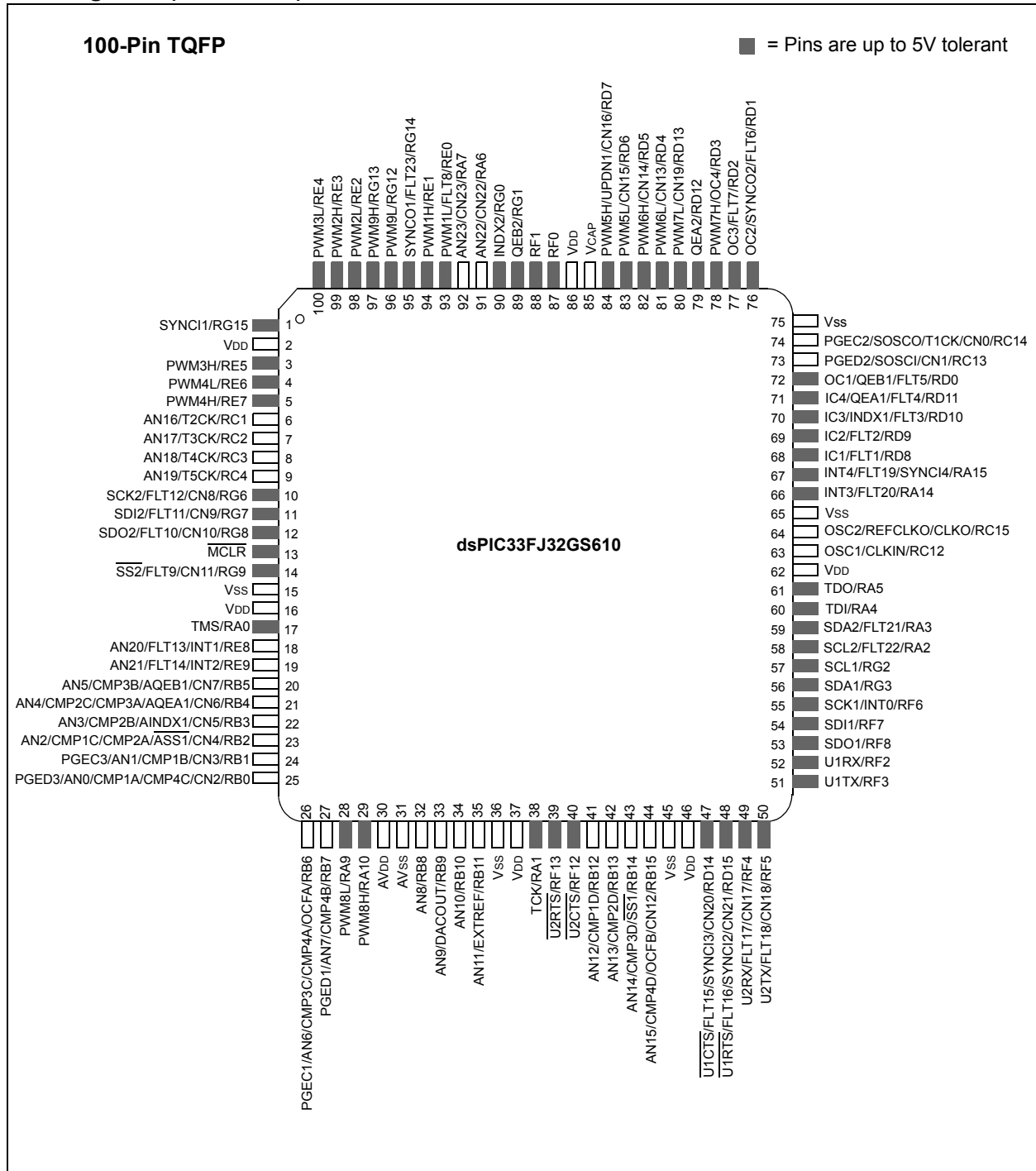
Pin Diagrams (Continued)



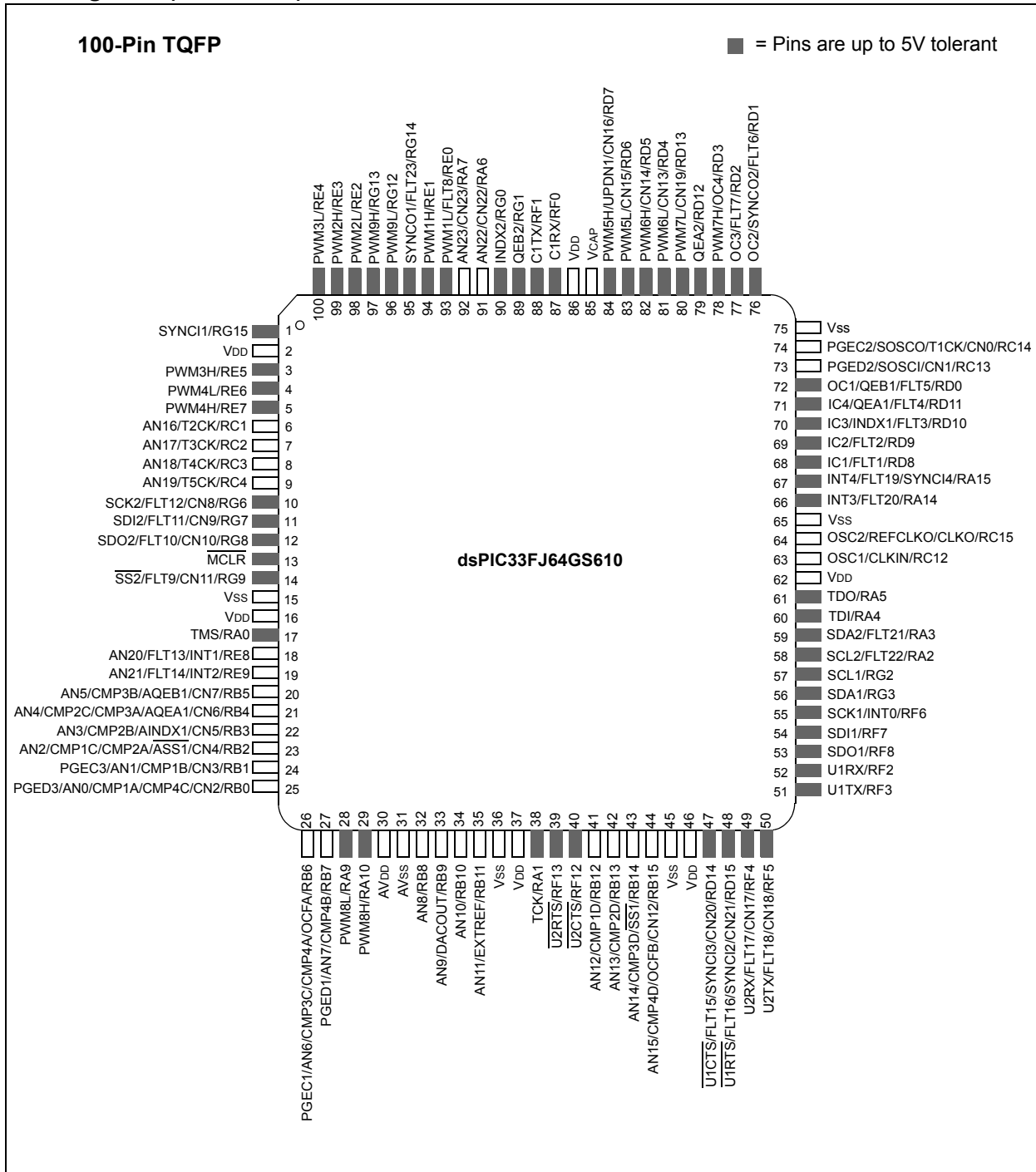
Pin Diagrams (Continued)



Pin Diagrams (Continued)



Pin Diagrams (Continued)



## Table of Contents

dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 Product Families .....	2
1.0 Device Overview .....	17
2.0 Guidelines for Getting Started with 16-bit Digital Signal Controllers .....	23
3.0 CPU .....	33
4.0 Memory Organization .....	45
5.0 Flash Program Memory .....	109
6.0 Resets .....	115
7.0 Interrupt Controller .....	123
8.0 Direct Memory Access (DMA) .....	177
9.0 Oscillator Configuration .....	187
10.0 Power-Saving Features .....	199
11.0 I/O Ports .....	209
12.0 Timer1 .....	211
13.0 Timer2/3/4/5 features .....	213
14.0 Input Capture .....	219
15.0 Output Compare .....	221
16.0 High-Speed PWM .....	225
17.0 Quadrature Encoder Interface (QE1) Module .....	255
18.0 Serial Peripheral Interface (SPI) .....	259
19.0 Inter-Integrated Circuit (I <sup>2</sup> C™) .....	265
20.0 Universal Asynchronous Receiver Transmitter (UART) .....	273
21.0 Enhanced CAN (ECAN™) Module .....	279
22.0 High-Speed 10-bit Analog-to-Digital Converter (ADC) .....	305
23.0 High-Speed Analog Comparator .....	337
24.0 Special Features .....	341
25.0 Instruction Set Summary .....	349
26.0 Development Support .....	357
27.0 Electrical Characteristics .....	361
28.0 50 MIPS Electrical Characteristics .....	409
29.0 DC and AC Device Characteristics Graphs .....	417
30.0 Packaging Information .....	419
Appendix A: Migrating from dsPIC33FJ06GS101/X02 and dsPIC33FJ16GSX02/X04 to dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 Devices .....	433
Appendix B: Revision History .....	434
The Microchip Web Site .....	447
Customer Change Notification Service .....	447
Customer Support .....	447
Reader Response .....	448
Product Identification System .....	449

## 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](mailto: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](http://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 primary reference for the operation of a particular module or device feature.

**Note 1:** To access the documents listed below, browse to the documentation section of the [dsPIC33FJ64GS610](#) product page of the Microchip web site ([www.microchip.com](http://www.microchip.com)) to 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 2. “CPU”** (DS70204)
- **Section 3. “Data Memory”** (DS70202)
- **Section 4. “Program Memory”** (DS70203)
- **Section 5. “Flash Programming”** (DS70191)
- **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 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™ (I<sup>2</sup>C™)”** (DS70195)
- **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)
- **Section 26. “Development Tool Support”** (DS70200)
- **Section 42. “Oscillator (Part IV)”** (DS70307)
- **Section 43. “High-Speed PWM”** (DS70323)
- **Section 44. “High-Speed 10-Bit Analog-to-Digital Converter (ADC)”** (DS70321)
- **Section 45. “High-Speed Analog Comparator”** (DS70296)
- **Section 47. “Interrupts (Part V)”** (DS70597)

NOTES:



## 1.0 DEVICE OVERVIEW

**Note:** This data sheet summarizes the features of the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 families of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to the latest sections in the “dsPIC33F/PIC24H Family Reference Manual”, which are available from the Microchip web site ([www.microchip.com](http://www.microchip.com)).

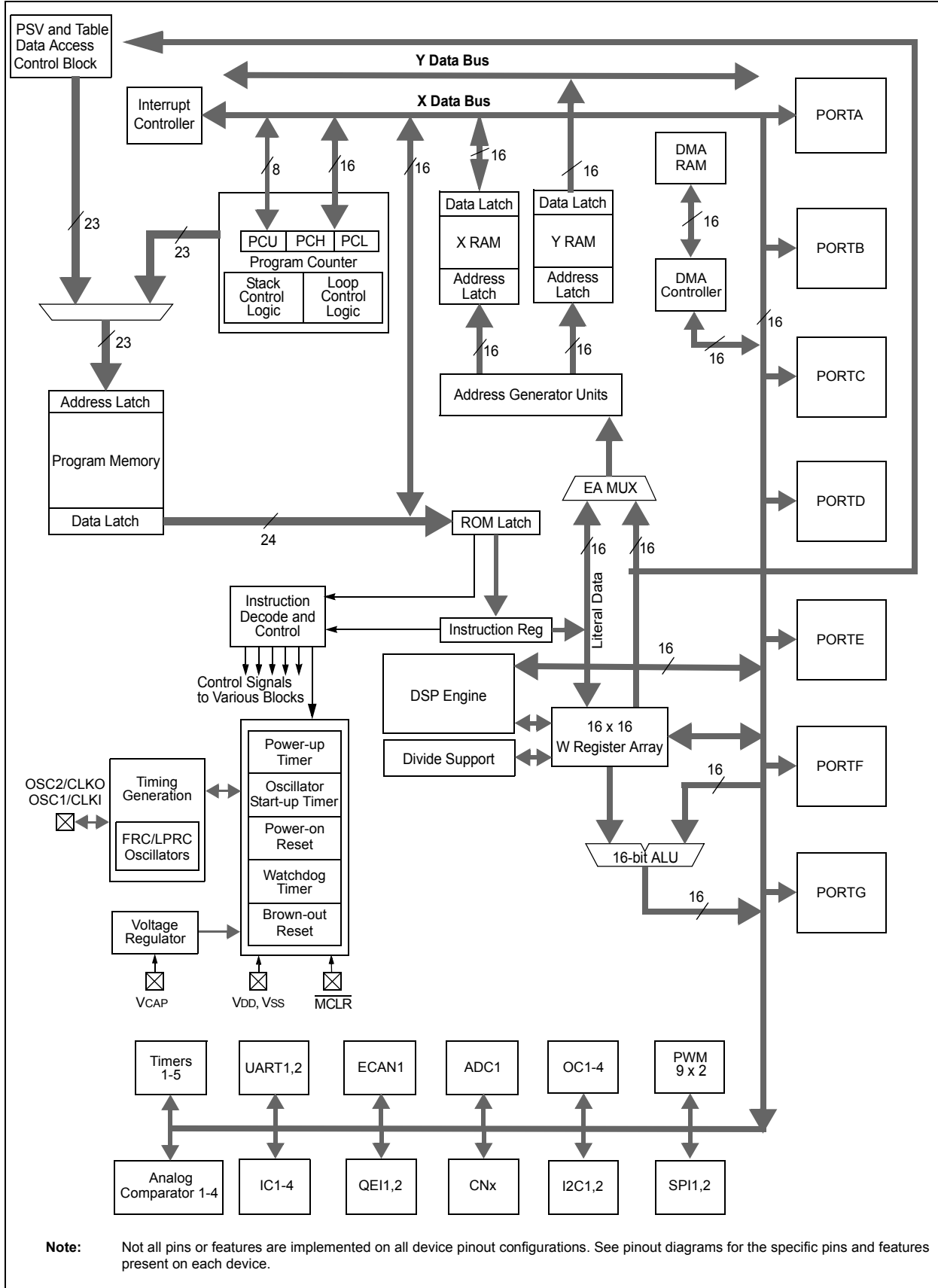
This document contains device-specific information for the following dsPIC33F Digital Signal Controller (DSC) devices:

- dsPIC33FJ32GS406
- dsPIC33FJ32GS606
- dsPIC33FJ32GS608
- dsPIC33FJ32GS610
- dsPIC33FJ64GS406
- dsPIC33FJ64GS606
- dsPIC33FJ64GS608
- dsPIC33FJ64GS610

The dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 families of devices contain extensive Digital Signal Processor (DSP) functionality with a high-performance 16-bit microcontroller (MCU) architecture.

Figure 1-1 shows a general block diagram of the core and peripheral modules in the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 devices. Table 1-1 lists the functions of the various pins shown in the pinout diagrams.

FIGURE 1-1: DEVICE BLOCK DIAGRAM



**TABLE 1-1: PINOUT I/O DESCRIPTIONS**

Pin Name	Pin Type	Buffer Type	Description
AN0-AN23	I	Analog	Analog input channels.
CLKI CLKO	I O	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.
OSC1 OSC2	I I/O	ST/CMOS —	Oscillator crystal input. ST buffer when configured in RC mode; CMOS otherwise. Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode. Optionally functions as CLKO in RC and EC modes.
SOSCI SOSCO	I O	ST/CMOS —	32.768 kHz low-power oscillator crystal input; CMOS otherwise. 32.768 kHz low-power oscillator crystal output.
CN0-CN23	I	ST	Change notification inputs. Can be software programmed for internal weak pull-ups on all inputs.
C1RX C1TX	I O	ST —	ECAN1 bus receive pin. ECAN1 bus transmit pin.
IC1-IC4	I	ST	Capture inputs 1 through 4.
INDX1, INDX2, AINDX1 QEA1, QEA2, AQEA1 QEB1, QEB2, AQEB1 UPDN1	I I I O	ST ST ST CMOS	Quadrature Encoder Index Pulse input. Quadrature Encoder Phase A input in QEI mode. Auxiliary Timer External Clock/Gate input in Timer mode. Quadrature Encoder Phase A input in QEI mode. Auxiliary Timer External Clock/Gate input in Timer mode. Position Up/Down Counter Direction State.
OCFA OCFB OC1-OC4	I I O	ST ST —	Compare Fault A input (for Compare Channels 1 and 2). Compare Fault B input (for Compare Channels 3 and 4). Compare Outputs 1 through 4.
INT0 INT1 INT2 INT3 INT4	I I I I I	ST ST ST ST ST	External Interrupt 0. External Interrupt 1. External Interrupt 2. External Interrupt 3. External Interrupt 4.
RA0-RA15	I/O	ST	PORTA is a bidirectional I/O port.
RB0-RB15	I/O	ST	PORTB is a bidirectional I/O port.
RC0-RC15	I/O	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-RF13	I/O	ST	PORTF is a bidirectional I/O port.
RG0-RG15	I/O	ST	PORTG is a bidirectional I/O port.
T1CK T2CK T3CK T4CK T5CK	I I I I I	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.

**Legend:** CMOS = CMOS compatible input or output      Analog = Analog input      I = Input  
 ST = Schmitt Trigger input with CMOS levels      P = Power      O = Output  
 TTL = Transistor-Transistor Logic

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

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

Pin Name	Pin Type	Buffer Type	Description
<u>U1CTS</u>	I	ST	UART1 clear to send.
U1RTS	O	—	UART1 ready to send.
U1RX	I	ST	UART1 receive.
U1TX	O	—	UART1 transmit.
<u>U2CTS</u>	I	ST	UART2 clear to send.
U2RTS	O	—	UART2 ready to send.
U2RX	I	ST	UART2 receive.
U2TX	O	—	UART2 transmit.
SCK1	I/O	ST	Synchronous serial clock input/output for SPI1.
SDI1	I	ST	SPI1 data in.
<u>SDO1</u>	O	—	SPI1 data out.
SS1, ASS1	I/O	ST	SPI1 slave synchronization or frame pulse I/O.
SCK2	I/O	ST	Synchronous serial clock input/output for SPI2.
SDI2	I	ST	SPI2 data in.
SDO2	O	—	SPI2 data out.
SS2	I/O	ST	SPI2 slave synchronization or frame pulse I/O.
SCL1	I/O	ST	Synchronous serial clock input/output for I2C1.
SDA1	I/O	ST	Synchronous serial data input/output for I2C1.
SCL2	I/O	ST	Synchronous serial clock input/output for I2C2.
SDA2	I/O	ST	Synchronous serial data input/output for I2C2.
TMS	I	TTL	JTAG Test mode select pin.
TCK	I	TTL	JTAG test clock input pin.
TDI	I	TTL	JTAG test data input pin.
TDO	O	—	JTAG test data output pin.
CMP1A	I	Analog	Comparator 1 Channel A.
CMP1B	I	Analog	Comparator 1 Channel B.
CMP1C	I	Analog	Comparator 1 Channel C.
CMP1D	I	Analog	Comparator 1 Channel D.
CMP2A	I	Analog	Comparator 2 Channel A.
CMP2B	I	Analog	Comparator 2 Channel B.
CMP2C	I	Analog	Comparator 2 Channel C.
CMP2D	I	Analog	Comparator 2 Channel D.
CMP3A	I	Analog	Comparator 3 Channel A.
CMP3B	I	Analog	Comparator 3 Channel B.
CMP3C	I	Analog	Comparator 3 Channel C.
CMP3D	I	Analog	Comparator 3 Channel D.
CMP4A	I	Analog	Comparator 4 Channel A.
CMP4B	I	Analog	Comparator 4 Channel B.
CMP4C	I	Analog	Comparator 4 Channel C.
CMP4D	I	Analog	Comparator 4 Channel D.
DACOUT	O	—	DAC output voltage
EXTREF	I	Analog	External Voltage Reference Input for the Reference DACs
REFCLK	O	—	REFCLK output signal is a postscaled derivative of the system clock

**Legend:** CMOS = CMOS compatible input or output      Analog = Analog input      I = Input  
 ST = Schmitt Trigger input with CMOS levels      P = Power      O = Output  
 TTL = Transistor-Transistor Logic

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

Pin Name	Pin Type	Buffer Type	Description
FLT1-FLT23	I	ST	Fault Inputs to PWM Module.
SYNCI1-SYNCI4	I	ST	External synchronization signal to PWM Master Time Base.
SYNCO1-SYNCO2	O	—	PWM Master Time Base for external device synchronization.
PWM1L	O	—	PWM1 Low output.
PWM1H	O	—	PWM1 High output.
PWM2L	O	—	PWM2 Low output.
PWM2H	O	—	PWM2 High output.
PWM3L	O	—	PWM3 Low output.
PWM3H	O	—	PWM3 High output.
PWM4L	O	—	PWM4 Low output.
PWM4H	O	—	PWM4 High output.
PWM5L	O	—	PWM5 Low output.
PWM5H	O	—	PWM5 High output.
PWM6L	O	—	PWM6 Low output.
PWM6H	O	—	PWM6 High output.
PWM7L	O	—	PWM7 Low output.
PWM7H	O	—	PWM7 High output.
PWM8L	O	—	PWM8 Low output.
PWM8H	O	—	PWM8 High output.
PWM9L	O	—	PWM9 Low output.
PWM9H	O	—	PWM9 High output.
PGED1	I/O	ST	Data I/O pin for programming/debugging communication Channel 1.
PGEC1	I	ST	Clock input pin for programming/debugging communication Channel 1.
PGED2	I/O	ST	Data I/O pin for programming/debugging communication Channel 2.
PGEC2	I	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	I	ST	Clock input pin for programming/debugging communication Channel 3.
MCLR	I/P	ST	Master Clear (Reset) input. This pin is an active-low Reset to the device.
AVDD	P	P	Positive supply for analog modules.
AVSS	P	P	Ground reference for analog modules.
VDD	P	—	Positive supply for peripheral logic and I/O pins.
VCAP	P	—	CPU logic filter capacitor connection.
VSS	P	—	Ground reference for logic and I/O pins.

**Legend:** CMOS = CMOS compatible input or output      Analog = Analog input      I = Input  
 ST = Schmitt Trigger input with CMOS levels      P = Power      O = Output  
 TTL = Transistor-Transistor Logic

NOTES:

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

**Note 1:** This data sheet summarizes the features of the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 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”. Please see the Microchip web site ([www.microchip.com](http://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.

### 2.1 Basic Connection Requirements

Getting started with the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 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 “Capacitor on Internal Voltage Regulator \(VCAP\)”](#))
- 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”](#))

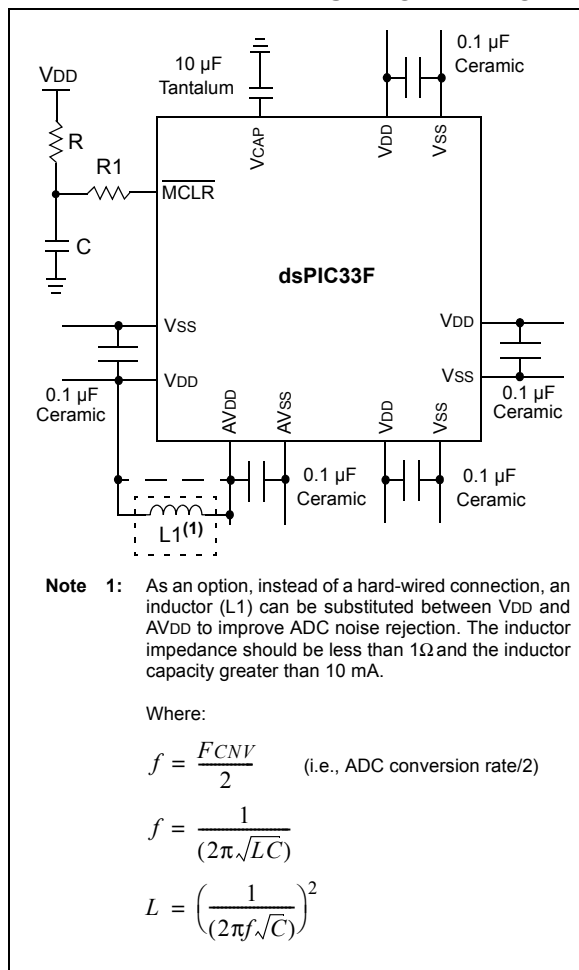
### 2.2 Decoupling Capacitors

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  $\mu\text{F}$  (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  $\mu\text{F}$  to 0.001  $\mu\text{F}$ . 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  $\mu\text{F}$  in parallel with 0.001  $\mu\text{F}$ .
- **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**



**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 µF to 47 µF.

**2.3 Capacitor on Internal Voltage Regulator (VCAP)**

A low-ESR (< 0.5 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 minimum capacitor 22 µF, 16V connected to ground. The type can be ceramic or tantalum. Refer to [Section 27.0 “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 24.2 “On-Chip Voltage Regulator”](#) for details.

**2.4 Master Clear (MCLR) Pin**

The MCLR pin provides for two specific device functions:

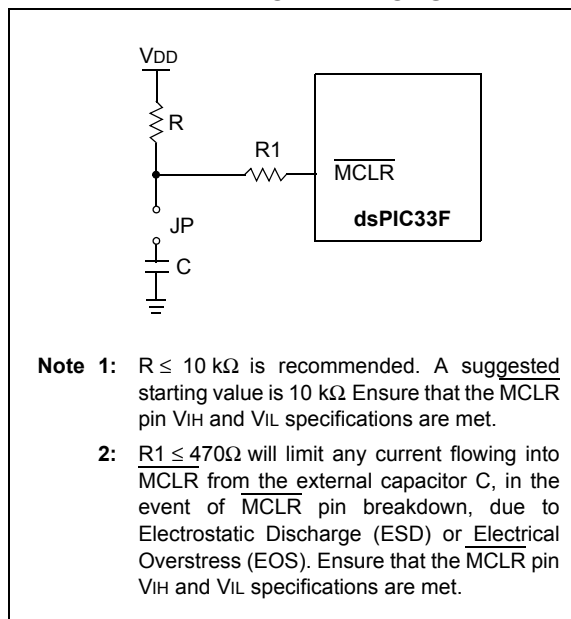
- Device Reset
- Device programming and debugging

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 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 MCLR pin during programming and debugging operations.

Place the components shown in [Figure 2-2](#) within one-quarter inch (6 mm) from the MCLR pin.

**FIGURE 2-2: EXAMPLE OF MCLR PIN CONNECTIONS<sup>(1,2)</sup>**





## 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 respective device Flash programming specification for information on capacitive loading limits and pin input voltage high (V<sub>IH</sub>) and input low (V<sub>IL</sub>) requirements.

Ensure that the “Communication Channel Select” (i.e., PGECx/PGEDx pins) programmed into the device matches the physical connections for the ICSP to MPLAB® ICD 3 or MPLAB REAL ICE™.

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

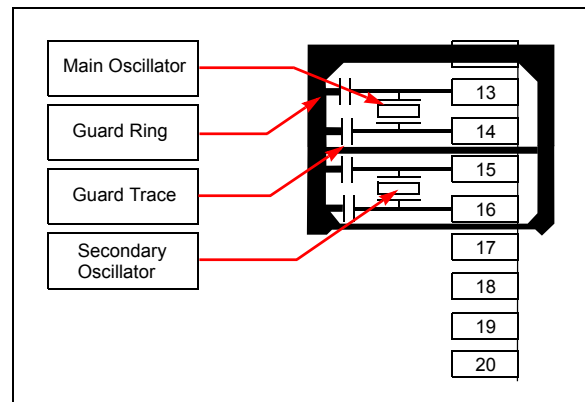
- “Using MPLAB® ICD 3” (poster) (DS51765)
- “MPLAB® ICD 3 Design Advisory” (DS51764)
- “MPLAB® REAL ICE™ In-Circuit Debugger User’s Guide” (DS51616)
- “Using MPLAB® REAL ICE™” (poster) (DS51749)

## 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  $4 \text{ MHz} < F_{IN} < 8 \text{ MHz}$  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 MPLAB ICD 3 or REAL ICE is selected as a debugger, it automatically initializes all of the analog-to-digital input pins (ANx) as “digital” pins, by setting all bits in the ADPCFG and ADPCFG2 registers.

The bits in the registers that correspond to the analog-to-digital pins that are initialized by MPLAB ICD 2, ICD 3, or REAL ICE, 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 analog-to-digital pins as analog input pins during the debug session, the user application must clear the corresponding bits in the ADPCFG and ADPCFG2 registers during initialization of the ADC module.

When MPLAB ICD 3 or REAL ICE is used as a programmer, the user application firmware must correctly configure the ADPCFG and ADPCFG2 registers. Automatic initialization of these registers is only done during debugger operation. Failure to correctly configure the register(s) will result in all analog-to-digital 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

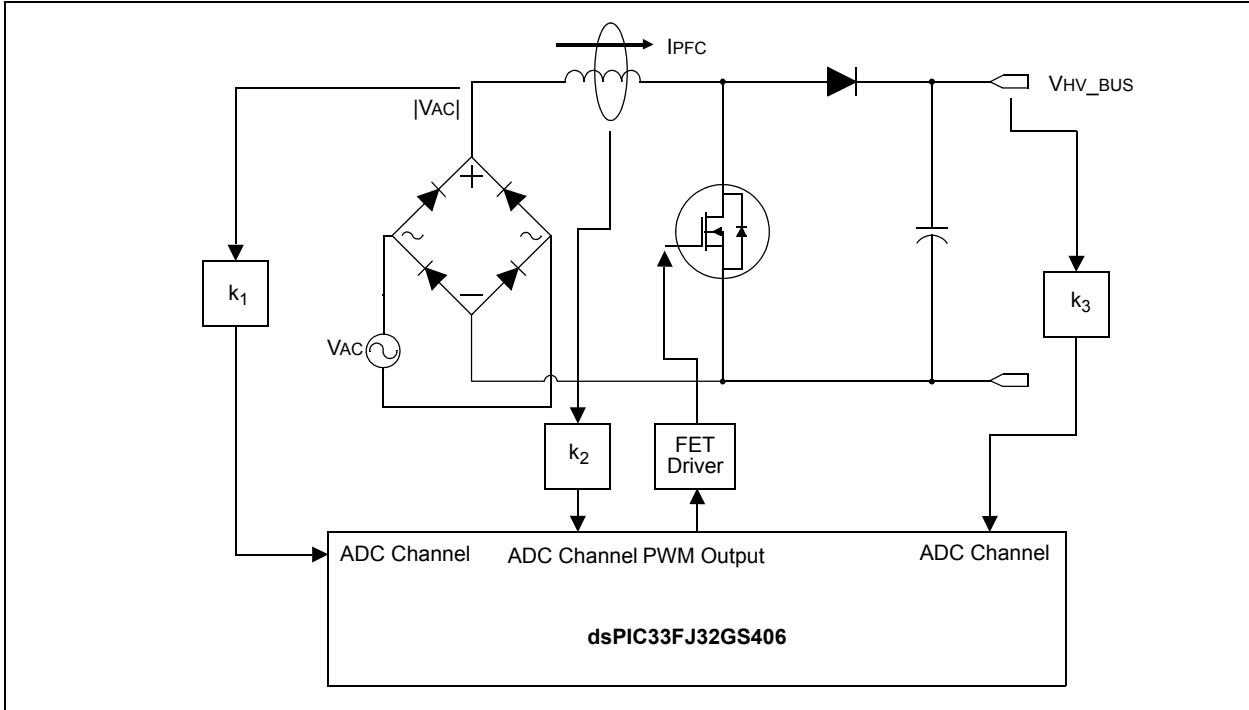
Unused I/O pins should be configured as outputs and driven to a logic-low state.

Alternatively, connect a 1k to 10k resistor between V<sub>SS</sub> and unused pins and drive the output to logic low.

## 2.10 Typical Application Connection Examples

Examples of typical application connections are shown in [Figure 2-4](#) through [Figure 2-11](#).

**FIGURE 2-4: DIGITAL PFC**



**FIGURE 2-5: BOOST CONVERTER IMPLEMENTATION**

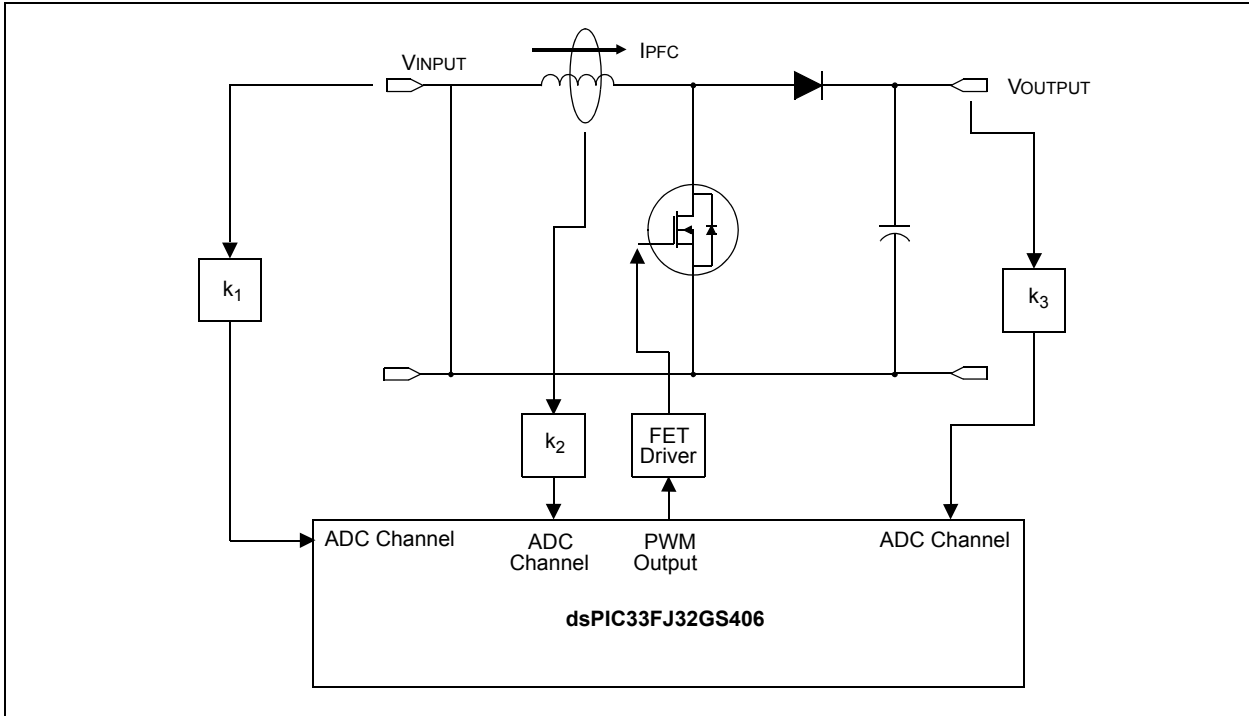


FIGURE 2-6: SINGLE-PHASE SYNCHRONOUS BUCK CONVERTER

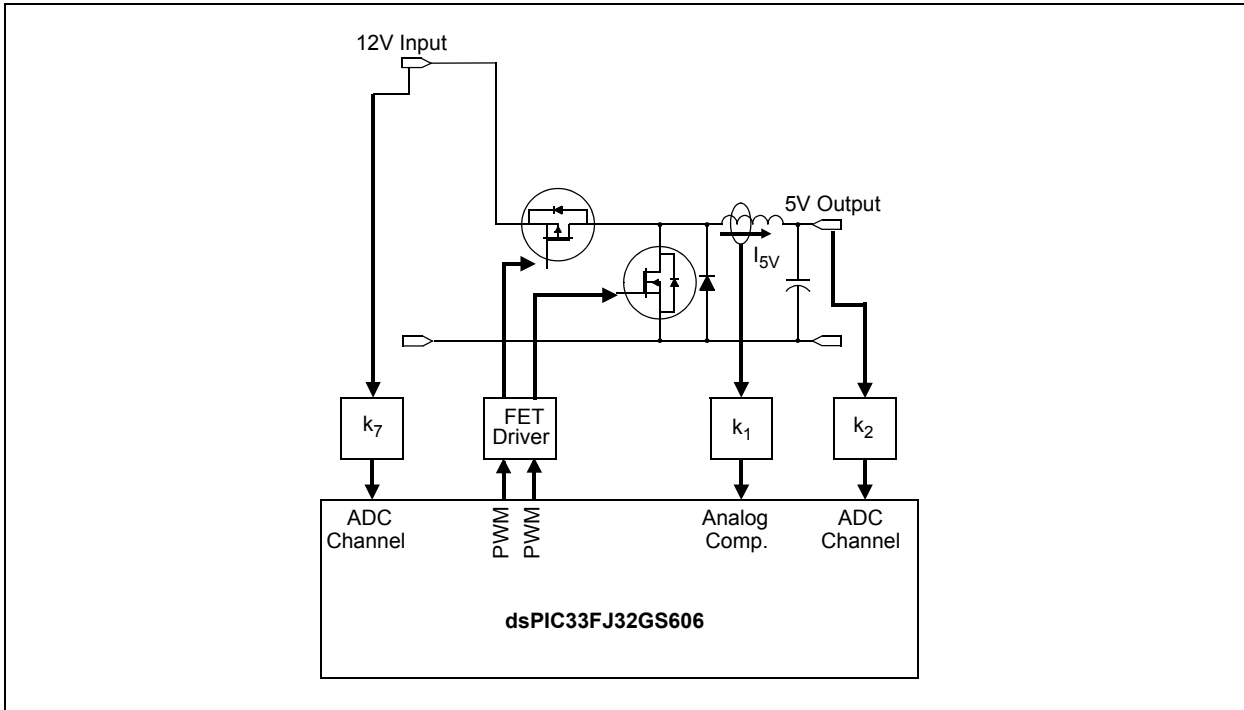


FIGURE 2-7: MULTI-PHASE SYNCHRONOUS BUCK CONVERTER

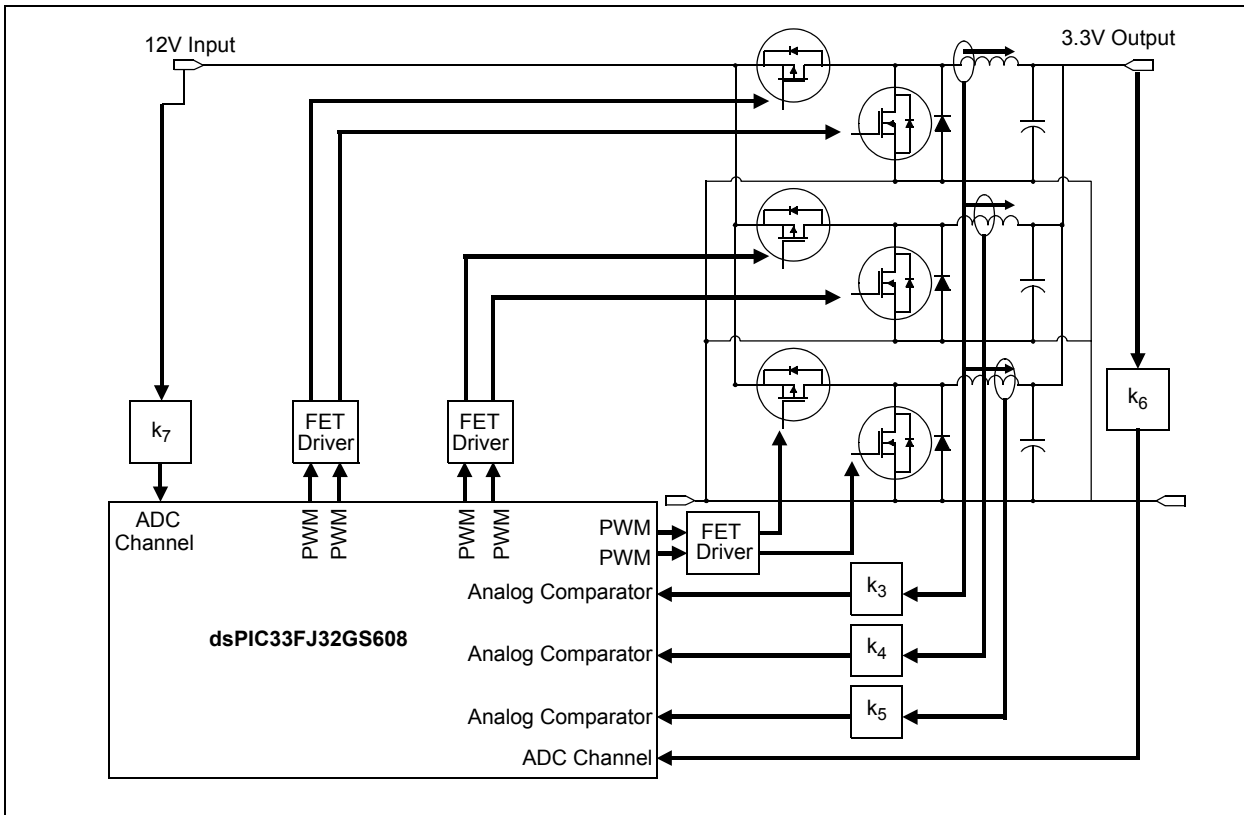
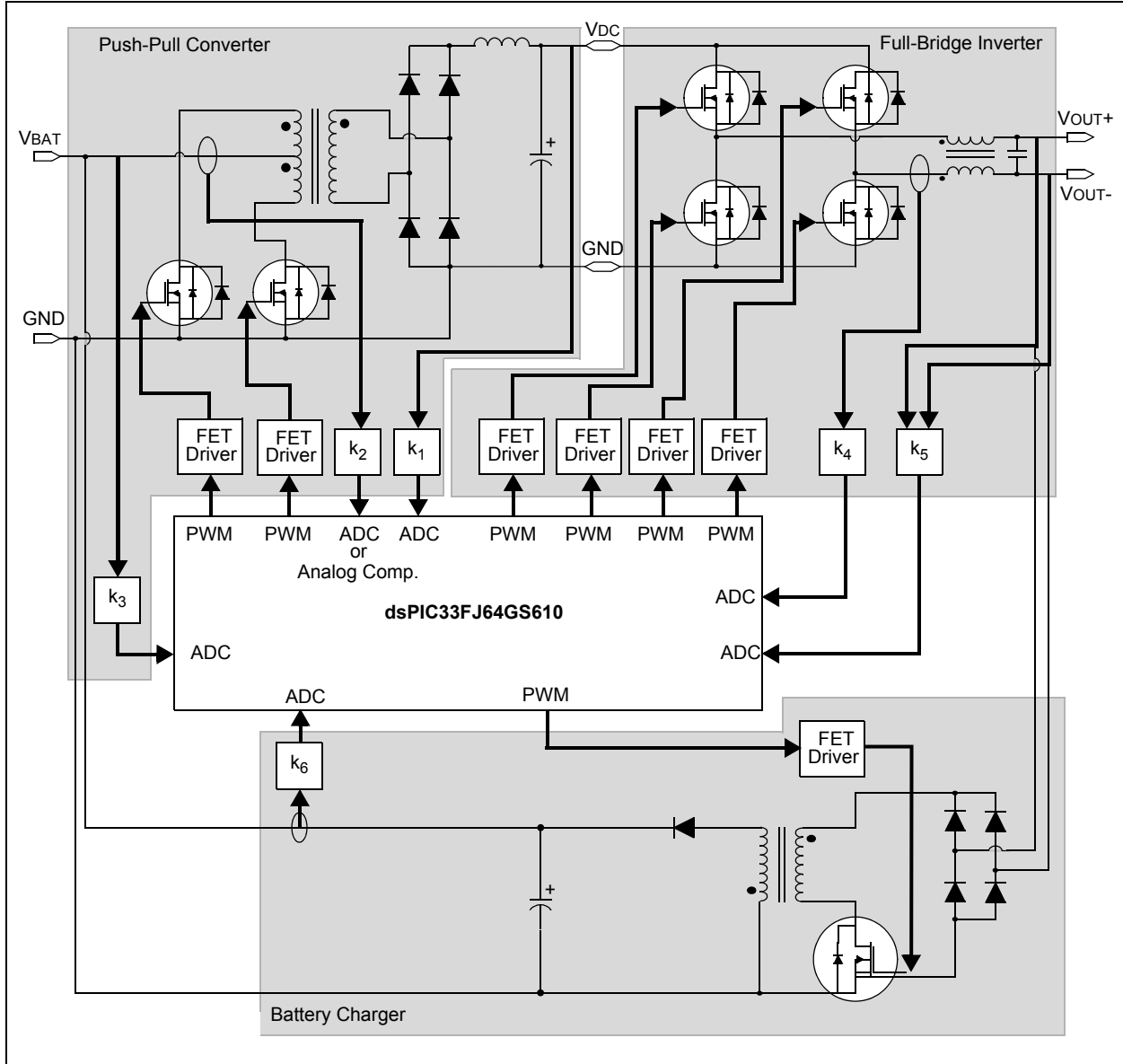


FIGURE 2-8: OFF-LINE UPS



**FIGURE 2-9: INTERLEAVED PFC**

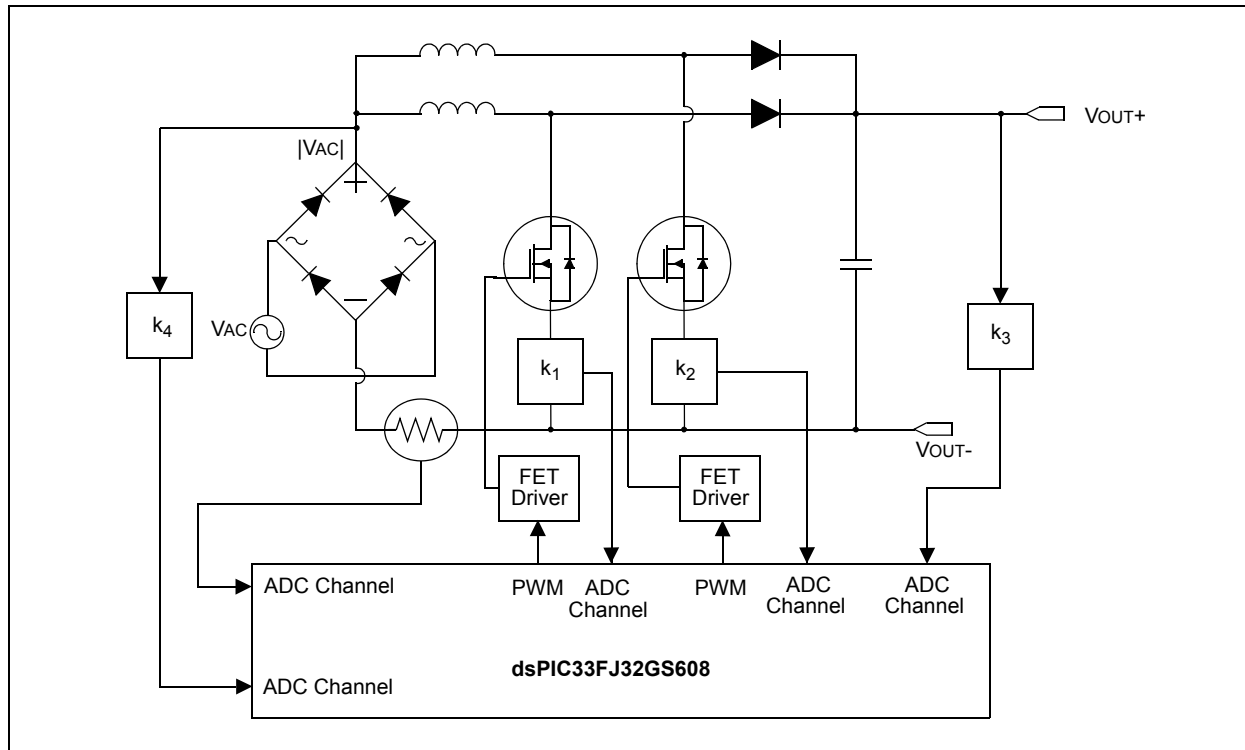


FIGURE 2-10: PHASE-SHIFTED FULL-BRIDGE CONVERTER

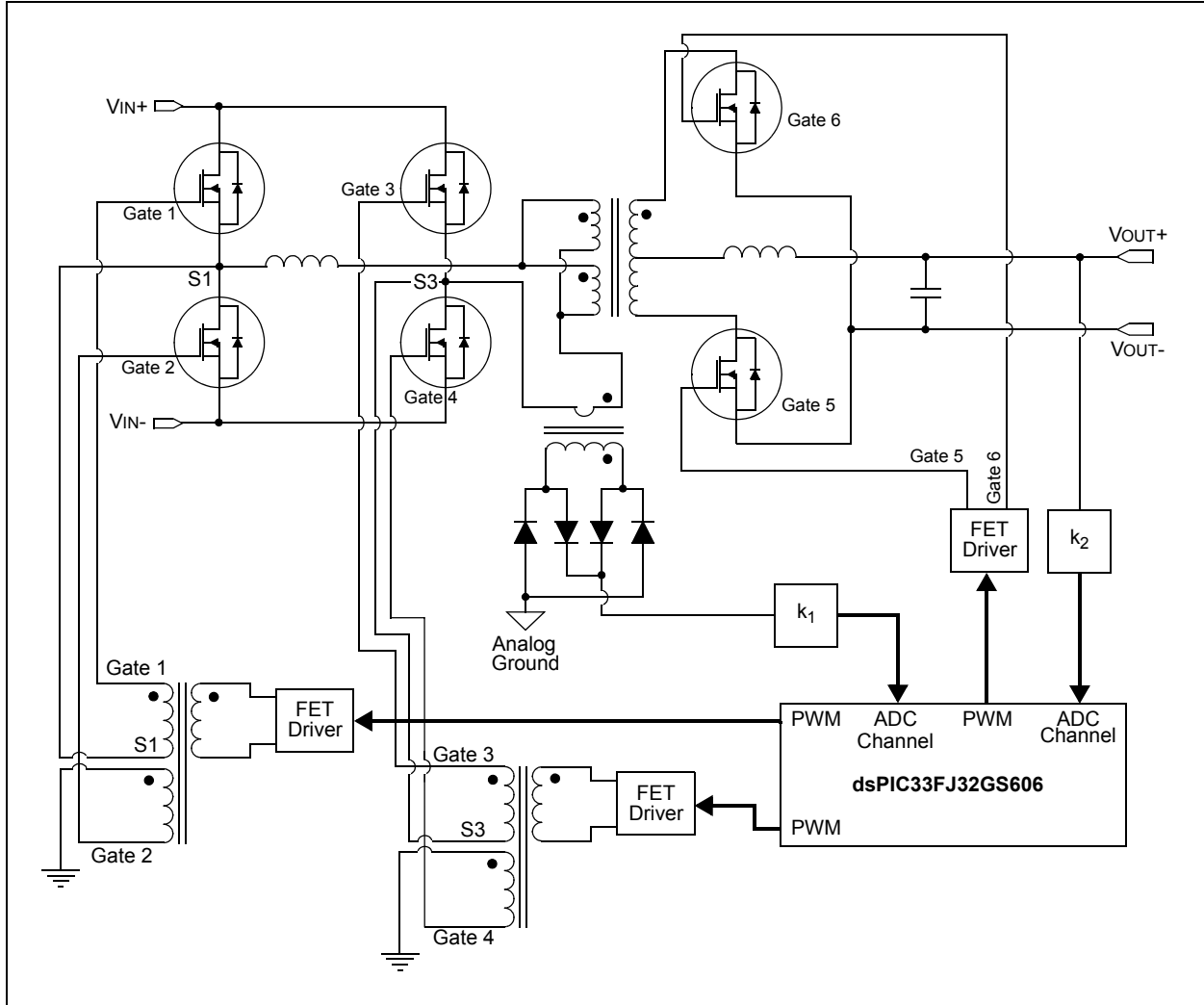
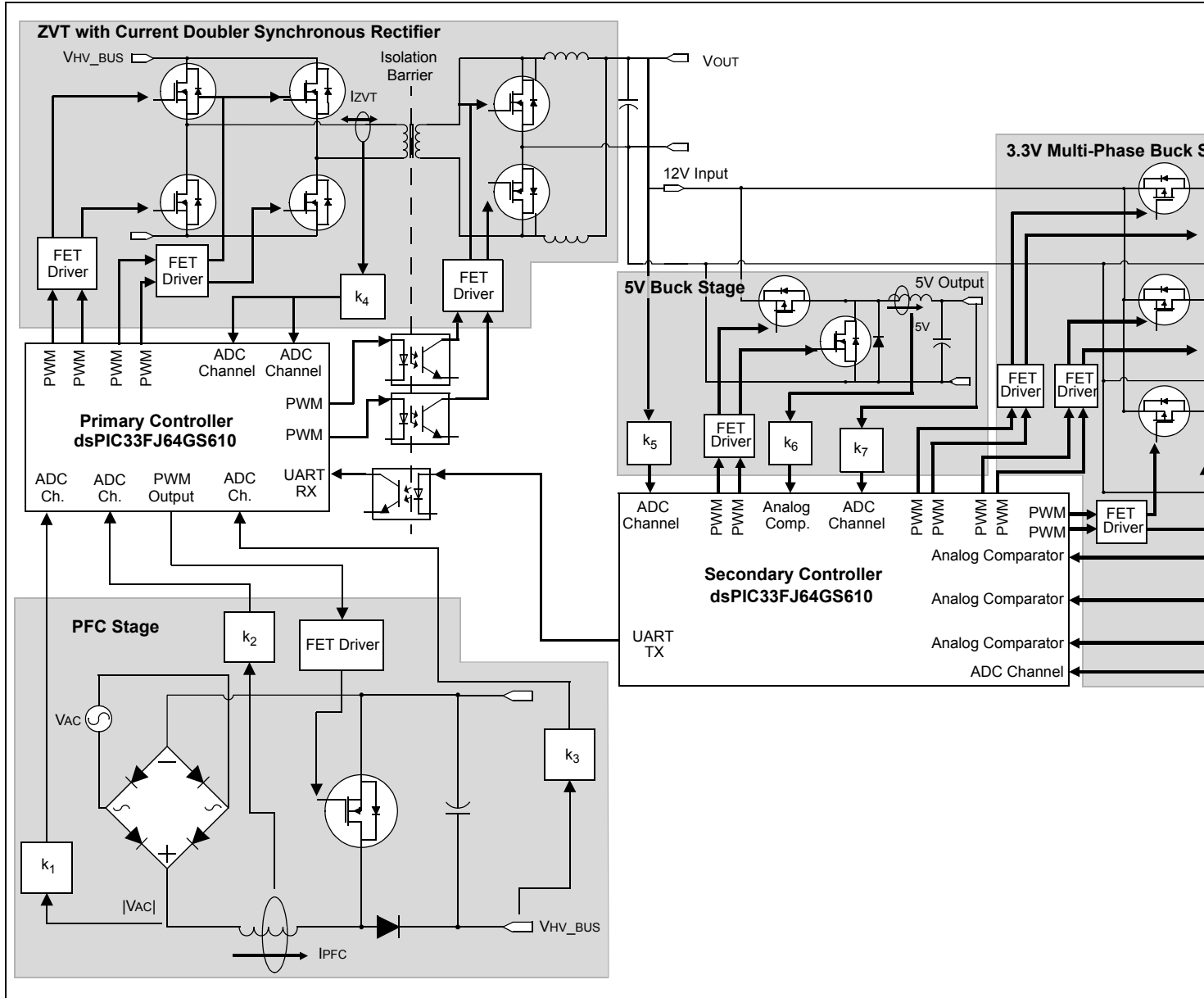


FIGURE 2-11: AC-TO-DC POWER SUPPLY WITH PFC AND THREE OUTPUTS (12V, 5V, AND 3.3V)





## 3.0 CPU

**Note 1:** This data sheet summarizes the features of the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 families of devices. 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](http://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 dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 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 from device to 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 dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 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 sixteenth working register (W15) operates as a software Stack Pointer (SP) for interrupts and calls.

There are two classes of instruction in the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 devices: 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 dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 is capable of executing a data (or program data) memory read, a working register (data) read, a data memory write and a program (instruction) memory read per instruction cycle. As a result, three parameter instructions can be supported, allowing  $A + B = C$  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 dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 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 (PSVPAG) register. The program-to-data space mapping feature lets any instruction access program space as if it were data space.

### 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 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 dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 features 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 dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 supports 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 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: dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 CPU CORE BLOCK DIAGRAM**

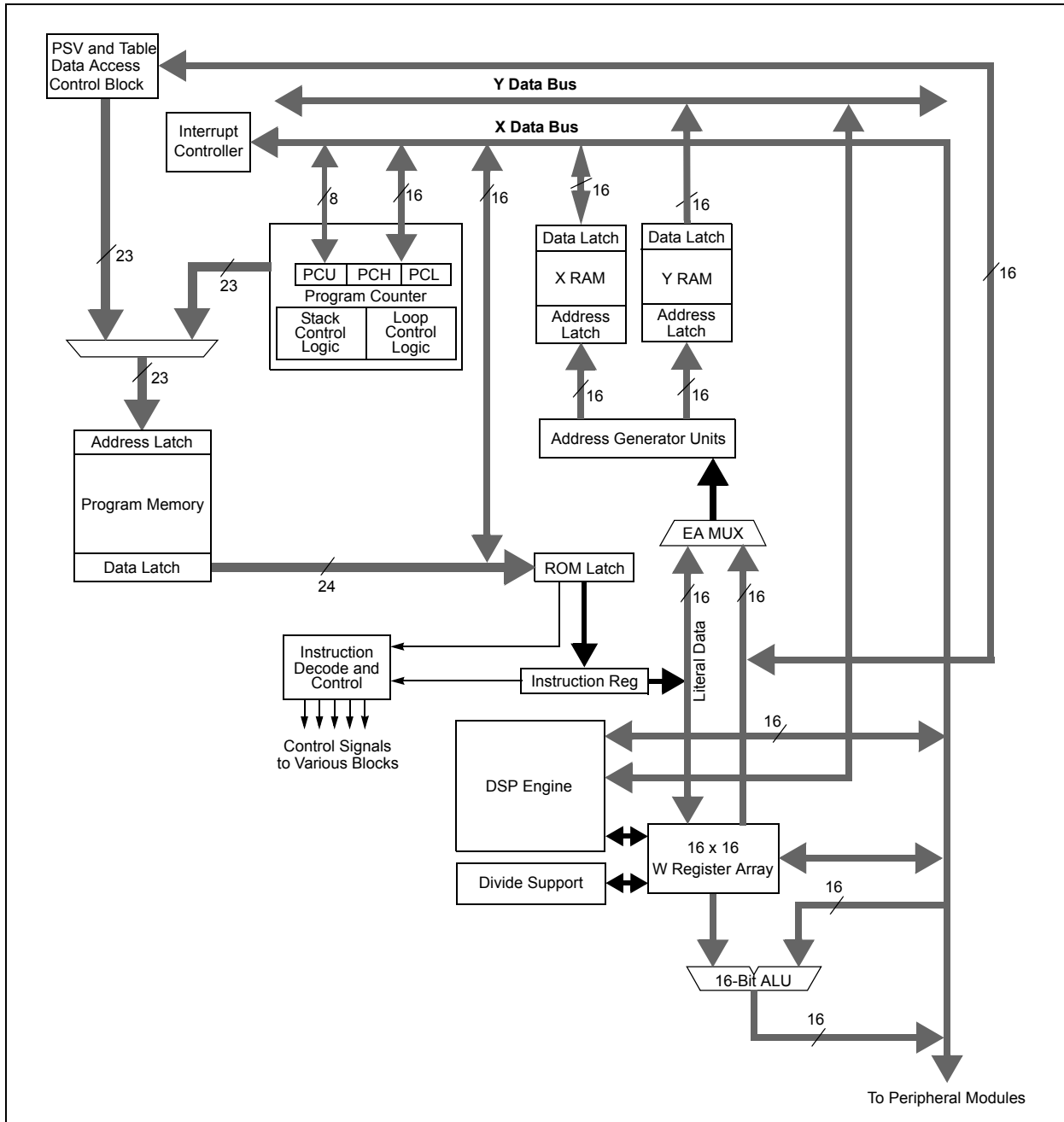
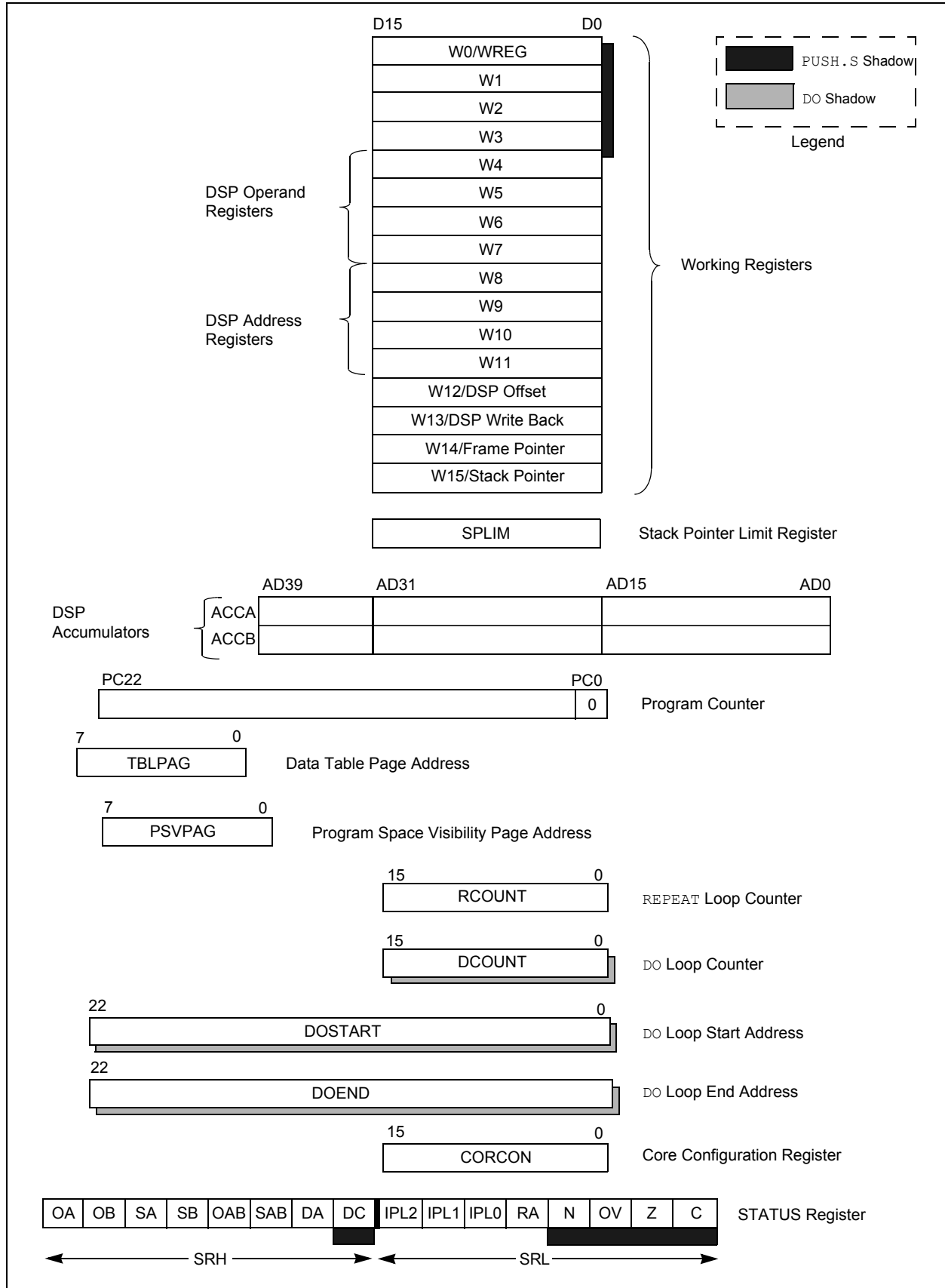


FIGURE 3-2: PROGRAMMER'S MODEL



## 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	OB	SA <sup>(1)</sup>	SB <sup>(1)</sup>	OAB	SAB <sup>(1,4)</sup>	DA	DC
bit 15							bit 8

R/W-0 <sup>(2)</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	C
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<sup>(1)</sup>  
             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<sup>(1,4)</sup>  
             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 can 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 (IPL). 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:** Clearing this bit will clear SA and SB.

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

bit 7-5	<b>IPL&lt;2:0&gt;</b> : 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)
bit 4	<b>RA</b> : REPEAT Loop Active bit 1 = REPEAT loop in progress 0 = REPEAT loop not in progress
bit 3	<b>N</b> : MCU ALU Negative bit 1 = Result was negative 0 = Result was non-negative (zero or positive)
bit 2	<b>OV</b> : MCU ALU Overflow bit This bit is used for signed arithmetic (2's complement). It indicates an overflow of a 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	<b>Z</b> : MCU ALU Zero bit 1 = An operation that affects the Z bit has set it at some time in the past 0 = The most recent operation that affects the Z bit has cleared it (i.e., a non-zero result)
bit 0	<b>C</b> : 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 can 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 (IPL). 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:** Clearing this bit will clear SA and SB.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## 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

<b>Legend:</b>	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 unsigned
  - 0 = DSP engine multiplies are signed
- bit 11       **EDT:** Early  $\text{DO}$  Loop Termination Control bit<sup>(1)</sup>
  - 1 = Terminate executing  $\text{DO}$  loop at end of current loop iteration
  - 0 = No effect
- bit 10-8     **DL<2:0>:**  $\text{DO}$  Loop Nesting Level Status bits
  - 111 = 7  $\text{DO}$  loops active
  - 
  - 
  - 
  - 001 = 1  $\text{DO}$  loop active
  - 000 = 0  $\text{DO}$  loops active
- bit 7        **SATA:** ACCA Saturation Enable bit
  - 1 = Accumulator A saturation enabled
  - 0 = Accumulator A saturation disabled
- bit 6        **SATB:** ACCB Saturation Enable bit
  - 1 = Accumulator B saturation enabled
  - 0 = Accumulator B saturation disabled
- bit 5        **SATDW:** Data Space Write from DSP Engine Saturation Enable bit
  - 1 = Data space write saturation enabled
  - 0 = 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 space
  - 0 = Program space not visible in data space
- bit 1        **RND:** Rounding Mode Select bit
  - 1 = Biased (conventional) rounding enabled
  - 0 = Unbiased (convergent) rounding enabled
- bit 0        **IF:** Integer or Fractional Multiplier Mode Select bit
  - 1 = Integer mode enabled for DSP multiply ops
  - 0 = 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 dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 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 can 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 dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 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:

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

**3.5.2 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:

- 32-bit signed/16-bit signed divide
- 32-bit unsigned/16-bit unsigned divide
- 16-bit signed/16-bit signed divide
- 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/subtractor (with two target accumulators, round and saturation logic).

The dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 is 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 can be used concurrently by the same instruction (for example, ED, EDAC).

The DSP engine can also perform inherent accumulator-to-accumulator operations that require no additional data. These instructions are ADD, SUB and NEG.

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

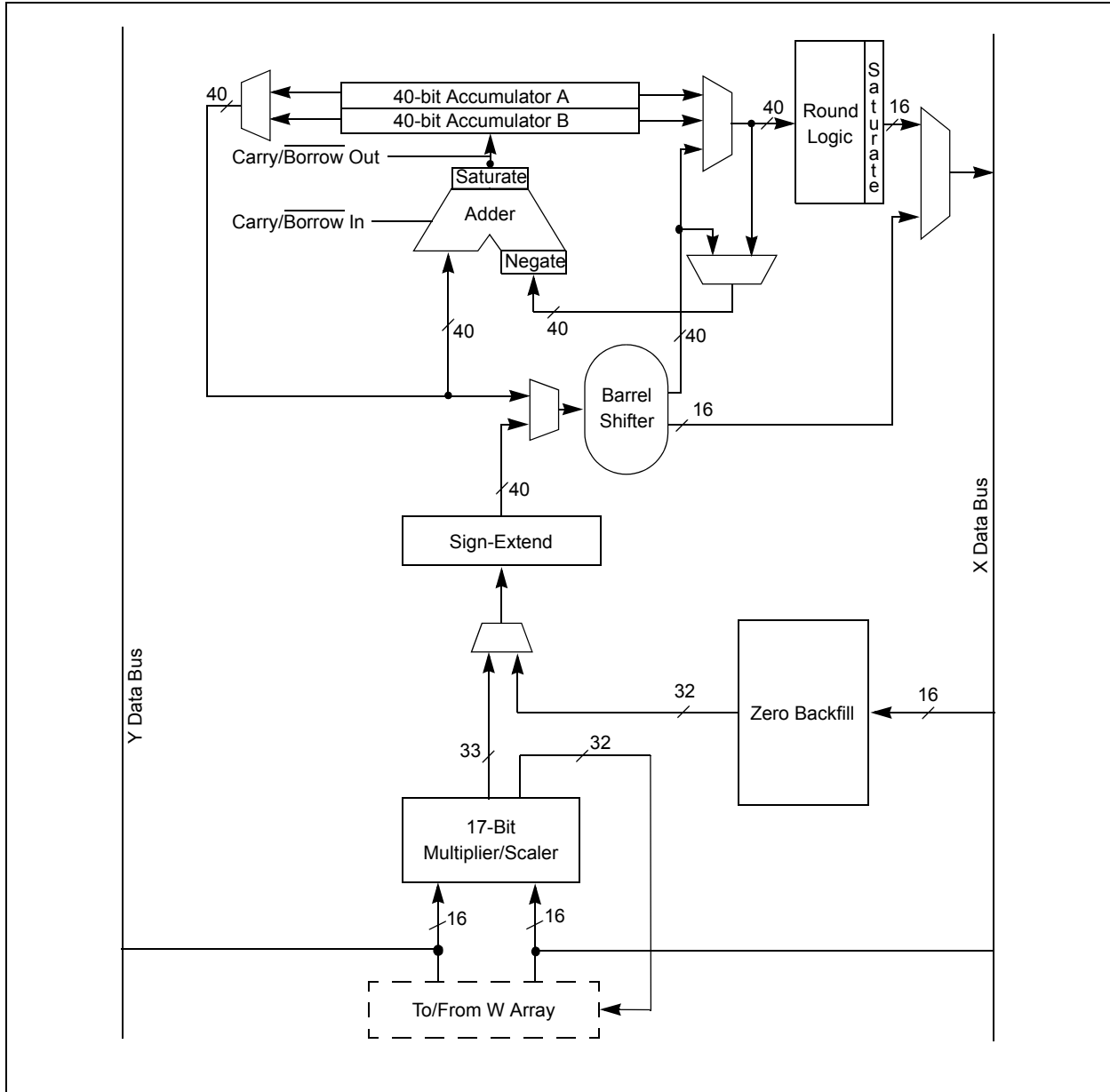
- Fractional or integer DSP multiply (IF)
- Signed or unsigned DSP multiply (US)
- Conventional or convergent rounding (RND)
- Automatic saturation on/off for ACCA (SATA)
- Automatic saturation on/off for ACCB (SATB)
- Automatic saturation on/off for writes to data memory (SATDW)
- Accumulator Saturation mode selection (ACC-SAT)

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	$A = A + (x * y)$	Yes
MAC	$A = A + x^2$	No
MOVSAC	No change in A	Yes
MPY	$A = x * y$	No
MPY	$A = x^2$	No
MPY.N	$A = -x * y$	No
MSC	$A = A - x * y$	Yes

FIGURE 3-3: DSP ENGINE BLOCK DIAGRAM





### 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 that is sign-extended to 40 bits. Integer data is inherently represented as a signed 2's complement value, where the Most Significant bit (MSb) is defined as a sign bit. The range of an N-bit 2'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 2'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 2'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  $3.01518 \times 10^{-5}$ . In Fractional mode, the 16 x 16 multiply operation generates a 1.31 product that 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 multiply operations.

The `MUL` instruction can 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/subtractor 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 using the barrel shifter prior to accumulation.

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

The adder/subtractor 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).
- In the case of subtraction, the Carry/Borrow input is active-low and the other input is complemented.

The adder/subtractor 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 that controls accumulator data saturation, if selected. It uses the result of the adder, the Overflow Status bits described previously 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 support saturation and overflow:

- OA: ACCA overflowed into guard bits
- OB: ACCB overflowed into guard bits
- SA: ACCA saturated (bit 31 overflow and saturation)  
or  
ACCA overflowed into guard bits and saturated (bit 39 overflow and saturation)
- SB: ACCB saturated (bit 31 overflow and saturation)  
or  
ACCB overflowed into guard bits and saturated (bit 39 overflow and saturation)
- OAB: 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/subtractor. 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 set and the corresponding Overflow Trap Flag Enable bits (OVATE, OVBTE) in the INTCON1 register are set (refer to [Section 7.0 "Interrupt Controller"](#)). This allows the user application to take immediate action, for example, to correct system gain.

The SA and SB bits are modified each time data passes through the adder/subtractor, but can only be cleared by the user application. 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). Programmers can 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 is useful for complex number arithmetic, which typically uses both accumulators.

The device supports three Saturation and Overflow modes:

- **Bit 39 Overflow and Saturation:**  
When bit 39 overflow and saturation occurs, the saturation logic loads the maximally positive 9.31 (0x7FFFFFFF) or maximally negative 9.31 value (0x80000000) into the target accumulator. The SA or SB bit is set and remains set until cleared by the user application. This condition is referred to as 'super saturation' and provides protection against erroneous data or unexpected algorithm problems (such as gain calculations).
- **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 application. When this Saturation mode is in effect, the guard bits are not used, so the OA, OB or OAB bits are never set.
- **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 application. 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.

### 3.6.3 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:

- **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).

#### 3.6.3.1 Round Logic

The round logic is a combinational block that 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 that 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 3.6.3.2 "Data Space Write Saturation"](#)). For the MAC class of instructions, the accumulator write-back operation functions in the same manner, addressing combined MCU (X and Y) data space through the X bus. For this class of instructions, the data is always subject to rounding.

### 3.6.3.2 Data Space Write Saturation

In addition to adder/subtractor 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.

### 3.6.4 BARREL SHIFTER

The barrel shifter can perform 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 and 31 for right shifts, and between bit positions 0 and 16 for left shifts.

NOTES:

4.0 MEMORY ORGANIZATION

**Note:** This data sheet summarizes the features of the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 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 4. Program Memory” (DS70203), which is available from the Microchip web site ([www.microchip.com](http://www.microchip.com)).

The dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 architecture features separate program and data memory spaces and buses. This architecture also allows the direct access to program memory from the data space during code execution.

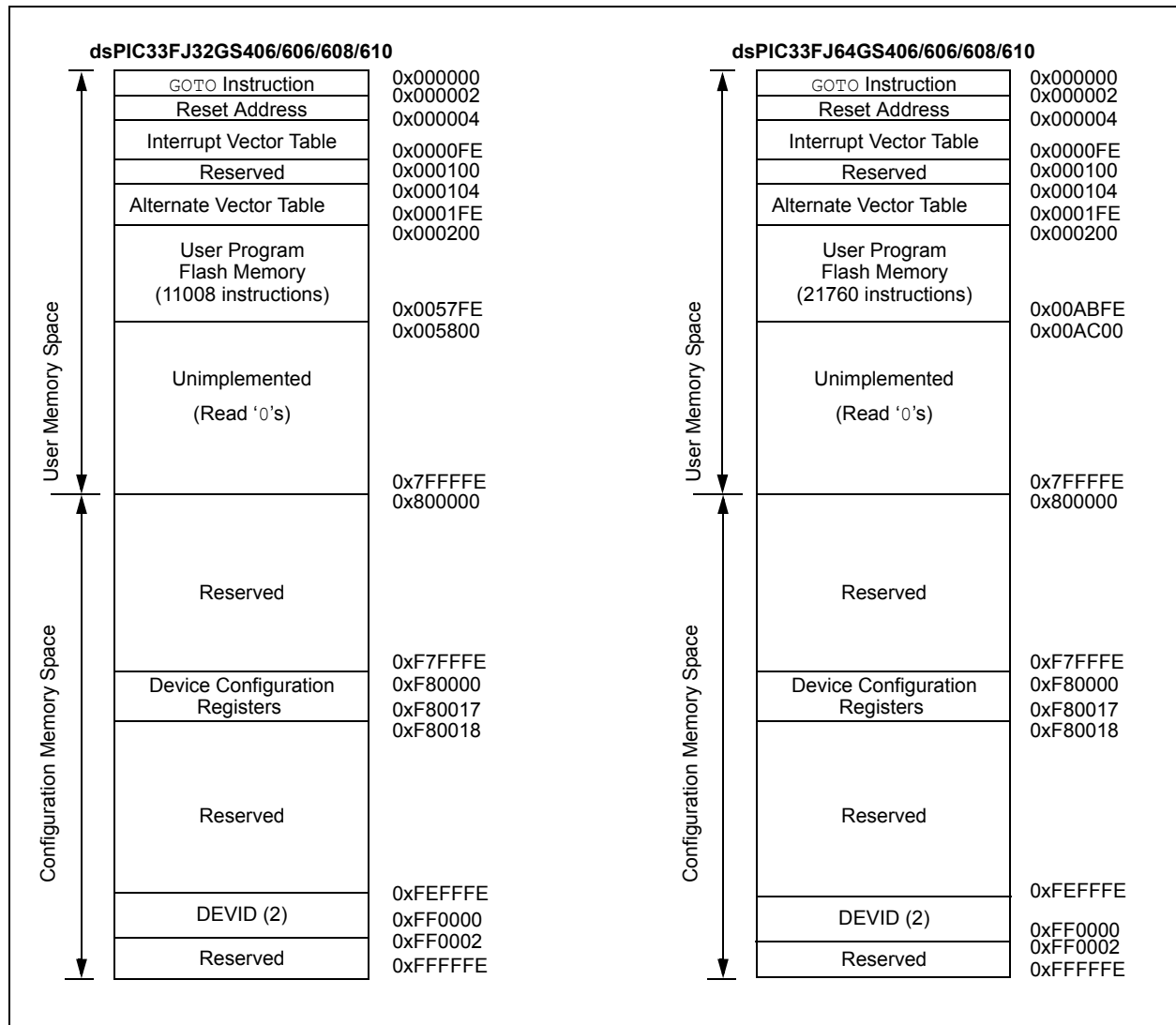
4.1 Program Address Space

The program address memory space is 4M instructions. The space is addressable by a 24-bit value derived either from 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 application access to the program memory space is restricted to the lower half of the address range (0x000000 to 0x7FFFFFFF). 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.

The memory maps are shown in Figure 4-1.

FIGURE 4-1: PROGRAM MEMORY MAPS FOR dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 DEVICES



## 4.1.1 PROGRAM MEMORY ORGANIZATION

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

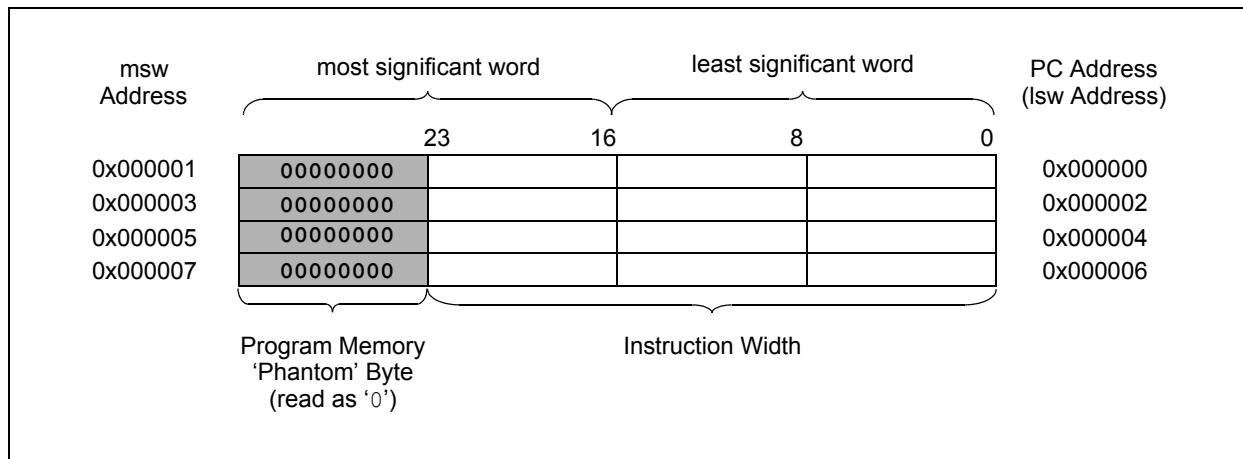
Program memory addresses are always word-aligned on the lower word, and addresses are incremented or decremented by two during the code execution. This arrangement provides compatibility with data memory space addressing and makes data in the program memory space accessible.

## 4.1.2 INTERRUPT AND TRAP VECTORS

All dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 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 application at 0x000000, with the actual address for the start of code at 0x000002.

The dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 devices also have two interrupt vector tables, located from 0x000004 to 0x0000FF and 0x000100 to 0x0001FF. These vector tables allow each of the 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”.

**FIGURE 4-2: PROGRAM MEMORY ORGANIZATION**



## 4.2 Data Address Space

The 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. The data memory maps is shown in [Figure 4-3](#).

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”](#)).

The dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 devices implement up to 9 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 (LSBs) of each word have even addresses, while the Most Significant Bytes (MSBs) have odd addresses.

### 4.2.2 DATA MEMORY ORGANIZATION AND ALIGNMENT

To maintain backward compatibility with PIC<sup>®</sup> MCU devices and improve data space memory usage efficiency, the 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++] that results 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 that 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 the error occurred on a write, the instruction is executed but the write does not occur. In either case, a trap is then executed, allowing the system and/or user application 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 user applications to translate 8-bit signed data to 16-bit signed values. Alternatively, for 16-bit unsigned data, user applications 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 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'.

<p><b>Note:</b> The actual set of peripheral features and interrupts varies by the device. Refer to the corresponding device tables and pin-out diagrams for device-specific information.</p>
---

### 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 DEVICES WITH 4 KB RAM**

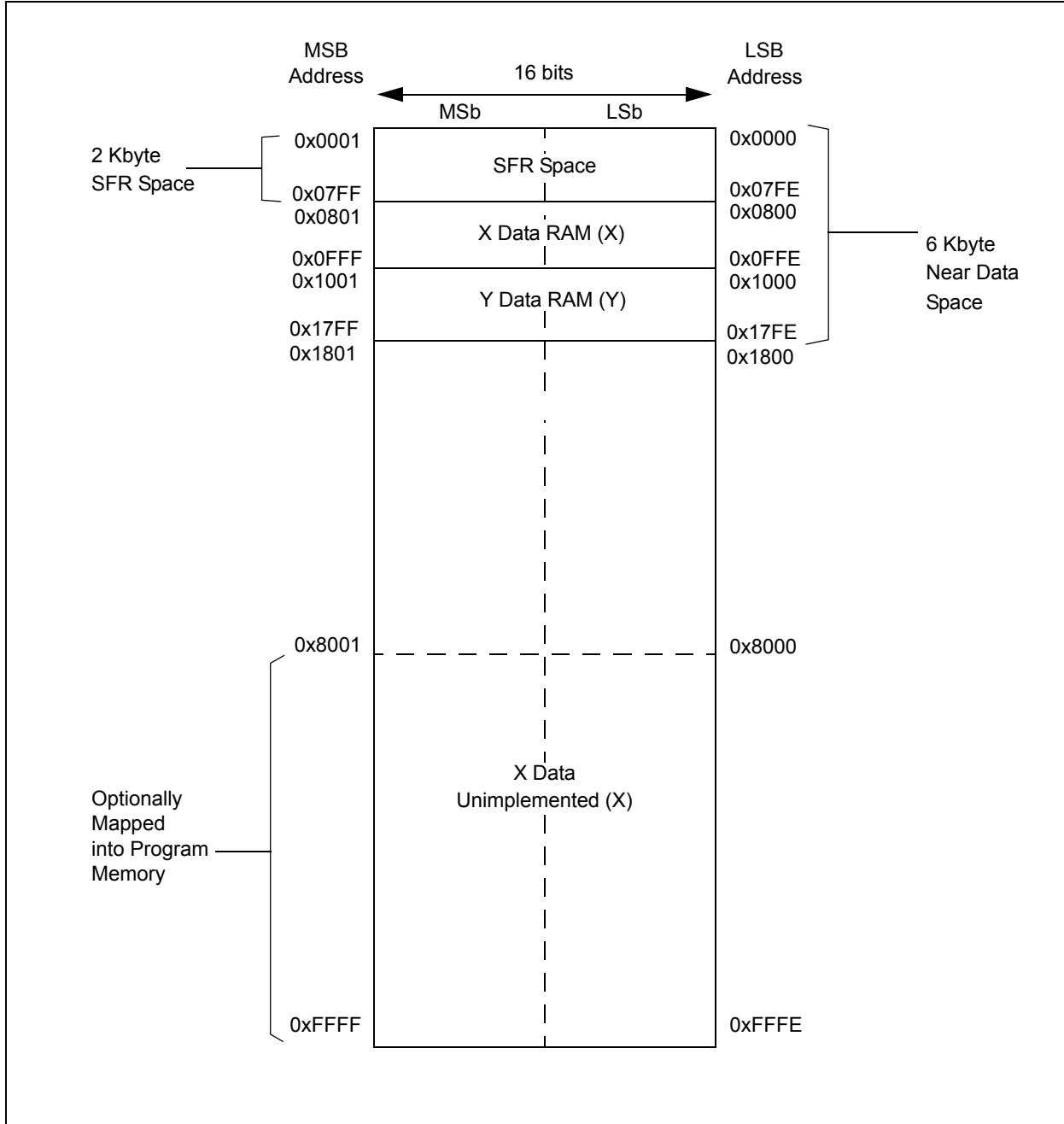
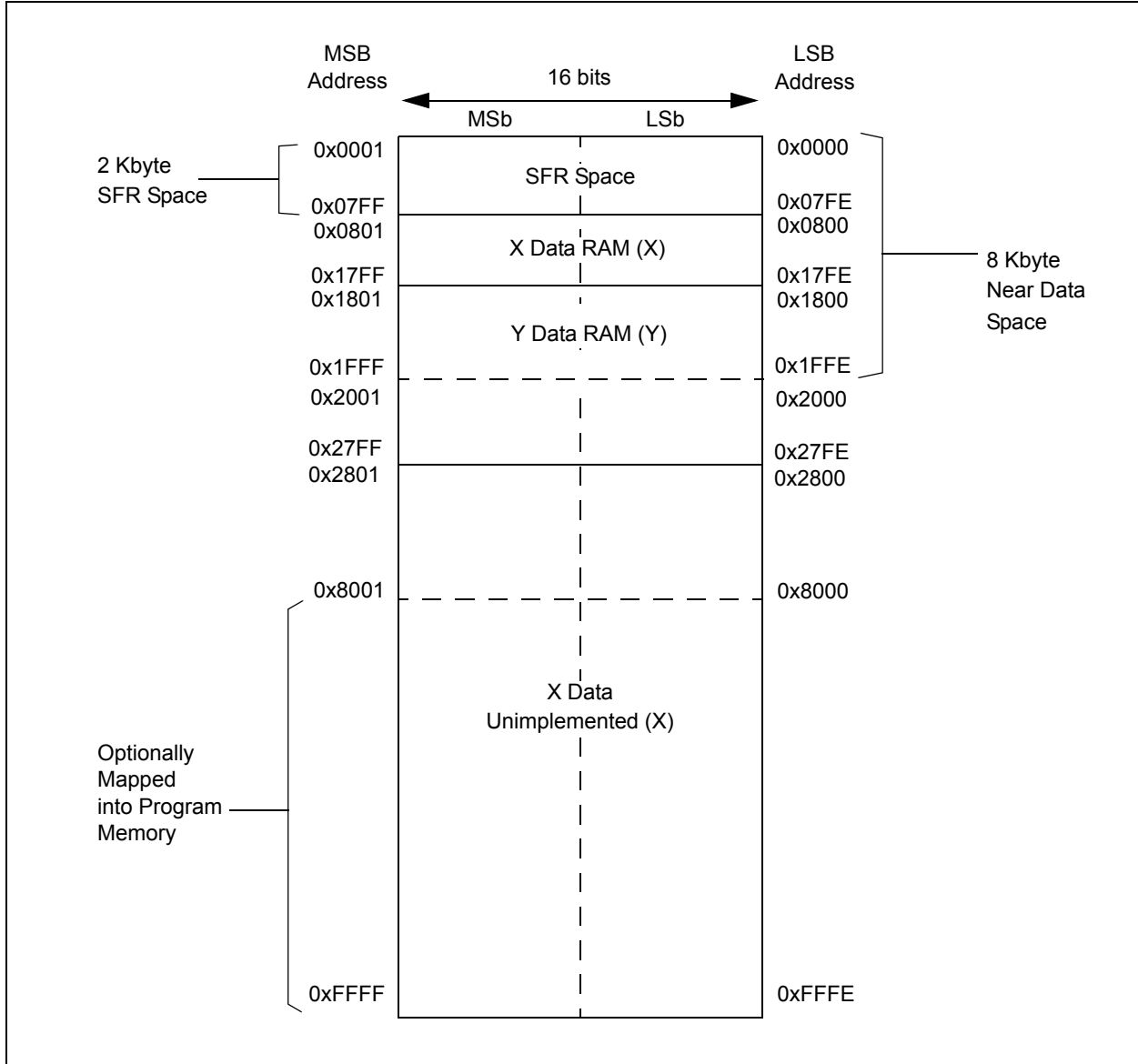
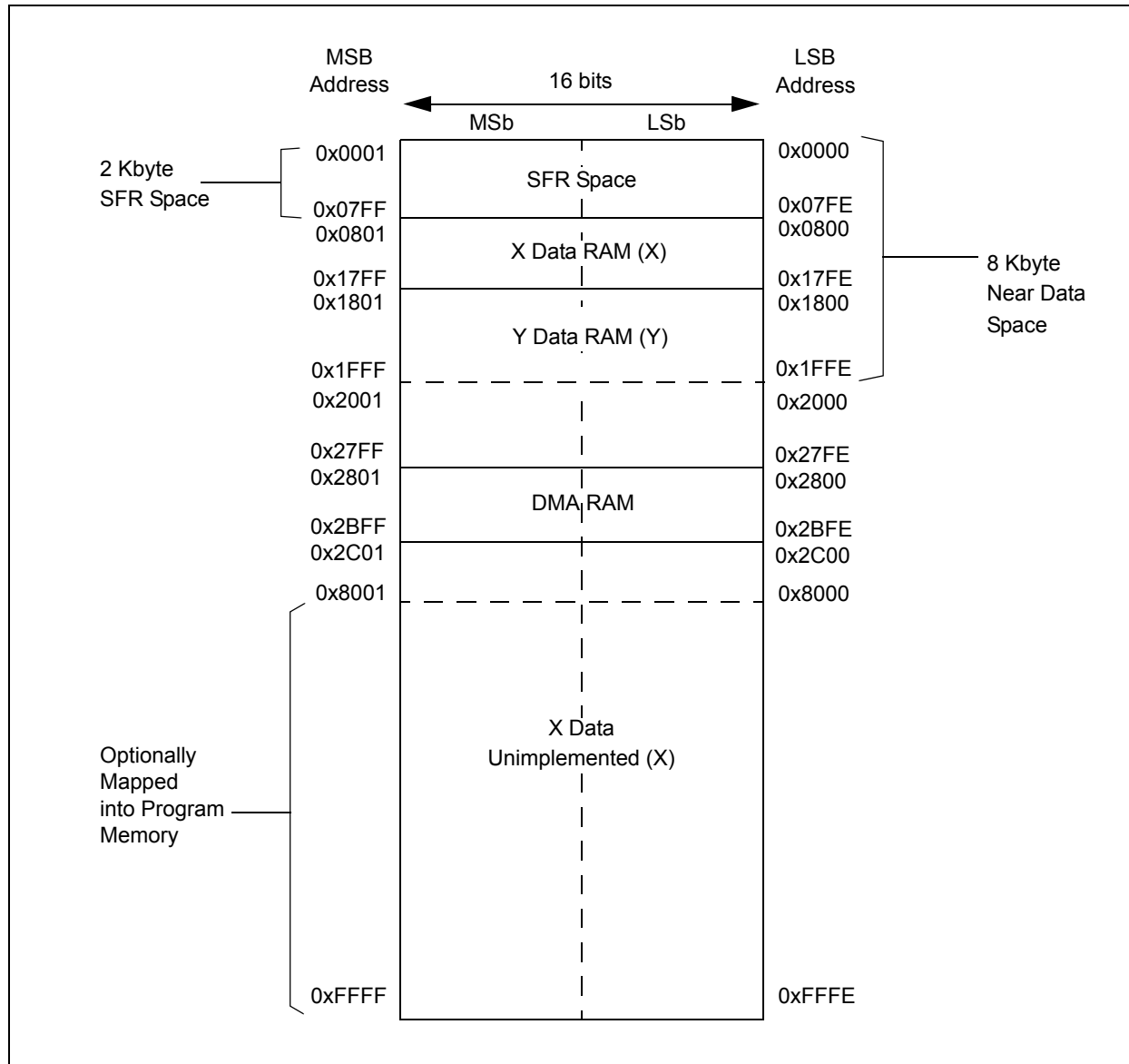




FIGURE 4-4: DATA MEMORY MAP FOR DEVICES WITH 8 KB RAM



**FIGURE 4-5: DATA MEMORY MAP FOR DEVICES WITH 9 KB 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. X data space has separate read and write data buses. 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, MOV SAC, 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

Some devices contain 1 Kbyte of dual ported DMA RAM, which is located at the end of Y data space. Memory locations that are part of Y data RAM and are in the DMA RAM space are 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 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
WREG0	0000	Working Register 0												
WREG1	0002	Working Register 1												
WREG2	0004	Working Register 2												
WREG3	0006	Working Register 3												
WREG4	0008	Working Register 4												
WREG5	000A	Working Register 5												
WREG6	000C	Working Register 6												
WREG7	000E	Working Register 7												
WREG8	0010	Working Register 8												
WREG9	0012	Working Register 9												
WREG10	0014	Working Register 10												
WREG11	0016	Working Register 11												
WREG12	0018	Working Register 12												
WREG13	001A	Working Register 13												
WREG14	001C	Working Register 14												
WREG15	001E	Working Register 15												
SPLIM	0020	Stack Pointer Limit Register												
ACCAL	0022	ACCAL												
ACCAH	0024	ACCAH												
ACCAU	0026	ACCA<39>	ACCA<39>	ACCA<39>	ACCA<39>	ACCA<39>	ACCA<39>	ACCA<39>	ACCA<39>	ACCA<39>	ACCAU			
ACCBL	0028	ACCBL												
ACCBH	002A	ACCBH												
ACCBU	002C	ACCB<39>	ACCB<39>	ACCB<39>	ACCB<39>	ACCB<39>	ACCB<39>	ACCB<39>	ACCB<39>	ACCB<39>	ACCBU			
PCL	002E	Program Counter Low Word Register												
PCH	0030	—	—	—	—	—	—	—	—	—	Program Counter High Byte			
TBLPAG	0032	—	—	—	—	—	—	—	—	Table Page Address Pointer				
PSVPAG	0034	—	—	—	—	—	—	—	—	Program Memory Visibility Page Address				
RCOUNT	0036	Repeat Loop Counter Register												
DCOUNT	0038	DCOUNT<15:0>												
DOSTARTL	003A	DOSTARTL<15:1>												
DOSTARTH	003C	—	—	—	—	—	—	—	—	—	—	DOSTARTH		
DOENDL	003E	DOENDL<15:1>												
DOENDH	0040	—	—	—	—	—	—	—	—	—	DOENDH			
SR	0042	OA	OB	SA	SB	OAB	SAB	DA	DC	IPL2	IPL1	IPL0	RA	N
CORCON	0044	—	—	—	US	EDT	DL<2:0>			SATA	SATB	SATDW	ACCSAT	IPL3
MODCON	0046	XMODEN	YMODEN	—	—	BWM<3:0>				YWM<3:0>				

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

**TABLE 4-1: CPU CORE REGISTER 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
XMODSRT	0048	XS<15:1>												
XMODEND	004A	XE<15:1>												
YMODSRT	004C	YS<15:1>												
YMODEND	004E	YE<15:1>												
XBREV	0050	BREN												
DISICNT	0052	—	—											

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

**TABLE 4-2: CHANGE NOTIFICATION REGISTER MAP FOR dsPIC33FJ32GS608/610 AND dsPIC33FJ64GS608/610**

File 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
CNEN1	0060	CN15IE	CN14IE	CN13IE	CN12IE	CN11IE	CN10IE	CN9IE	CN8IE	CN7IE	CN6IE	CN5IE	CN4IE	CN3IE	CN2IE
CNEN2	0062	—	—	—	—	—	—	—	—	CN23IE	CN22IE	CN21IE	CN20IE	CN19IE	CN18IE
CNPU1	0068	CN15PUE	CN14PUE	CN13PUE	CN12PUE	CN11PUE	CN10PUE	CN9PUE	CN8PUE	CN7PUE	CN6PUE	CN5PUE	CN4PUE	CN3PUE	CN2PUE
CNPU2	006A	—	—	—	—	—	—	—	—	CN23PUE	CN22PUE	CN21PUE	CN20PUE	CN19PUE	CN18PUE

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

**TABLE 4-3: CHANGE NOTIFICATION REGISTER MAP FOR dsPIC33FJ32GS406/606 AND dsPIC33FJ64GS406/606**

File 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
CNEN1	0060	CN15IE	CN14IE	CN13IE	CN12IE	CN11IE	CN10IE	CN9IE	CN8IE	CN7IE	CN6IE	CN5IE	CN4IE	CN3IE	CN2IE
CNEN2	0062	—	—	—	—	—	—	—	—	CN23IE	CN22IE	—	—	—	CN18IE
CNPU1	0068	CN15PUE	CN14PUE	CN13PUE	CN12PUE	CN11PUE	CN10PUE	CN9PUE	CN8PUE	CN7PUE	CN6PUE	CN5PUE	CN4PUE	CN3PUE	CN2PUE
CNPU2	006A	—	—	—	—	—	—	—	—	CN23PUE	CN22PUE	—	—	—	CN18PUE

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

**TABLE 4-4: INTERRUPT CONTROLLER REGISTER MAP FOR dsPIC33FJ64GS610 DEVICES**

File 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
INTCON1	0080	NSTDIS	OVAERR	OVBERR	COVAERR	COVBERR	OVATE	OVBTE	COVTE	SFTACERR	DIV0ERR	DMACERR	MATHERR	ADDRERR	STKE
INTCON2	0082	ALTIVT	DISI	—	—	—	—	—	—	—	—	—	INT4EP	INT3EP	INT2EP
IFS0	0084	—	DMA1IF	ADIF	U1TXIF	U1RXIF	SPI1IF	SPI1EIF	T3IF	T2IF	OC2IF	IC2IF	DMA0IF	T1IF	OC1IF
IFS1	0086	U2TXIF	U2RXIF	INT2IF	T5IF	T4IF	OC4IF	OC3IF	DMA2IF	—	—	—	INT1IF	CNIF	AC1IF
IFS2	0088	—	—	—	—	—	—	—	—	—	IC4IF	IC3IF	DMA3IF	C1IF	C1RXIF
IFS3	008A	—	—	—	—	—	QE11IF	PSEMIF	—	—	INT4IF	INT3IF	—	—	MI2C1IF
IFS4	008C	—	—	—	—	QE12IF	—	PSESMIF	—	—	C1TXIF	—	—	—	U2EIF
IFS5	008E	PWM2IF	PWM1IF	ADCP12IF	—	—	—	—	—	—	—	—	ADCP11IF	ADCP10IF	ADCP9IF
IFS6	0090	ADCP11IF	ADCP0IF	—	—	—	—	AC4IF	AC3IF	AC2IF	PWM9IF	PWM8IF	PWM7IF	PWM6IF	PWM5IF
IFS7	0092	—	—	—	—	—	—	—	—	—	—	ADCP7IF	ADCP6IF	ADCP5IF	ADCP4IF
IEC0	0094	—	DMA1IE	ADIE	U1TXIE	U1RXIE	SPI1IE	SPI1EIE	T3IE	T2IE	OC2IE	IC2IE	DMA0IE	T1IE	OC1IE
IEC1	0096	U2TXIE	U2RXIE	INT2IE	T5IE	T4IE	OC4IE	OC3IE	DMA2IE	—	—	—	INT1IE	CNIE	AC1IE
IEC2	0098	—	—	—	—	—	—	—	—	—	IC4IE	IC3IE	DMA3IE	C1IE	C1RXIE
IEC3	009A	—	—	—	—	—	QE11IE	PSEMIE	—	—	INT4IE	INT3IE	—	—	MI2C1IE
IEC4	009C	—	—	—	—	QE12IE	—	PSESMIE	—	—	C1TXIE	—	—	—	U2EIE
IEC5	009E	PWM2IE	PWM1IE	ADCP12IE	—	—	—	—	—	—	—	—	ADCP11IE	ADCP10IE	ADCP9IE
IEC6	00A0	ADCP11IE	ADCP0IE	—	—	—	—	AC4IE	AC3IE	AC2IE	PWM9IE	PWM8IE	PWM7IE	PWM6IE	PWM5IE
IEC7	00A2	—	—	—	—	—	—	—	—	—	—	ADCP7IE	ADCP6IE	ADCP5IE	ADCP4IE
IPC0	00A4	—	T1IP<2:0>			—	OC1IP<2:0>			—	IC1IP<2:0>			—	—
IPC1	00A6	—	T2IP<2:0>			—	OC2IP<2:0>			—	IC2IP<2:0>			—	—
IPC2	00A8	—	U1RXIP<2:0>			—	SPI1IP<2:0>			—	SPI1EIP<2:0>			—	—
IPC3	00AA	—	—	—	—	—	DMA1IP<2:0>			—	ADIP<2:0>			—	—
IPC4	00AC	—	CNIP<2:0>			—	AC1IP<2:0>			—	MI2C1IP<2:0>			—	—
IPC5	00AE	—	—	—	—	—	—	—	—	—	—	—	—	—	—
IPC6	00B0	—	T4IP<2:0>			—	OC4IP<2:0>			—	OC3IP<2:0>			—	—
IPC7	00B2	—	U2TXIP<2:0>			—	U2RXIP<2:0>			—	INT2IP<2:0>			—	—
IPC8	00B4	—	C1IP<2:0>			—	C1RXIP<2:0>			—	SPI2IP<2:0>			—	—
IPC9	00B6	—	—	—	—	—	IC4IP<2:0>			—	IC3IP<2:0>			—	—
IPC12	00BC	—	—	—	—	—	MI2C2IP<2:0>			—	SI2C2IP<2:0>			—	—
IPC13	00BE	—	—	—	—	—	INT4IP<2:0>			—	INT3IP<2:0>			—	—
IPC14	00C0	—	—	—	—	—	QE11IP<2:0>			—	PSEMIP<2:0>			—	—
IPC16	00C4	—	—	—	—	—	U2EIP<2:0>			—	U1EIP<2:0>			—	—
IPC17	00C6	—	—	—	—	—	C1TXIP<2:0>			—	—	—	—	—	—
IPC18	00C8	—	QE12IP<2:0>			—	—	—	—	—	PSESMIP<2:0>			—	—
IPC20	00CC	—	ADCP10IP<2:0>			—	ADCP9IP<2:0>			—	ADCP8IP<2:0>			—	—

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

**TABLE 4-4: INTERRUPT CONTROLLER REGISTER MAP FOR dsPIC33FJ64GS610 DEVICES (CONTINUED)**

File 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
IPC21	00CE	—	—	—	—	—	—	—	—	—	ADCP12IP<2:0>			—	—
IPC23	00D2	—	PWM2IP<2:0>			—	PWM1IP<2:0>			—	—	—	—	—	—
IPC24	00D4	—	PWM6IP<2:0>			—	PWM5IP<2:0>			—	PWM4IP<2:0>			—	—
IPC25	00D6	—	AC2IP<2:0>			—	PWM9IP<2:0>			—	PWM8IP<2:0>			—	—
IPC26	00D8	—	—	—	—	—	—	—	—	—	AC4IP<2:0>			—	—
IPC27	00DA	—	ADCP1IP<2:0>			—	ADCP0IP<2:0>			—	—	—	—	—	—
IPC28	00DC	—	ADCP5IP<2:0>			—	ADCP4IP<2:0>			—	ADCP3IP<2:0>			—	—
IPC29	00DE	—	—	—	—	—	—	—	—	—	ADCP7IP<2:0>			—	—
INTTREG	00E0	—	—	—	—	ILR<3:0>				—	VECNUM<6:0>				

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



**TABLE 4-5: INTERRUPT CONTROLLER REGISTER MAP FOR dsPIC33FJ64GS608 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
INTCON1	0080	NSTDIS	OVAERR	OVBERR	COVAERR	COVBERR	OVATE	OVBTE	COVTE	SFTACERR	DIV0ERR	DMACERR	MATHERR	ADDRERR	STKE
INTCON2	0082	ALTIPT	DISI	—	—	—	—	—	—	—	—	—	INT4EP	INT3EP	INT2EP
IFS0	0084	—	DMA1IF	ADIF	U1TXIF	U1RXIF	SPI1IF	SPI1EIF	T3IF	T2IF	OC2IF	IC2IF	DMA0IF	T1IF	OC1IF
IFS1	0086	U2TXIF	U2RXIF	INT2IF	T5IF	T4IF	OC4IF	OC3IF	DMA2IF	—	—	—	INT1IF	CNIF	AC1IF
IFS2	0088	—	—	—	—	—	—	—	—	—	IC4IF	IC3IF	DMA3IF	C1IF	C1RXIF
IFS3	008A	—	—	—	—	—	QE11IF	PSEMIF	—	—	INT4IF	INT3IF	—	—	MI2C1IF
IFS4	008C	—	—	—	—	—	QE12IF	PSESMIF	—	—	C1TXIF	—	—	—	U2EIF
IFS5	008E	PWM2IF	PWM1IF	ADCP12IF	—	—	—	—	—	—	—	—	—	—	—
IFS6	0090	ADCP1IF	ADCP0IF	—	—	—	—	AC4IF	AC3IF	AC2IF	—	PWM8IF	PWM7IF	PWM6IF	PWM5IF
IFS7	0092	—	—	—	—	—	—	—	—	—	—	ADCP7IF	ADCP6IF	ADCP5IF	ADCP4IF
IEC0	0094	—	DMA1IE	ADIE	U1TXIE	U1RXIE	SPI1IE	SPI1EIE	T3IE	T2IE	OC2IE	IC2IE	DMA0IE	T1IE	OC1IE
IEC1	0096	U2TXIE	U2RXIE	INT2IE	T5IE	T4IE	OC4IE	OC3IE	DMA2IE	—	—	—	INT1IE	CNIE	AC1IE
IEC2	0098	—	—	—	—	—	—	—	—	—	IC4IE	IC3IE	DMA3IE	C1IE	C1RXIE
IEC3	009A	—	—	—	—	—	QE11IE	PSEMIE	—	—	INT4IE	INT3IE	—	—	MI2C1IE
IEC4	009C	—	—	—	—	—	QE12IE	PSESMIE	—	—	C1TXIE	—	—	—	U2EIE
IEC5	009E	PWM2IE	PWM1IE	ADCP12IE	—	—	—	—	—	—	—	—	—	—	—
IEC6	00A0	ADCP1IE	ADCP0IE	—	—	—	—	AC4IE	AC3IE	AC2IE	—	PWM8IE	PWM7IE	PWM6IE	PWM5IE
IEC7	00A2	—	—	—	—	—	—	—	—	—	—	ADCP7IE	ADCP6IE	ADCP5IE	ADCP4IE
IPC0	00A4	—	T1IP<2:0>			—	OC1IP<2:0>			—	IC1IP<2:0>			—	—
IPC1	00A6	—	T2IP<2:0>			—	OC2IP<2:0>			—	IC2IP<2:0>			—	—
IPC2	00A8	—	U1RXIP<2:0>			—	SPI1IP<2:0>			—	SPI1EIP<2:0>			—	—
IPC3	00AA	—	—	—	—	—	DMA1IP<2:0>			—	ADIP<2:0>			—	—
IPC4	00AC	—	CNIP<2:0>			—	AC1IP<2:0>			—	MI2C1IP<2:0>			—	—
IPC5	00AE	—	—	—	—	—	—	—	—	—	—	—	—	—	—
IPC6	00B0	—	T4IP<2:0>			—	OC4IP<2:0>			—	OC3IP<2:0>			—	—
IPC7	00B2	—	U2TXIP<2:0>			—	U2RXIP<2:0>			—	INT2IP<2:0>			—	—
IPC8	00B4	—	C1IP<2:0>			—	C1RXIP<2:0>			—	SPI2IP<2:0>			—	—
IPC9	00B6	—	—	—	—	—	IC4IP<2:0>			—	IC3IP<2:0>			—	—
IPC12	00BC	—	—	—	—	—	MI2C2IP<2:0>			—	SI2C2IP<2:0>			—	—
IPC13	00BE	—	—	—	—	—	INT4IP<2:0>			—	INT3IP<2:0>			—	—
IPC14	00C0	—	—	—	—	—	QE11IP<2:0>			—	PSEMIP<2:0>			—	—
IPC16	00C4	—	—	—	—	—	U2EIP<2:0>			—	U1EIP<2:0>			—	—
IPC17	00C6	—	—	—	—	—	C1TXIP<2:0>			—	—	—	—	—	—
IPC18	00C8	—	QE12IP<2:0>			—	—	—	—	—	PSESMIP<2:0>			—	—

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

**TABLE 4-5: INTERRUPT CONTROLLER REGISTER MAP FOR dsPIC33FJ64GS608 DEVICES (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
IPC20	00CC	—	—	—	—	—	—	—	—	—	ADCP8IP<2:0>			—	—
IPC21	00CE	—	—	—	—	—	—	—	—	—	ADCP12IP			—	—
IPC23	00D2	—	PWM2IP<2:0>			—	PWM1IP<2:0>			—	—	—	—	—	—
IPC24	00D4	—	PWM6IP<2:0>			—	PWM5IP<2:0>			—	PWM4IP<2:0>			—	—
IPC25	00D6	—	AC2IP<2:0>			—	—	—	—	—	PWM8IP<2:0>			—	—
IPC26	00D8	—	—	—	—	—	—	—	—	—	AC4IP<2:0>			—	—
IPC27	00DA	—	ADCP1IP<2:0>			—	ADCP0IP<2:0>			—	—	—	—	—	—
IPC28	00DC	—	ADCP5IP<2:0>			—	ADCP4IP<2:0>			—	ADCP3IP<2:0>			—	—
IPC29	00DE	—	—	—	—	—	—	—	—	—	ADCP7IP<2:0>			—	—
INTTREG	00E0	—	—	—	—	ILR<3:0>				—	VECNUM<6:0>				

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

**TABLE 4-6: INTERRUPT CONTROLLER REGISTER MAP FOR dsPIC33FJ64GS606 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	
INTCON1	0080	NSTDIS	OVAERR	OVBERR	COVAERR	COVBERR	OVATE	OVBTE	COVTE	SFTACERR	DIV0ERR	DMACERR	MATHERR	ADDRERR	STKE	
INTCON2	0082	ALTIVT	DISI	—	—	—	—	—	—	—	—	—	INT4EP	INT3EP	INT2EP	
IFS0	0084	—	DMA1IF	ADIF	U1TXIF	U1RXIF	SPI1IF	SPI1EIF	T3IF	T2IF	OC2IF	IC2IF	DMA0IF	T1IF	OC1IF	
IFS1	0086	U2TXIF	U2RXIF	INT2IF	T5IF	T4IF	OC4IF	OC3IF	DMA2IF	—	—	—	INT1IF	CNIF	AC1IF	
IFS2	0088	—	—	—	—	—	—	—	—	—	IC4IF	IC3IF	DMA3IF	C1IF	C1RXIF	
IFS3	008A	—	—	—	—	—	QE11F	PSEMIF	—	—	INT4IF	INT3IF	—	—	MI2C1IF	
IFS4	008C	—	—	—	—	QE12IF	—	PSESMIF	—	—	C1TXIF	—	—	—	U2EIF	
IFS5	008E	PWM2IF	PWM1IF	ADCP12IF	—	—	—	—	—	—	—	—	—	—	—	
IFS6	0090	ADCP1IF	ADCP0IF	—	—	—	—	AC4IF	AC3IF	AC2IF	—	—	—	PWM6IF	PWM5IF	
IFS7	0092	—	—	—	—	—	—	—	—	—	—	ADCP7IF	ADCP6IF	ADCP5IF	ADCP4IF	
IEC0	0094	—	DMA1IE	ADIE	U1TXIE	U1RXIE	SPI1IE	SPI1EIE	T3IE	T2IE	OC2IE	IC2IE	DMA0IE	T1IE	OC1IE	
IEC1	0096	U2TXIE	U2RXIE	INT2IE	T5IE	T4IE	OC4IE	OC3IE	DMA2IE	—	—	—	INT1IE	CNIE	AC1IE	
IEC2	0098	—	—	—	—	—	—	—	—	—	IC4IE	IC3IE	DMA3IE	C1IE	C1RXIE	
IEC3	009A	—	—	—	—	—	QE11IE	PSEMIE	—	—	INT4IE	INT3IE	—	—	MI2C1IE	
IEC4	009C	—	—	—	—	QE12IE	—	PSESMIE	—	—	C1TXIE	—	—	—	U2EIE	
IEC5	009E	PWM2IE	PWM1IE	ADCP12IE	—	—	—	—	—	—	—	—	—	—	—	
IEC6	00A0	ADCP1IE	ADCP0IE	—	—	—	—	AC4IE	AC3IE	AC2IE	—	—	—	PWM6IE	PWM5IE	
IEC7	00A2	—	—	—	—	—	—	—	—	—	—	ADCP7IE	ADCP6IE	ADCP5IE	ADCP4IE	
IPC0	00A4	—	T1IP<2:0>			—	OC1IP<2:0>			—	IC1IP<2:0>			—	—	
IPC1	00A6	—	T2IP<2:0>			—	OC2IP<2:0>			—	IC2IP<2:0>			—	—	
IPC2	00A8	—	U1RXIP<2:0>			—	SPI1IP<2:0>			—	SPI1EIP<2:0>			—	—	
IPC3	00AA	—	—	—	—	—	DMA1IP<2:0>			—	ADIP<2:0>			—	—	
IPC4	00AC	—	CNIP<2:0>			—	AC1IP<2:0>			—	MI2C1IP<2:0>			—	—	
IPC5	00AE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
IPC6	00B0	—	T4IP<2:0>			—	OC4IP<2:0>			—	OC3IP<2:0>			—	—	
IPC7	00B2	—	U2TXIP<2:0>			—	U2RXIP<2:0>			—	INT2IP<2:0>			—	—	
IPC8	00B4	—	C1IP<2:0>			—	C1RXIP<2:0>			—	SPI2IP<2:0>			—	—	
IPC9	00B6	—	—	—	—	—	IC4IP<2:0>			—	IC3IP<2:0>			—	—	
IPC12	00BC	—	—	—	—	—	MI2C2IP<2:0>			—	SI2C2IP<2:0>			—	—	
IPC13	00BE	—	—	—	—	—	INT4IP<2:0>			—	INT3IP<2:0>			—	—	
IPC14	00C0	—	—	—	—	—	QE11IP<2:0>			—	PSEMIP<2:0>			—	—	
IPC16	00C4	—	—	—	—	—	U2EIP<2:0>			—	U1EIP<2:0>			—	—	
IPC17	00C6	—	—	—	—	—	C1TXIP<2:0>			—	—	—	—	—	—	
IPC18	00C8	—	QE12IP<2:0>			—	—	—	—	—	—	PSESMIP<2:0>			—	—

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

**TABLE 4-6: INTERRUPT CONTROLLER REGISTER MAP FOR dsPIC33FJ64GS606 DEVICES (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
IPC21	00CE	—	—	—	—	—	—	—	—	—	ADCP12IP<2:0>			—	—
IPC23	00D2	—	PWM2IP<2:0>			—	PWM1IP<2:0>			—	—	—	—	—	—
IPC24	00D4	—	PWM6IP<2:0>			—	PWM5IP<2:0>			—	PWM4IP<2:0>			—	—
IPC25	00D6	—	AC2IP<2:0>			—	—	—	—	—	—	—	—	—	—
IPC26	00D8	—	—	—	—	—	—	—	—	—	AC4IP<2:0>			—	—
IPC27	00DA	—	ADCP1IP<2:0>			—	ADCP0IP<2:0>			—	—	—	—	—	—
IPC28	00DC	—	ADCP5IP<2:0>			—	ADCP4IP<2:0>			—	ADCP3IP<2:0>			—	—
IPC29	00DE	—	—	—	—	—	—	—	—	—	ADCP7IP<2:0>			—	—
INTTREG	00E0	—	—	—	—	ILR<3:0>			—	VECNUM<6:0>					

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

**TABLE 4-7: INTERRUPT CONTROLLER REGISTER MAP FOR dsPIC33FJ32GS406 AND dsPIC33FJ64GS406 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
INTCON1	0080	NSTDIS	OVAERR	OVBERR	COVAERR	COVBERR	OVATE	OVBTE	COVTE	SFTACERR	DIV0ERR	—	MATHERR	ADDRERR	STKE
INTCON2	0082	ALTIVT	DISI	—	—	—	—	—	—	—	—	—	INT4EP	INT3EP	INT2EP
IFS0	0084	—	—	ADIF	U1TXIF	U1RXIF	SPI1IF	SPI1EIF	T3IF	T2IF	OC2IF	IC2IF	—	T1IF	OC1IF
IFS1	0086	U2TXIF	U2RXIF	INT2IF	T5IF	T4IF	OC4IF	OC3IF	—	—	—	—	INT1IF	CNIF	—
IFS2	0088	—	—	—	—	—	—	—	—	—	IC4IF	IC3IF	—	—	—
IFS3	008A	—	—	—	—	—	QE11IF	PSEMIF	—	—	INT4IF	INT3IF	—	—	MI2CIF
IFS4	008C	—	—	—	—	—	—	PSESMIF	—	—	—	—	—	—	U2EIF
IFS5	008E	PWM2IF	PWM1IF	ADCP12IF	—	—	—	—	—	—	—	—	—	—	—
IFS6	0090	ADCP11IF	ADCP0IF	—	—	—	—	—	—	—	—	—	—	PWM6IF	PWM5IF
IFS7	0092	—	—	—	—	—	—	—	—	—	—	ADCP7IF	ADCP6IF	ADCP5IF	ADCP4IF
IEC0	0094	—	—	ADIE	U1TXIE	U1RXIE	SPI1IE	SPI1EIE	T3IE	T2IE	OC2IE	IC2IE	—	T1IE	OC1IE
IEC1	0096	U2TXIE	U2RXIE	INT2IE	T5IE	T4IE	OC4IE	OC3IE	—	—	—	—	INT1IE	CNIE	—
IEC2	0098	—	—	—	—	—	—	—	—	—	IC4IE	IC3IE	—	—	—
IEC3	009A	—	—	—	—	—	QE11IE	PSEMIE	—	—	INT4IE	INT3IE	—	—	MI2CIE
IEC4	009C	—	—	—	—	—	—	PSESMIE	—	—	—	—	—	—	U2EIE
IEC5	009E	PWM2IE	PWM1IE	ADCP12IE	—	—	—	—	—	—	—	—	—	—	—
IEC6	00A0	ADCP11IE	ADCP0IE	—	—	—	—	—	—	—	—	—	—	PWM6IE	PWM5IE
IEC7	00A2	—	—	—	—	—	—	—	—	—	—	ADCP7IE	ADCP6IE	ADCP5IE	ADCP4IE
IPC0	00A4	—	—	T1IP<2:0>		—	—	OC1IP<2:0>		—	IC1IP<2:0>		—	—	—
IPC1	00A6	—	—	T2IP<2:0>		—	—	OC2IP<2:0>		—	IC2IP<2:0>		—	—	—
IPC2	00A8	—	—	U1RXIP<2:0>		—	—	SPI1IP<2:0>		—	SPI1EIP<2:0>		—	—	—
IPC3	00AA	—	—	—	—	—	—	—	—	—	ADIP<2:0>		—	—	—
IPC4	00AC	—	—	CNIP<2:0>		—	—	—	—	—	MI2C1IP<2:0>		—	—	—
IPC5	00AE	—	—	—	—	—	—	—	—	—	—	—	—	—	—
IPC6	00B0	—	—	T4IP<2:0>		—	—	OC4IP<2:0>		—	OC3IP<2:0>		—	—	—
IPC7	00B2	—	—	U2TXIP<2:0>		—	—	U2RXIP<2:0>		—	INT2IP<2:0>		—	—	—
IPC8	00B4	—	—	—	—	—	—	—	—	—	SPI2IP<2:0>		—	—	—
IPC9	00B6	—	—	—	—	—	—	IC4IP<2:0>		—	IC3IP<2:0>		—	—	—
IPC12	00BC	—	—	—	—	—	—	MI2C2IP<2:0>		—	SI2C2IP<2:0>		—	—	—
IPC13	00BE	—	—	—	—	—	—	INT4IP<2:0>		—	INT3IP<2:0>		—	—	—
IPC14	00C0	—	—	—	—	—	—	QE11IP<2:0>		—	PSEMIP<2:0>		—	—	—
IPC16	00C4	—	—	—	—	—	—	U2EIP<2:0>		—	U1EIP<2:0>		—	—	—
IPC18	00C8	—	—	—	—	—	—	—	—	—	PSESMIP<2:0>		—	—	—
IPC23	00D2	—	—	PWM2IP<2:0>		—	—	PWM1IP<2:0>		—	—	—	—	—	—

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

**TABLE 4-7: INTERRUPT CONTROLLER REGISTER MAP FOR dsPIC33FJ32GS406 AND dsPIC33FJ64GS406 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
IPC24	00D4	—	PWM6IP<2:0>			—	PWM5IP<2:0>			—	PWM4IP<2:0>			—	—
IPC27	00DA	—	ADCP1IP<2:0>			—	ADCP0IP<2:0>			—	—	—	—	—	—
IPC28	00DC	—	ADCP5IP<2:0>			—	ADCP4IP<2:0>			—	ADCP3IP<2:0>			—	—
IPC29	00DE	—	—	—	—	—	—	—	—	—	ADCP7IP<2:0>			—	—
INTTREG	00E0	—	—	—	—	ILR<3:0>			—	VECNUM<6:0>					

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

**TABLE 4-8: INTERRUPT CONTROLLER REGISTER MAP FOR dsPIC33FJ32GS610 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
INTCON1	0080	NSTDIS	OVAERR	OVBERR	COVAERR	COVBERR	OVATE	OVBTE	COVTE	SFTACERR	DIV0ERR	—	MATHERR	ADDRRERR	STKER
INTCON2	0082	ALTVT	DISI	—	—	—	—	—	—	—	—	—	INT4EP	INT3EP	INT2EP
IFS0	0084	—	—	ADIF	U1TXIF	U1RXIF	SPI1IF	SPI1EIF	T3IF	T2IF	OC2IF	IC2IF	—	T1IF	OC1IF
IFS1	0086	U2TXIF	U2RXIF	INT2IF	T5IF	T4IF	OC4IF	OC3IF	—	—	—	—	INT1IF	CNIF	AC1IF
IFS2	0088	—	—	—	—	—	—	—	—	—	IC4IF	IC3IF	—	—	—
IFS3	008A	—	—	—	—	—	QE11IF	PSEMIF	—	—	INT4IF	INT3IF	—	—	M12C2IF
IFS4	008C	—	—	—	—	QE12IF	—	PSESMIF	—	—	—	—	—	—	U2EIF
IFS5	008E	PWM2IF	PWM1IF	ADCP12IF	—	—	—	—	—	—	—	—	ADCP11IF	ADCP10IF	ADCP9IF
IFS6	0090	ADCP1IF	ADCP0IF	—	—	—	—	AC4IF	AC3IF	AC2IF	PWM9IF	PWM8IF	PWM7IF	PWM6IF	PWM5IF
IFS7	0092	—	—	—	—	—	—	—	—	—	—	ADCP7IF	ADCP6IF	ADCP5IF	ADCP4IF
IEC0	0094	—	—	ADIE	U1TXIE	U1RXIE	SPI1IE	SPI1EIE	T3IE	T2IE	OC2IE	IC2IE	—	T1IE	OC1IE
IEC1	0096	U2TXIE	U2RXIE	INT2IE	T5IE	T4IE	OC4IE	OC3IE	—	—	—	—	INT1IE	CNIE	AC1IE
IEC2	0098	—	—	—	—	—	—	—	—	—	IC4IE	IC3IE	—	—	—
IEC3	009A	—	—	—	—	—	QE11IE	PSEMIE	—	—	INT4IE	INT3IE	—	—	M12C2IF
IEC4	009C	—	—	—	—	QE12IE	—	PSESMIE	—	—	—	—	—	—	U2EIF
IEC5	009E	PWM2IE	PWM1IE	ADCP12IE	—	—	—	—	—	—	—	—	ADCP11IE	ADCP10IE	ADCP9IF
IEC6	00A0	ADCP1IE	ADCP0IE	—	—	—	—	AC4IE	AC3IE	AC2IE	PWM9IE	PWM8IE	PWM7IE	PWM6IE	PWM5IF
IEC7	00A2	—	—	—	—	—	—	—	—	—	—	ADCP7IE	ADCP6IE	ADCP5IE	ADCP4IF
IPC0	00A4	—	—	T1IP<2:0>		—	—	OC1IP<2:0>		—	—	IC1IP<2:0>		—	—
IPC1	00A6	—	—	T2IP<2:0>		—	—	OC2IP<2:0>		—	—	IC2IP<2:0>		—	—
IPC2	00A8	—	—	U1RXIP<2:0>		—	—	SPI1IP<2:0>		—	—	SPI1EIP<2:0>		—	—
IPC3	00AA	—	—	—	—	—	—	—	—	—	—	ADIP<2:0>		—	—
IPC4	00AC	—	—	CNIP<2:0>		—	—	AC1IP<2:0>		—	—	M12C1IP<2:0>		—	—
IPC5	00AE	—	—	—	—	—	—	—	—	—	—	—	—	—	—
IPC6	00B0	—	—	T4IP<2:0>		—	—	OC4IP<2:0>		—	—	OC3IP<2:0>		—	—
IPC7	00B2	—	—	U2TXIP<2:0>		—	—	U2RXIP<2:0>		—	—	INT2IP<2:0>		—	—
IPC8	00B4	—	—	—	—	—	—	—	—	—	—	SPI2IP<2:0>		—	—
IPC9	00B6	—	—	—	—	—	—	IC4IP<2:0>		—	—	IC3IP<2:0>		—	—
IPC12	00BC	—	—	—	—	—	—	M12C2IP<2:0>		—	—	S12C2IP<2:0>		—	—
IPC13	00BE	—	—	—	—	—	—	INT4IP<2:0>		—	—	INT3IP<2:0>		—	—
IPC14	00C0	—	—	—	—	—	—	QE11IP<2:0>		—	—	PSEMIP<2:0>		—	—
IPC16	00C4	—	—	—	—	—	—	U2EIP<2:0>		—	—	U1EIP<2:0>		—	—
IPC18	00C8	—	—	QE12IP<2:0>		—	—	—	—	—	—	PSESMIP<2:0>		—	—
IPC20	00CC	—	—	ADCP10IP<2:0>		—	—	ADCP9IP<2:0>		—	—	ADCP8IP<2:0>		—	—

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

**TABLE 4-8: INTERRUPT CONTROLLER REGISTER MAP FOR dsPIC33FJ32GS610 DEVICES (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
IPC21	00CE	—	—	—	—	—	—	—	—	—	ADCP12IP<2:0>			—	—
IPC23	00D2	—	PWM2IP<2:0>			—	PWM1IP<2:0>			—	—	—	—	—	—
IPC24	00D4	—	PWM6IP<2:0>			—	PWM5IP<2:0>			—	PWM4IP<2:0>			—	—
IPC25	00D6	—	AC2IP<2:0>			—	PWM9IP<2:0>			—	PWM8IP<2:0>			—	—
IPC26	00D8	—	—	—	—	—	—	—	—	—	AC4IP<2:0>			—	—
IPC27	00DA	—	ADCP1IP<2:0>			—	ADCP0IP<2:0>			—	—	—	—	—	—
IPC28	00DC	—	ADCP5IP<2:0>			—	ADCP4IP<2:0>			—	ADCP3IP<2:0>			—	—
IPC29	00DE	—	—	—	—	—	—	—	—	—	ADCP7IP<2:0>			—	—
INTTREG	00E0	—	—	—	—	ILR<3:0>			—	VECNUM<6:0>					

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



**TABLE 4-9: INTERRUPT CONTROLLER REGISTER MAP FOR dsPIC33FJ32GS608**

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
INTCON1	0080	NSTDIS	OVAERR	OVBERR	COVAERR	COVBERR	OVATE	OVBT	COVTE	SFTACERR	DIV0ERR	—	MATHERR	ADDRERR	STKERR
INTCON2	0082	ALTVT	DISI	—	—	—	—	—	—	—	—	—	INT4EP	INT3EP	INT2EP
IFS0	0084	—	—	ADIF	U1TXIF	U1RXIF	SPI1IF	SPI1EIF	T3IF	T2IF	OC2IF	IC2IF	—	T1IF	OC1IF
IFS1	0086	U2TXIF	U2RXIF	INT2IF	T5IF	T4IF	OC4IF	OC3IF	—	—	—	—	INT1IF	CNIF	AC1IF
IFS2	0088	—	—	—	—	—	—	—	—	—	IC4IF	IC3IF	—	—	—
IFS3	008A	—	—	—	—	—	QE11IF	PSEMIF	—	—	INT4IF	INT3IF	—	—	MI2C1IF
IFS4	008C	—	—	—	—	QE12IF	—	PSESMIF	—	—	—	—	—	—	U2E1IF
IFS5	008E	PWM2IF	PWM1IF	ADCP12IF	—	—	—	—	—	—	—	—	—	—	—
IFS6	0090	ADCP1IF	ADCP0IF	—	—	—	—	AC4IF	AC3IF	AC2IF	—	PWM8IF	PWM7IF	PWM6IF	PWM5IF
IFS7	0092	—	—	—	—	—	—	—	—	—	—	ADCP7IF	ADCP6IF	ADCP5IF	ADCP4IF
IEC0	0094	—	—	ADIE	U1TXIE	U1RXIE	SPI1IE	SPI1EIE	T3IE	T2IE	OC2IE	IC2IE	—	T1IE	OC1IE
IEC1	0096	U2TXIE	U2RXIE	INT2IE	T5IE	T4IE	OC4IE	OC3IE	—	—	—	—	INT1IE	CNIE	—
IEC2	0098	—	—	—	—	—	—	—	—	—	IC4IE	IC3IE	—	—	—
IEC3	009A	—	—	—	—	—	QE11IE	PSEMIE	—	—	INT4IE	INT3IE	—	—	MI2C1IE
IEC4	009C	—	—	—	—	QE12IE	—	PSESMIE	—	—	—	—	—	—	U2E1IE
IEC5	009E	PWM2IE	PWM1IE	ADCP12IE	—	—	—	—	—	—	—	—	—	—	—
IEC6	00A0	ADCP1IE	ADCP0IE	—	—	—	—	AC4IE	AC3IE	AC2IE	—	PWM8IE	PWM7IE	PWM6IE	PWM5IE
IEC7	00A2	—	—	—	—	—	—	—	—	—	—	ADCP7IE	ADCP6IE	ADCP5IE	ADCP4IE
IPC0	00A4	—	—	T1IP<2:0>			—	OC1IP<2:0>			—	IC1IP<2:0>			—
IPC1	00A6	—	—	T2IP<2:0>			—	OC2IP<2:0>			—	IC2IP<2:0>			—
IPC2	00A8	—	—	U1RXIP<2:0>			—	SPI1IP<2:0>			—	SPI1EIP<2:0>			—
IPC3	00AA	—	—	—	—	—	—	—	—	—	—	ADIP<2:0>			—
IPC4	00AC	—	—	CNIP<2:0>			—	AC1IP<2:0>			—	MI2C1IP<2:0>			—
IPC5	00AE	—	—	—	—	—	—	—	—	—	—	—	—	—	—
IPC6	00B0	—	—	T4IP<2:0>			—	OC4IP<2:0>			—	OC3IP<2:0>			—
IPC7	00B2	—	—	U2TXIP<2:0>			—	U2RXIP<2:0>			—	INT2IP<2:0>			—
IPC8	00B4	—	—	—	—	—	—	—	—	—	—	SPI2IP<2:0>			—
IPC9	00B6	—	—	—	—	—	—	IC4IP<2:0>			—	IC3IP<2:0>			—
IPC12	00BC	—	—	—	—	—	—	MI2C2IP<2:0>			—	SI2C2IP<2:0>			—
IPC13	00BE	—	—	—	—	—	—	INT4IP<2:0>			—	INT3IP<2:0>			—
IPC14	00C0	—	—	—	—	—	—	QE11IP<2:0>			—	PSEMIP<2:0>			—
IPC16	00C4	—	—	—	—	—	—	U2EIP<2:0>			—	U1EIP<2:0>			—
IPC18	00C8	—	—	QE12IP<2:0>			—	—	—	—	—	PSESMIP<2:0>			—
IPC20	00CC	—	—	—	—	—	—	—	—	—	—	ADCP8IP<2:0>			—

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

**TABLE 4-9: INTERRUPT CONTROLLER REGISTER MAP FOR dsPIC33FJ32GS608 (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
IPC21	00CE	—	—	—	—	—	—	—	—	—	ADCP12IP<2:0>			—	—
IPC23	00D2	—	PWM2IP<2:0>			—	PWM1IP<2:0>			—	—	—	—	—	—
IPC24	00D4	—	PWM6IP<2:0>			—	PWM5IP<2:0>			—	PWM4IP<2:0>			—	—
IPC25	00D6	—	AC2IP<2:0>			—	—	—	—	—	PWM8IP<2:0>			—	—
IPC26	00D8	—	—	—	—	—	—	—	—	—	AC4IP<2:0>			—	—
IPC27	00DA	—	ADCP1IP<2:0>			—	ADCP0IP<2:0>			—	—	—	—	—	—
IPC28	00DC	—	ADCP5IP<2:0>			—	ADCP4IP<2:0>			—	ADCP3IP<2:0>			—	—
IPC29	00DE	—	—	—	—	—	—	—	—	—	ADCP7IP<2:0>			—	—
INTTREG	00E0	—	—	—	—	ILR<3:0>			—	VECNUM<6:0>					

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

**TABLE 4-10: INTERRUPT CONTROLLER REGISTER MAP FOR dsPIC33FJ32GS606 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
INTCON1	0080	NSTDIS	OVAERR	OVBERR	COVAERR	COVBERR	OVATE	OVBTE	COVTE	SFTACERR	DIV0ERR	—	MATHERR	ADDRERR	STKE
INTCON2	0082	ALTVT	DISI	—	—	—	—	—	—	—	—	—	INT4EP	INT3EP	INT2EP
IFS0	0084	—	—	ADIF	U1TXIF	U1RXIF	SPI1IF	SPI1EIF	T3IF	T2IF	OC2IF	IC2IF	—	T1IF	OC1IF
IFS1	0086	U2TXIF	U2RXIF	INT2IF	T5IF	T4IF	OC4IF	OC3IF	—	—	—	—	INT1IF	CNIF	AC1IF
IFS2	0088	—	—	—	—	—	—	—	—	—	IC4IF	IC3IF	—	—	—
IFS3	008A	—	—	—	—	—	QE11IF	PSEMIF	—	—	INT4IF	INT3IF	—	—	MI2C1IF
IFS4	008C	—	—	—	—	QE12IF	—	PSESMIF	—	—	—	—	—	—	U2EIF
IFS5	008E	PWM2IF	PWM1IF	ADCP12IF	—	—	—	—	—	—	—	—	—	—	—
IFS6	0090	ADCP1IF	ADCP0IF	—	—	—	—	AC4IF	AC3IF	AC2IF	—	—	—	PWM6IF	PWM5IF
IFS7	0092	—	—	—	—	—	—	—	—	—	—	ADCP7IF	ADCP6IF	ADCP5IF	ADCP4IF
IEC0	0094	—	—	ADIE	U1TXIE	U1RXIE	SPI1IE	SPI1EIE	T3IE	T2IE	OC2IE	IC2IE	—	T1IE	OC1IE
IEC1	0096	U2TXIE	U2RXIE	INT2IE	T5IE	T4IE	OC4IE	OC3IE	—	—	—	—	INT1IE	CNIE	AC1IE
IEC2	0098	—	—	—	—	—	—	—	—	—	IC4IE	IC3IE	—	—	—
IEC3	009A	—	—	—	—	—	QE11IE	PSEMIE	—	—	INT4IE	INT3IE	—	—	MI2C1IE
IEC4	009C	—	—	—	—	QE12IE	—	PSESMIE	—	—	—	—	—	—	U2EIE
IEC5	009E	PWM2IE	PWM1IE	ADCP12IE	—	—	—	—	—	—	—	—	—	—	—
IEC6	00A0	ADCP1IE	ADCP0IE	—	—	—	—	AC4IE	AC3IE	AC2IE	—	—	—	PWM6IE	PWM5IE
IEC7	00A2	—	—	—	—	—	—	—	—	—	—	ADCP7IE	ADCP6IE	ADCP5IE	ADCP4IE
IPC0	00A4	—	—	T1IP<2:0>		—	—	OC1IP<2:0>		—	IC1IP<2:0>		—	—	—
IPC1	00A6	—	—	T2IP<2:0>		—	—	OC2IP<2:0>		—	IC2IP<2:0>		—	—	—
IPC2	00A8	—	—	U1RXIP<2:0>		—	—	SPI1IP<2:0>		—	SPI1EIP<2:0>		—	—	—
IPC3	00AA	—	—	—	—	—	—	—	—	—	ADIP<2:0>		—	—	—
IPC4	00AC	—	—	CNIP<2:0>		—	—	AC1IP<2:0>		—	MI2C1IP<2:0>		—	—	—
IPC5	00AE	—	—	—	—	—	—	—	—	—	—	—	—	—	—
IPC6	00B0	—	—	T4IP<2:0>		—	—	OC4IP<2:0>		—	OC3IP<2:0>		—	—	—
IPC7	00B2	—	—	U2TXIP<2:0>		—	—	U2RXIP<2:0>		—	INT2IP<2:0>		—	—	—
IPC8	00B4	—	—	—	—	—	—	—	—	—	SPI2IP<2:0>		—	—	—
IPC9	00B6	—	—	—	—	—	—	IC4IP<2:0>		—	IC3IP<2:0>		—	—	—
IPC12	00BC	—	—	—	—	—	—	MI2C2IP<2:0>		—	SI2C2IP<2:0>		—	—	—
IPC13	00BE	—	—	—	—	—	—	INT4IP<2:0>		—	INT3IP<2:0>		—	—	—
IPC14	00C0	—	—	—	—	—	—	QE11IP<2:0>		—	PSEMIP<2:0>		—	—	—
IPC16	00C4	—	—	—	—	—	—	U2EIP<2:0>		—	U1EIP<2:0>		—	—	—
IPC18	00C8	—	—	QE12IP<2:0>		—	—	—	—	—	PSESMIP<2:0>		—	—	—
IPC21	00CE	—	—	—	—	—	—	—	—	—	ADCP12IP<2:0>		—	—	—

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

**TABLE 4-10: INTERRUPT CONTROLLER REGISTER MAP FOR dsPIC33FJ32GS606 DEVICES (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	
IPC23	00D2	—	PWM2IP<2:0>			—	PWM1IP<2:0>			—	—	—	—	—	—	—
IPC24	00D4	—	PWM6IP<2:0>			—	PWM5IP<2:0>			—	PWM4IP<2:0>			—	—	
IPC25	00D6	—	AC2IP<2:0>			—	—	—	—	—	—	—	—	—	—	—
IPC26	00D8	—	—	—	—	—	—	—	—	—	AC4IP<2:0>			—	—	
IPC27	00DA	—	ADCP1IP<2:0>			—	ADCP0IP<2:0>			—	—	—	—	—	—	
IPC28	00DC	—	ADCP5IP<2:0>			—	ADCP4IP<2:0>			—	ADCP3IP<2:0>			—	—	
IPC29	00DE	—	—	—	—	—	—	—	—	—	ADCP7IP<2:0>			—	—	
INTTREG	00E0	—	—	—	—	ILR<3:0>			—	VECNUM<6:0>						

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

**TABLE 4-11: TIMERS 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
TMR1	0100	Timer1 Register													
PR1	0102	Period Register 1													
T1CON	0104	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>	—	TSYN	—
TMR2	0106	Timer2 Register													
TMR3HLD	0108	Timer3 Holding Register (for 32-bit timer operations only)													
TMR3	010A	Timer3 Register													
PR2	010C	Period Register 2													
PR3	010E	Period Register 3													
T2CON	0110	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>	T32	—	—
T3CON	0112	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>	—	—	—
TMR4	0114	Timer4 Register													
TMR5HLD	0116	Timer5 Holding Register (for 32-bit timer operations only)													
TMR5	0118	Timer5 Register													
PR4	011A	Period Register 4													
PR5	011C	Period Register 5													
T4CON	011E	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>	T32	—	—
T5CON	0120	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>	—	—	—

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

**TABLE 4-12: 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
IC1BUF	0140	Input 1 Capture Register													
IC1CON	0142	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>	ICOV	ICBNE	—	—
IC2BUF	0144	Input 2 Capture Register													
IC2CON	0146	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>	ICOV	ICBNE	—	—
IC3BUF	0148	Input 3 Capture Register													
IC3CON	014A	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>	ICOV	ICBNE	—	—
IC4BUF	014C	Input 4 Capture Register													
IC4CON	014E	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>	ICOV	ICBNE	—	—

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

**TABLE 4-13: OUTPUT COMPARE 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	
OC1RS	0180	Output Compare 1 Secondary Register														
OC1R	0182	Output Compare 1 Register														
OC1CON	0184	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	—	
OC2RS	0186	Output Compare 2 Secondary Register														
OC2R	0188	Output Compare 2 Register														
OC2CON	018A	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	—	
OC3RS	018C	Output Compare 3 Secondary Register														
OC3R	018E	Output Compare 3 Register														
OC3CON	0190	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	—	
OC4RS	0192	Output Compare 4 Secondary Register														
OC4R	0194	Output Compare 4 Register														
OC4CON	0196	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	—	

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

**TABLE 4-14: QE1 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
QE1CON	01E0	CNTERR	—	QEISIDL	INDX	UPDN	QEIM<2:0>			SWPAB	PCDOUT	TQGATE	TQCKPS<1:0>		POSRES
DFLT1CON	01E2	—	—	—	—	—	IMV<1:0>	CEID	QEOUT	QECK<2:0>				—	—
POS1CNT	01E4	Position Counter<15:0>													
MAX1CNT	01E6	Maximum Count<15:0>													

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

**TABLE 4-15: QE2 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
QE2CON	01F0	CNTERR	—	QEISIDL	INDX	UPDN	QEIM<2:0>			SWPAB	PCDOUT	TQGATE	TQCKPS<1:0>		POSRES
DFLT2CON	01F2	—	—	—	—	—	IMV<1:0>	CEID	QEOUT	QECK<2:0>				—	—
POS2CNT	01F4	Position Counter<15:0>													
MAX2CNT	01F6	Maximum Count<15:0>													

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

**TABLE 4-16: HIGH-SPEED PWM REGISTER MAP**

File Name	Addr Offset	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
PTCON	0400	PTEN	—	PTSIDL	SESTAT	SEIEN	EIPU	SYNCPOL	SYNCOEN	SYNCEN	SYNCSRC<2:0>		—	—	SE
PTCON2	0402	—	—	—	—	—	—	—	—	—	—	—	—	—	—
PTPER	0404	PTPER<15:0>													
SEVTCMP	0406	SEVTCMP<15:3>													
MDC	040A	MDC<15:0>													
STCON	040E	—	—	—	SESTAT	SEIEN	EIPU	SYNCPOL	SYNCOEN	SYNCEN	SYNCSRC<2:0>		—	—	SE
STCON2	0410	—	—	—	—	—	—	—	—	—	—	—	—	—	—
STPER	0412	PTPER<15:0>													
SSEVTCMP	0414	SSEVTCMP<15:3>													
CHOP	041A	CHPCLKEN	—	—	—	—	—	—	—	—	CHOP<9:3>		—	—	—

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

**TABLE 4-17: HIGH-SPEED PWM GENERATOR 1 REGISTER MAP**

File Name	Addr Offset	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
PWMCON1	0420	FLTSTAT	CLSTAT	TRGSTAT	FLTIEEN	CLIEEN	TRGIEEN	ITB	MDCS	DTC<1:0>		DTCP	—	MTBS	CAM
IOCON1	0422	PENH	PENL	POLH	POLL	PMOD<1:0>		OVRENH	OVRENL	OVRDAT<1:0>		FLTDAT<1:0>		CLDAT<1:0>	
FCLCON1	0424	IFLTMOD	CLSRC<4:0>				CLPOL	CLMOD	FLTSRC<4:0>				FLTPOL		
PDC1	0426	PDC1<15:0>													
PHASE1	0428	PHASE1<15:0>													
DTR1	042A	—	—	DTR1<13:0>											
ALTDTR1	042C	—	—	ALTDTR1<13:0>											
SDC1	042E	SDC1<15:0>													
SPHASE1	0430	SPHASE1<15:0>													
TRIG1	0432	TRGCMP<15:3>													
TRGCON1	0434	TRGDIV<3:0>				—	—	—	—	DTM	—	TRGSTRT<5:0>			
STRIG1	0436	STRGCMP<15:3>													
PWMCAP1	0438	PWMCAP1<15:3>													
LEBCON1	043A	PHR	PHF	PLR	PLF	FLTLEBEN	CLLEBEN	—	—	—	—	BCH	BCL	BPHH	BPHL
LEBDLY1	043C	—	—	—	—	LEB<11:3>									—
AUXCON1	043E	HRPDIS	HRDDIS	—	—	BLANKSEL<3:0>				—	—	CHOPSEL<3:0>			

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

**TABLE 4-18: HIGH-SPEED PWM GENERATOR 2 REGISTER MAP**

File Name	Addr Offset	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	
PWMCON2	0440	FLTSTAT	CLSTAT	TRGSTAT	FLTIEN	CLIEN	TRGIEN	ITB	MDCS	DTC<1:0>		DTCP	—	MTBS	CAM	
IOCON2	0442	PENH	PENL	POLH	POLL	PMOD<1:0>		OVRENH	OVRENL	OVRDAT<1:0>		FLTDAT<1:0>		CLDAT<1:0>		
FCLCON2	0444	IFLTMOD	CLSRC<4:0>					CLPOL	CLMOD	FLTSRC<4:0>					FLTPOL	
PDC2	0446	PDC2<15:0>														
PHASE2	0448	PHASE2<15:0>														
DTR2	044A	—	—	DTR2<13:0>												
ALTDTR2	044C	—	—	ALTDTR2<13:0>												
SDC2	044E	SDC2<15:0>														
SPHASE2	0450	SPHASE2<15:0>														
TRIG2	0452	TRGCMP<15:3>														
TRGCON2	0454	TRGDIV<3:0>				—	—	—	—	DTM	—	TRGSTRT<5:0>				
STRIG2	0456	STRGCMP<15:3>														
PWMCAP2	0458	PWMCAP2<15:3>														
LEBCON2	045A	PHR	PHF	PLR	PLF	FLTLEBEN	CLLEBEN	—	—	—	—	BCH	BCL	BPHH	BPHL	
LEBDLY2	045C	—	—	—	—	LEB<11:3>									—	
AUXCON2	045E	HRPDIS	HRDDIS	—	—	BLANKSEL<3:0>				—	—	CHOPSEL<3:0>				

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



**TABLE 4-19: HIGH-SPEED PWM GENERATOR 3 REGISTER MAP**

File Name	Addr Offset	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	
PWMCON3	0460	FLTSTAT	CLSTAT	TRGSTAT	FLTIEN	CLIEEN	TRGIEEN	ITB	MDCS	DTC<1:0>		DTCP	—	MTBS	CAM	
IOCON3	0462	PENH	PENL	POLH	POLL	PMOD<1:0>		OVRENH	OVRENL	OVRDAT<1:0>	FLTDAT<1:0>		CLDAT<1:0>			
FCLCON3	0464	IFLTMOD	CLSRC<4:0>					CLPOL	CLMOD	FLTSRC<4:0>				FLTPOL		
PDC3	0466	PDC3<15:0>														
PHASE3	0468	PHASE3<15:0>														
DTR3	046C	—	—	DTR3<13:0>												
ALTDTR3	046C	—	—	ALTDTR3<13:0>												
SDC3	046E	SDC3<15:0>														
SPHASE3	0470	SPHASE3<15:0>														
TRIG3	0472	TRGCMP<15:3>														
TRGCON3	0474	TRGDIV<3:0>				—	—	—	—	DTM	—	TRGSTRT<5:0>				
STRIG3	0476	STRGCMP<15:3>														
PWMCAP3	0478	PWMCAP3<15:3>														
LEBCON3	047A	PHR	PHF	PLR	PLF	FLTLEBEN	CLLEBEN	—	—	—	—	BCH	BCL	BPBH	BPHL	
LEBDLY3	047C	—	—	—	—	LEB<11:3>									—	
AUXCON3	047E	HRPDIS	HRDDIS	—	—	BLANKSEL<3:0>				—	—	CHOPSEL<3:0>				

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

**TABLE 4-20: HIGH-SPEED PWM GENERATOR 4 REGISTER MAP**

File Name	Addr Offset	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	
PWMCON4	0480	FLTSTAT	CLSTAT	TRGSTAT	FLTIEN	CLIEN	TRGIEN	ITB	MDCS	DTC<1:0>		DTCP	—	MTBS	CAM	
IOCON4	0482	PENH	PENL	POLH	POLL	PMOD<1:0>		OVRENH	OVRENL	OVRDAT<1:0>		FLTDAT<1:0>		CLDAT<1:0>		
FCLCON4	0484	IFLTMOD	CLSRC<4:0>					CLPOL	CLMOD	FLTSRC<4:0>					FLTPOL	
PDC4	0486	PDC4<15:0>														
PHASE4	0488	PHASE4<15:0>														
DTR4	048A	—	—	DTR4<13:0>												
ALTDTR4	048A	—	—	ALTDTR4<13:0>												
SDC4	048E	SDC4<15:0>														
SPHASE4	0490	SPHASE4<15:0>														
TRIG4	0492	TRGCMP<15:3>														
TRGCON4	0494	TRGDIV<3:0>				—	—	—	—	DTM	—	TRGSTRT<5:0>				
STRIG4	0496	STRGCMP<15:3>														
PWMCAP4	0498	PWMCAP4<15:3>														
LEBCON4	049A	PHR	PHF	PLR	PLF	FLTLEBEN	CLLEBEN	—	—	—	—	BCH	BCL	BPHH	BPHL	
LEBDLY4	049C	—	—	—	—	LEB<11:3>									—	
AUXCON4	049E	HRPDIS	HRDDIS	—	—	BLANKSEL<3:0>				—	—	CHOPSEL<3:0>				

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

**TABLE 4-21: HIGH-SPEED PWM GENERATOR 5 REGISTER MAP**

File Name	Addr Offset	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	
PWMCON5	04A0	FLTSTAT	CLSTAT	TRGSTAT	FLTIEN	CLIEEN	TRGIEEN	ITB	MDCS	DTC<1:0>		DTCP	—	MTBS	CAM	
IOCON5	04A2	PENH	PENL	POLH	POLL	PMOD<1:0>		OVRENH	OVRENL	OVRDAT<1:0>		FLTDAT<1:0>		CLDAT<1:0>		
FCLCON5	04A4	IFLTMOD	CLSRC<4:0>				CLPOL	CLMOD	FLTSRC<4:0>				FLTPOL			
PDC5	04A6	PDC5<15:0>														
PHASE5	04A8	PHASE5<15:0>														
DTR5	04AA	—	—	DTR5<13:0>												
ALTDTR5	04AA	—	—	ALTDTR5<13:0>												
SDC5	04AE	SDC5<15:0>														
SPHASE5	04B0	SPHASE5<15:0>														
TRIG5	04B2	TRGCMP<15:3>														
TRGCON5	04B4	TRGDIV<3:0>				—	—	—	—	DTM	—	TRGSTRT<5:0>				
STRIG5	04B6	STRGCMP<15:3>														
PWMCAP5	04B8	PWMCAP5<15:3>														
LEBCON5	04BA	PHR	PHF	PLR	PLF	FLTLEBEN	CLLEBEN	—	—	—	—	BCH	BCL	BPHH	BPHL	
LEBDLY5	04BC	—	—	—	—	LEB<11:3>									—	
AUXCON5	04BE	HRPDIS	HRDDIS	—	—	BLANKSEL<3:0>				—	—	CHOPSEL<3:0>				

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

**TABLE 4-22: HIGH-SPEED PWM GENERATOR 6 REGISTER MAP**

File Name	Addr Offset	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	
PWMCON6	04C0	FLTSTAT	CLSTAT	TRGSTAT	FLTIEN	CLIEEN	TRGIEEN	ITB	MDCS	DTC<1:0>		DTCP	—	MTBS	CAM	
IOCON6	04C2	PENH	PENL	POLH	POLL	PMOD<1:0>		OVRENH	OVRENL	OVRDAT<1:0>		FLTDAT<1:0>		CLDAT<1:0>		
FCLCON6	04C4	IFLTMOD	CLSRC<4:0>				CLPOL	CLMOD	FLTSRC<4:0>				FLTPOL			
PDC6	04C6	PDC6<15:0>														
PHASE6	04C8	PHASE6<15:0>														
DTR6	04CA	—	—	DTR6<13:0>												
ALTDTR6	04CA	—	—	ALTDTR6<13:0>												
SDC6	04CE	SDC6<15:0>														
SPHASE6	04D0	SPHASE6<15:0>														
TRIG6	04D2	TRGCMP<15:3>														
TRGCON6	04D4	TRGDIV<3:0>				—	—	—	—	DTM	—	TRGSTRT<5:0>				
STRIG6	04D6	STRGCMP<15:3>														
PWMCAP6	04D8	PWMCAP6<15:3>														
LEBCON6	04DA	PHR	PHF	PLR	PLF	FLTLEBEN	CLLEBEN	—	—	—	—	BCH	BCL	BPHH	BPHL	
LEBDLY6	04DC	—	—	—	—	LEB<11:3>									—	
AUXCON6	04DE	HRPDIS	HRDDIS	—	—	BLANKSEL<3:0>				—	—	CHOPSEL<3:0>				

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

**TABLE 4-23: HIGH-SPEED PWM GENERATOR 7 REGISTER MAP (EXCLUDES dsPIC33FJ32GS406 AND dsPIC33FJ32GS406)**

File Name	Addr Offset	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	
PWMCON7	04E0	FLTSTAT	CLSTAT	TRGSTAT	FLTIEN	CLIEEN	TRGIEEN	ITB	MDCS	DTC<1:0>		DTCP	—	MTBS	CAM	
IOCON7	04E2	PENH	PENL	POLH	POLL	PMOD<1:0>		OVRENH	OVRENL	OVRDAT<1:0>		FLTDAT<1:0>		CLDAT<1:0>		
FCLCON7	04E4	IFLTMOD	CLSRC<4:0>				CLPOL	CLMOD	FLTSRC<4:0>				FLTPOL			
PDC7	04E6	PDC7<15:0>														
PHASE7	04E8	PHASE7<15:0>														
DTR7	04EA	—	—	DTR7<13:0>												
ALTDTR7	04EA	—	—	ALTDTR7<13:0>												
SDC7	04EE	SDC7<15:0>														
SPHASE7	04F0	SPHASE7<15:0>														
TRIG7	04F2	TRGCMP<15:3>														
TRGCON7	04F4	TRGDIV<3:0>				—	—	—	—	DTM	—	TRGSTRT<5:0>				
STRIG7	04F6	STRGCMP<15:3>														
PWMCAP7	04F8	PWMCAP7<15:3>														
LEBCON7	04FA	PHR	PHF	PLR	PLF	FLTLEBEN	CLLEBEN	—	—	—	—	BCH	BCL	BPHH	BPHL	
LEBDLY7	04FC	—	—	—	—	LEB<11:3>									—	
AUXCON7	04FE	HRPDIS	HRDDIS	—	—	BLANKSEL<3:0>				—	—	CHOPSEL<3:0>				

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

**TABLE 4-24: HIGH-SPEED PWM GENERATOR 8 REGISTER MAP (EXCLUDES dsPIC33FJ32GS406 AND dsPIC33FJ)**

File Name	Addr Offset	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	
PWMCON8	0500	FLTSTAT	CLSTAT	TRGSTAT	FLTIEN	CLIEEN	TRGIEEN	ITB	MDCS	DTC<1:0>		DTCP	—	MTBS	CAM	
IOCON8	0502	PENH	PENL	POLH	POLL	PMOD<1:0>		OVRENH	OVRENL	OVRDAT<1:0>		FLTDAT<1:0>		CLDAT<1:0>		
FCLCON8	0504	IFLTMOD	CLSRC<4:0>				CLPOL	CLMOD	FLTSRC<4:0>				FLTPOL			
PDC8	0506	PDC8<15:0>														
PHASE8	0508	PHASE8<15:0>														
DTR8	050A	—	—	DTR8<13:0>												
ALTDTR8	050A	—	—	ALTDTR8<13:0>												
SDC8	050E	SDC8<15:0>														
SPHASE8	0510	SPHASE8<15:0>														
TRIG8	0512	TRGCMP<15:3>														
TRGCON8	0514	TRGDIV<3:0>				—	—	—	—	DTM	—	TRGSTRT<5:0>				
STRIG8	0516	STRGCMP<15:3>														
PWMCAP8	0518	PWMCAP8<15:3>														
LEBCON8	051A	PHR	PHF	PLR	PLF	FLTLEBEN	CLLEBEN	—	—	—	—	BCH	BCL	BPHH	BPHL	
LEBDLY8	051C	—	—	—	—	LEB<11:3>									—	
AUXCON8	051E	HRPDIS	HRDDIS	—	—	BLANKSEL<3:0>				—	—	CHOPSEL<3:0>				

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

**TABLE 4-25: HIGH-SPEED PWM GENERATOR 9 REGISTER MAP FOR dsPIC33FJ32GS610 AND dsPIC33FJ64GS610**

File Name	Addr Offset	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	
PWMCON9	0520	FLTSTAT	CLSTAT	TRGSTAT	FLTIEN	CLIEEN	TRGIEEN	ITB	MDCS	DTC<1:0>		DTCP	—	MTBS	CAM	
IOCON9	0522	PENH	PENL	POLH	POLL	PMOD<1:0>		OVRENH	OVRENL	OVRDAT<1:0>		FLTDAT<1:0>		CLDAT<1:0>		
FCLCON9	0524	IFLTMOD	CLSRC<4:0>				CLPOL	CLMOD	FLTSRC<4:0>				FLTPOL			
PDC9	0526	PDC9<15:0>														
PHASE9	0528	PHASE9<15:0>														
DTR9	052A	—	—	DTR9<13:0>												
ALTDTR9	052A	—	—	ALTDTR9<13:0>												
SDC9	052E	SDC9<15:0>														
SPHASE9	0530	SPHASE9<15:0>														
TRIG9	0532	TRGCMP<15:3>														
TRGCON9	0534	TRGDIV<3:0>				—	—	—	—	DTM	—	TRGSTRT<5:0>				
STRIG9	0536	STRGCMP<15:3>														
PWMCAP9	0538	PWMCAP9<15:3>														
LEBCON9	053A	PHR	PHF	PLR	PLF	FLTLEBEN	CLLEBEN	—	—	—	—	BCH	BCL	BPHH	BPHL	
LEBDLY9	053C	—	—	—	—	LEB<11:3>									—	
AUXCON9	053E	HRPDIS	HRDDIS	—	—	BLANKSEL<3:0>				—	—	CHOPSEL<3:0>				

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

**TABLE 4-26: 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
I2C1RCV	0200	—	—	—	—	—	—	—	—	Receive Register					
I2C1TRN	0202	—	—	—	—	—	—	—	—	Transmit Register					
I2C1BRG	0204	—	—	—	—	—	—	—	Baud Rate Generator Register						
I2C1CON	0206	I2CEN	—	I2CSIDL	SCLREL	IPMIEN	A10M	DISSLW	SMEN	GCEN	STREN	ACKDT	ACKEN	RCEN	PEN
I2C1STAT	0208	ACKSTAT	TRSTAT	—	—	—	BCL	GCSTAT	ADD10	IWCOL	I2COV	D_A	P	S	R_W
I2C1ADD	020A	—	—	—	—	—	—	Address Register							
I2C1MSK	020C	—	—	—	—	—	—	Address Mask Register							

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

**TABLE 4-27: 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
I2C2RCV	0210	—	—	—	—	—	—	—	—	Receive Register					
I2C2TRN	0212	—	—	—	—	—	—	—	—	Transmit Register					
I2C2BRG	0214	—	—	—	—	—	—	—	Baud Rate Generator Register						
I2C2CON	0216	I2CEN	—	I2CSIDL	SCLREL	IPMIEN	A10M	DISSLW	SMEN	GCEN	STREN	ACKDT	ACKEN	RCEN	PEN
I2C2STAT	0218	ACKSTAT	TRSTAT	—	—	—	BCL	GCSTAT	ADD10	IWCOL	I2COV	D_A	P	S	R_W
I2C2ADD	021A	—	—	—	—	—	—	Address Register							
I2C2MSK	021C	—	—	—	—	—	—	Address Mask Register							

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

**TABLE 4-28: 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
U1MODE	0220	UARTEN	—	USIDL	IREN	RTSMD	—	UEN1	UEN0	WAKE	LPBACK	ABAUD	URXINV	BRGH	PDS
U1STA	0222	UTXISEL1	UTXINV	UTXISEL0	—	UTXBRK	UTXEN	UTXBF	TRMT	URXISEL<1:0>		ADDEN	RIDLE	PERR	FERR
U1TXREG	0224	—	—	—	—	—	—	—	UART Transmit Register						
U1RXREG	0226	—	—	—	—	—	—	—	UART Receive Register						
U1BRG	0228	Baud Rate Generator Prescaler													

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

**TABLE 4-29: 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
U2MODE	0230	UARTEN	—	USIDL	IREN	RTSMD	—	UEN1	UEN0	WAKE	LPBACK	ABAUD	URXINV	BRGH	PDS
U2STA	0232	UTXISEL1	UTXINV	UTXISEL0	—	UTXBRK	UTXEN	UTXBF	TRMT	URXISEL<1:0>		ADDEN	RIDLE	PERR	FERR
U2TXREG	0234	—	—	—	—	—	—	—	UART Transmit Register						
U2RXREG	0236	—	—	—	—	—	—	—	UART Receive Register						
U2BRG	0238	Baud Rate Generator Prescaler													

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



**TABLE 4-30: 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
SPI1STAT	0240	SPIEN	—	SPISIDL	—	—	—	—	—	—	SPIROV	—	—	—	—
SPI1CON1	0242	—	—	—	DISSCK	DISSDO	MODE16	SMP	CKE	SSEN	CKP	MSTEN	SPRE<2:0>		
SPI1CON2	0244	FRMEN	SPIFSD	FRMPOL	—	—	—	—	—	—	—	—	—	—	—
SPI1BUF	0248	SPI1 Transmit and Receive Buffer Register													

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

**TABLE 4-31: 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
SPI2STAT	0260	SPIEN	—	SPISIDL	—	—	—	—	—	—	SPIROV	—	—	—	—
SPI2CON1	0262	—	—	—	DISSCK	DISSDO	MODE16	SMP	CKE	SSEN	CKP	MSTEN	SPRE<2:0>		
SPI2CON2	0264	FRMEN	SPIFSD	FRMPOL	—	—	—	—	—	—	—	—	—	—	—
SPI2BUF	0268	SPI2 Transmit and Receive Buffer Register													

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

**TABLE 4-32: HIGH-SPEED 10-BIT ADC REGISTER MAP FOR dsPIC33FJ32GS610 AND dsPIC33FJ64GS610 DEVICE**

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
ADCON	0300	ADON	—	ADSIDL	SLOWCLK	—	GSWTRG	—	FORM	EIE	ORDER	SEQSAMP	ASYNCSAMP	—	—
ADPCFG	0302	PCFG15	PCFG14	PCFG13	PCFG12	PCFG11	PCFG10	PCFG9	PCFG8	PCFG7	PCFG6	PCFG5	PCFG4	PCFG3	PCFG2
ADPCFG2	0304	—	—	—	—	—	—	—	—	PCFG23	PCFG22	PCFG21	PCFG20	PCFG19	PCFG18
ADSTAT	0306	—	—	—	P12RDY	P11RDY	P10RDY	P9RDY	P8RDY	P7RDY	P6RDY	P5RDY	P4RDY	P3RDY	P2RDY
ADBASE	0308	ADBASE<15:1>													
ADCPC0	030A	IRQEN1	PEND1	SWTRG1	TRGSRC1<4:0>				IRQEN0	PEND0	SWTRG0	TRGSRC0<4:0>			
ADCPC1	030C	IRQEN3	PEND3	SWTRG3	TRGSRC3<4:0>				IRQEN2	PEND2	SWTRG2	TRGSRC2<4:0>			
ADCPC2	030E	IRQEN5	PEND5	SWTRG5	TRGSRC5<4:0>				IRQEN4	PEND4	SWTRG4	TRGSRC4<4:0>			
ADCPC3	0310	IRQEN7	PEND7	SWTRG7	TRGSRC7<4:0>				IRQEN6	PEND6	SWTRG6	TRGSRC6<4:0>			
ADCPC4	0312	IRQEN9	PEND9	SWTRG9	TRGSRC9<4:0>				IRQEN8	PEND8	SWTRG8	TRGSRC8<4:0>			
ADCPC5	0314	IRQEN11	PEND11	SWTRG11	TRGSRC11<4:0>				IRQEN10	PEND10	SWTRG10	TRGSRC10<4:0>			
ADCPC6	0316	—	—	—	—	—	—	—	—	IRQEN12	PEND12	SWTRG12	TRGSRC12<4:0>		
ADCBUF0	0340	ADC Data Buffer 0													
ADCBUF1	0342	ADC Data Buffer 1													
ADCBUF2	0344	ADC Data Buffer 2													
ADCBUF3	0346	ADC Data Buffer 3													
ADCBUF4	0348	ADC Data Buffer 4													
ADCBUF5	034A	ADC Data Buffer 5													
ADCBUF6	034C	ADC Data Buffer 6													
ADCBUF7	034E	ADC Data Buffer 7													
ADCBUF8	0350	ADC Data Buffer 8													
ADCBUF9	0352	ADC Data Buffer 9													
ADCBUF10	0354	ADC Data Buffer 10													
ADCBUF11	0356	ADC Data Buffer 11													
ADCBUF12	0358	ADC Data Buffer 12													
ADCBUF13	035A	ADC Data Buffer 13													
ADCBUF14	035C	ADC Data Buffer 14													
ADCBUF15	035E	ADC Data Buffer 15													
ADCBUF16	0360	ADC Data Buffer 16													
ADCBUF17	0362	ADC Data Buffer 17													
ADCBUF18	0364	ADC Data Buffer 18													
ADCBUF19	0366	ADC Data Buffer 19													
ADCBUF20	0368	ADC Data Buffer 20													
ADCBUF21	036A	ADC Data Buffer 21													

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

**TABLE 4-32: HIGH-SPEED 10-BIT ADC REGISTER MAP FOR dsPIC33FJ32GS610 AND dsPIC33FJ64GS610 DEVICE**

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
ADCBUF22	036C	ADC Data Buffer 22															
ADCBUF23	036E	ADC Data Buffer 23															
ADCBUF24	0370	ADC Data Buffer 24															
ADCBUF25	0372	ADC Data Buffer 25															

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

**TABLE 4-33: HIGH-SPEED 10-BIT ADC REGISTER MAP FOR dsPIC33FJ32GS608 AND dsPIC33FJ64GS608 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
ADCON	0300	ADON	—	ADSIDL	SLOWCLK	—	GSWTRG	—	FORM	EIE	ORDER	SEQSAMP	ASYNCSAMP	—	—
ADPCFG	0302	PCFG15	PCFG14	PCFG13	PCFG12	PCFG11	PCFG10	PCFG9	PCFG8	PCFG7	PCFG6	PCFG5	PCFG4	PCFG3	PCFG2
ADPCFG2	0304	—	—	—	—	—	—	—	—	—	—	—	—	—	—
ADSTAT	0306	—	—	—	P12RDY	—	—	—	P8RDY	P7RDY	P6RDY	P5RDY	P4RDY	P3RDY	P2RDY
ADBASE	0308	ADBASE<15:1>													
ADCPC0	030A	IRQEN1	PEND1	SWTRG1	TRGSRC1<4:0>				IRQEN0	PEND0	SWTRG0	TRGSRC0			
ADCPC1	030C	IRQEN3	PEND3	SWTRG3	TRGSRC3<4:0>				IRQEN2	PEND2	SWTRG2	TRGSRC2			
ADCPC2	030E	IRQEN5	PEND5	SWTRG5	TRGSRC5<4:0>				IRQEN4	PEND4	SWTRG4	TRGSRC4			
ADCPC3	0310	IRQEN7	PEND7	SWTRG7	TRGSRC7<4:0>				IRQEN6	PEND6	SWTRG6	TRGSRC6			
ADCPC4	0312	—	—	—	—	—	—	—	—	IRQEN8	PEND8	SWTRG8	TRGSRC8		
ADCPC6	0316	—	—	—	—	—	—	—	—	IRQEN12	PEND12	SWTRG12	TRGSRC12		
ADCBUF0	0340	ADC Data Buffer 0													
ADCBUF1	0342	ADC Data Buffer 1													
ADCBUF2	0344	ADC Data Buffer 2													
ADCBUF3	0346	ADC Data Buffer 3													
ADCBUF4	0348	ADC Data Buffer 4													
ADCBUF5	034A	ADC Data Buffer 5													
ADCBUF6	034C	ADC Data Buffer 6													
ADCBUF7	034E	ADC Data Buffer 7													
ADCBUF8	0350	ADC Data Buffer 8													
ADCBUF9	0352	ADC Data Buffer 9													
ADCBUF10	0354	ADC Data Buffer 10													
ADCBUF11	0356	ADC Data Buffer 11													
ADCBUF12	0358	ADC Data Buffer 12													
ADCBUF13	035A	ADC Data Buffer 13													
ADCBUF14	035C	ADC Data Buffer 14													
ADCBUF15	035E	ADC Data Buffer 15													
ADCBUF16	0360	ADC Data Buffer 16													
ADCBUF17	0362	ADC Data Buffer 17													
ADCBUF24	0370	ADC Data Buffer 24													
ADCBUF25	0372	ADC Data Buffer 25													

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

**TABLE 4-34: HIGH-SPEED 10-BIT ADC REGISTER MAP FOR dsPIC33FJ32GS606 AND dsPIC33FJ64GS606 DEVICE**

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
ADCON	0300	ADON	—	ADSIDL	SLOWCLK	—	GSWTRG	—	FORM	EIE	ORDER	SEQSAMP	ASYNCSAMP	—	—
ADPCFG	0302	PCFG15	PCFG14	PCFG13	PCFG12	PCFG11	PCFG10	PCFG9	PCFG8	PCFG7	PCFG6	PCFG5	PCFG4	PCFG3	PCFG2
ADSTAT	0306	—	—	—	P12RDY	—	—	—	—	P7RDY	P6RDY	P5RDY	P4RDY	P3RDY	P2RDY
ADBASE	0308	ADBASE<15:1>													
ADCPC0	030A	IRQEN1	PEND1	SWTRG1	TRGSRC1<4:0>				IRQEN0	PEND0	SWTRG0	TRGSRC0<4:0>			
ADCPC1	030C	IRQEN3	PEND3	SWTRG3	TRGSRC3<4:0>				IRQEN2	PEND2	SWTRG2	TRGSRC2<4:0>			
ADCPC2	030E	IRQEN5	PEND5	SWTRG5	TRGSRC5<4:0>				IRQEN4	PEND4	SWTRG4	TRGSRC4<4:0>			
ADCPC3	0310	IRQEN7	PEND7	SWTRG7	TRGSRC7<4:0>				IRQEN6	PEND6	SWTRG6	TRGSRC6<4:0>			
ADCPC6	0316	—	—	—	—	—	—	—	—	IRQEN12	PEND12	SWTRG12	TRGSRC12<4:0>		
ADCBUF0	0340	ADC Data Buffer 0													
ADCBUF1	0342	ADC Data Buffer 1													
ADCBUF2	0344	ADC Data Buffer 2													
ADCBUF3	0346	ADC Data Buffer 3													
ADCBUF4	0348	ADC Data Buffer 4													
ADCBUF5	034A	ADC Data Buffer 5													
ADCBUF6	034C	ADC Data Buffer 6													
ADCBUF7	034E	ADC Data Buffer 7													
ADCBUF8	0350	ADC Data Buffer 8													
ADCBUF9	0352	ADC Data Buffer 9													
ADCBUF10	0354	ADC Data Buffer 10													
ADCBUF11	0356	ADC Data Buffer 11													
ADCBUF12	0358	ADC Data Buffer 12													
ADCBUF13	035A	ADC Data Buffer 13													
ADCBUF14	035C	ADC Data Buffer 14													
ADCBUF15	035E	ADC Data Buffer 15													
ADCBUF24	0370	ADC Data Buffer 24													
ADCBUF25	0372	ADC Data Buffer 25													

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

**TABLE 4-35: HIGH-SPEED 10-BIT ADC REGISTER MAP FOR dsPIC33FJ32GS406 AND dsPIC33FJ64GS406 DEVICE**

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
ADCON	0300	ADON	—	ADSIDL	SLOWCLK	—	GSWTRG	—	FORM	EIE	ORDER	SEQSAMP	ASYNCAMP	—	—
ADPCFG	0302	PCFG15	PCFG14	PCFG13	PCFG12	PCFG11	PCFG10	PCFG9	PCFG8	PCFG7	PCFG6	PCFG5	PCFG4	PCFG3	PCFG2
ADSTAT	0306	—	—	—	P12RDY	—	—	—	—	P7RDY	P6RDY	P5RDY	P4RDY	P3RDY	P2RDY
ADBASE	0308	ADBASE<15:1>													
ADCPC0	030A	IRQEN1	PEND1	SWTRG1	TRGSRC1<4:0>				IRQEN0	PEND0	SWTRG0	TRGSRC0<4:0>			
ADCPC1	030C	IRQEN3	PEND3	SWTRG3	TRGSRC3<4:0>				IRQEN2	PEND2	SWTRG2	TRGSRC2<4:0>			
ADCPC2	030E	IRQEN5	PEND5	SWTRG5	TRGSRC5<4:0>				IRQEN4	PEND4	SWTRG4	TRGSRC4<4:0>			
ADCPC3	0310	IRQEN7	PEND7	SWTRG7	TRGSRC7<4:0>				IRQEN6	PEND6	SWTRG6	TRGSRC6<4:0>			
ADCBUF0	0340	ADC Data Buffer 0													
ADCBUF1	0342	ADC Data Buffer 1													
ADCBUF2	0344	ADC Data Buffer 2													
ADCBUF3	0346	ADC Data Buffer 3													
ADCBUF4	0348	ADC Data Buffer 4													
ADCBUF5	034A	ADC Data Buffer 5													
ADCBUF6	034C	ADC Data Buffer 6													
ADCBUF7	034E	ADC Data Buffer 7													
ADCBUF8	0350	ADC Data Buffer 8													
ADCBUF9	0352	ADC Data Buffer 9													
ADCBUF10	0354	ADC Data Buffer 10													
ADCBUF11	0356	ADC Data Buffer 11													
ADCBUF12	0358	ADC Data Buffer 12													
ADCBUF13	035A	ADC Data Buffer 13													
ADCBUF14	035C	ADC Data Buffer 14													
ADCBUF15	035E	ADC Data Buffer 15													

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

**TABLE 4-36: 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
DMA0CON	0380	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—
DMA0REQ	0382	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>				
DMA0STA	0384	STA<15:0>													
DMA0STB	0386	STB<15:0>													
DMA0PAD	0388	PAD<15:0>													
DMA0CNT	038A	—	—	—	—	—	—	CNT<9:0>							
DMA1CON	038C	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—
DMA1REQ	038E	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>				
DMA1STA	0390	STA<15:0>													
DMA1STB	0392	STB<15:0>													
DMA1PAD	0394	PAD<15:0>													
DMA1CNT	0396	—	—	—	—	—	—	CNT<9:0>							
DMA2CON	0398	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—
DMA2REQ	039A	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>				
DMA2STA	039C	STA<15:0>													
DMA2STB	039E	STB<15:0>													
DMA2PAD	03A0	PAD<15:0>													
DMA2CNT	03A2	—	—	—	—	—	—	CNT<9:0>							
DMA3CON	03A4	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—
DMA3REQ	03A6	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>				
DMA3STA	03A8	STA<15:0>													
DMA3STB	03AA	STB<15:0>													
DMA3PAD	03AC	PAD<15:0>													
DMA3CNT	03AE	—	—	—	—	—	—	CNT<9:0>							
DMACS0	03E0	—	—	—	—	PWCOL3	PWCOL2	PWCOL1	PWCOL0	—	—	—	—	XWCOL3	XWCOL2
DMACS1	03E2	—	—	—	—	LSTCH<3:0>				—	—	—	—	PPST3	PPST2
DSADR	03E4	DSADR<15:0>													

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

**TABLE 4-37: ECAN1 REGISTER MAP WHEN C1CTRL1.WIN = 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	
C1CTRL1	0600	—	—	CSIDL	ABAT	—	REQOP<2:0>			OPMODE<2:0>			—	CANCAP	—	
C1CTRL2	0602	—	—	—	—	—	—	—	—	—	—	—	—	—	DNCNT	
C1VEC	0604	—	—	—	FILHIT<4:0>				—	ICODE<6:0>						
C1FCTRL	0606	DMABS<2:0>			—	—	—	—	—	—	—	—	—	FSA<5:0>		
C1FIFO	0608	—	—	FBP<5:0>					—	—	FNRB<5:0>					
C1INTF	060A	—	—	TXBO	TXBP	RXBP	TXWAR	RXWAR	EWARN	IVRIF	WAKIF	ERRIF	—	FIFOIF	RBOIF	
C1INTE	060C	—	—	—	—	—	—	—	—	IVRIE	WAKIE	ERRIE	—	FIFOIE	RBOIE	
C1EC	060E	TERRCNT<7:0>							RERRCNT<7:0>							
C1CFG1	0610	—	—	—	—	—	—	—	—	SJW<1:0>		BRP<5:0>				
C1CFG2	0612	—	WAKFIL	—	—	—	SEG2PH<2:0>			SEG2PHTS	SAM	SEG1PH<2:0>				
C1FEN1	0614	FLTEN15	FLTEN14	FLTEN13	FLTEN12	FLTEN11	FLTEN10	FLTEN9	FLTEN8	FLTEN7	FLTEN6	FLTEN5	FLTEN4	FLTEN3	FLTEN2	
C1FMSKSEL1	0618	F7MSK<1:0>		F6MSK<1:0>		F5MSK<1:0>		F4MSK<1:0>		F3MSK<1:0>		F2MSK<1:0>		F1MSK<1:0>		
C1FMSKSEL2	061A	F15MSK<1:0>		F14MSK<1:0>		F13MSK<1:0>		F12MSK<1:0>		F11MSK<1:0>		F10MSK<1:0>		F9MSK<1:0>		

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

**TABLE 4-38: ECAN1 REGISTER MAP WHEN C1CTRL1.WIN = 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
	0600-061E	See definition when WIN = x													
C1RXFUL1	0620	RXFUL15	RXFUL14	RXFUL13	RXFUL12	RXFUL11	RXFUL10	RXFUL9	RXFUL8	RXFUL7	RXFUL6	RXFUL5	RXFUL4	RXFUL3	RXFUL2
C1RXFUL2	0622	RXFUL31	RXFUL30	RXFUL29	RXFUL28	RXFUL27	RXFUL26	RXFUL25	RXFUL24	RXFUL23	RXFUL22	RXFUL21	RXFUL20	RXFUL19	RXFUL18
C1RXOVF1	0628	RXOVF15	RXOVF14	RXOVF13	RXOVF12	RXOVF11	RXOVF10	RXOVF9	RXOVF8	RXOVF7	RXOVF6	RXOVF5	RXOVF4	RXOVF3	RXOVF2
C1RXOVF2	062A	RXOVF31	RXOVF30	RXOVF29	RXOVF28	RXOVF27	RXOVF26	RXOVF25	RXOVF24	RXOVF23	RXOVF22	RXOVF21	RXOVF20	RXOVF19	RXOVF18
C1TR01CON	0630	TXEN1	TXABT1	TXLARB1	TXERR1	TXREQ1	RTREN1	TX1PRI<1:0>		TXEN0	TXABT0	TXLARB0	TXERR0	TXREQ0	RTREN0
C1TR23CON	0632	TXEN3	TXABT3	TXLARB3	TXERR3	TXREQ3	RTREN3	TX3PRI<1:0>		TXEN2	TXABT2	TXLARB2	TXERR2	TXREQ2	RTREN2
C1TR45CON	0634	TXEN5	TXABT5	TXLARB5	TXERR5	TXREQ5	RTREN5	TX5PRI<1:0>		TXEN4	TXABT4	TXLARB4	TXERR4	TXREQ4	RTREN4
C1TR67CON	0636	TXEN7	TXABT7	TXLARB7	TXERR7	TXREQ7	RTREN7	TX7PRI<1:0>		TXEN6	TXABT6	TXLARB6	TXERR6	TXREQ6	RTREN6
C1RXD	0640	Received Data Word													
C1TXD	0642	Transmit Data Word													

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



**TABLE 4-39: ECAN1 REGISTER MAP WHEN C1CTRL1.WIN = 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	
	0600-061E	See definition when WIN = x														
C1BUFNT1	0620	F3BP<3:0>			F2BP<3:0>			F1BP<3:0>			F0BP<3:0>					
C1BUFNT2	0622	F7BP<3:0>			F6BP<3:0>			F5BP<3:0>			F4BP<3:0>					
C1BUFNT3	0624	F11BP<3:0>			F10BP<3:0>			F9BP<3:0>			F8BP<3:0>					
C1BUFNT4	0626	F15BP<3:0>			F14BP<3:0>			F13BP<3:0>			F12BP<3:0>					
C1RXM0SID	0630	SID<10:3>						SID<2:0>			—	MIDE	—			
C1RXM0EID	0632	EID<15:8>						EID<7:0>								
C1RXM1SID	0634	SID<10:3>						SID<2:0>			—	MIDE	—			
C1RXM1EID	0636	EID<15:8>						EID<7:0>								
C1RXM2SID	0638	SID<10:3>						SID<2:0>			—	MIDE	—			
C1RXM2EID	063A	EID<15:8>						EID<7:0>								
C1RXF0SID	0640	SID<10:3>						SID<2:0>			—	EXIDE	—			
C1RXF0EID	0642	EID<15:8>						EID<7:0>								
C1RXF1SID	0644	SID<10:3>						SID<2:0>			—	EXIDE	—			
C1RXF1EID	0646	EID<15:8>						EID<7:0>								
C1RXF2SID	0648	SID<10:3>						SID<2:0>			—	EXIDE	—			
C1RXF2EID	064A	EID<15:8>						EID<7:0>								
C1RXF3SID	064C	SID<10:3>						SID<2:0>			—	EXIDE	—			
C1RXF3EID	064E	EID<15:8>						EID<7:0>								
C1RXF4SID	0650	SID<10:3>						SID<2:0>			—	EXIDE	—			
C1RXF4EID	0652	EID<15:8>						EID<7:0>								
C1RXF5SID	0654	SID<10:3>						SID<2:0>			—	EXIDE	—			
C1RXF5EID	0656	EID<15:8>						EID<7:0>								
C1RXF6SID	0658	SID<10:3>						SID<2:0>			—	EXIDE	—			
C1RXF6EID	065A	EID<15:8>						EID<7:0>								
C1RXF7SID	065C	SID<10:3>						SID<2:0>			—	EXIDE	—			
C1RXF7EID	065E	EID<15:8>						EID<7:0>								
C1RXF8SID	0660	SID<10:3>						SID<2:0>			—	EXIDE	—			
C1RXF8EID	0662	EID<15:8>						EID<7:0>								
C1RXF9SID	0664	SID<10:3>						SID<2:0>			—	EXIDE	—			
C1RXF9EID	0666	EID<15:8>						EID<7:0>								
C1RXF10SID	0668	SID<10:3>						SID<2:0>			—	EXIDE	—			
C1RXF10EID	066A	EID<15:8>						EID<7:0>								
C1RXF11SID	066C	SID<10:3>						SID<2:0>			—	EXIDE	—			

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

**TABLE 4-39: ECAN1 REGISTER MAP WHEN C1CTRL1.WIN = 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	
C1RXF11EID	066E	EID<15:8>									EID<7:0>					
C1RXF12SID	0670	SID<10:3>									SID<2:0>		—	EXIDE	—	
C1RXF12EID	0672	EID<15:8>									EID<7:0>					
C1RXF13SID	0674	SID<10:3>									SID<2:0>		—	EXIDE	—	
C1RXF13EID	0676	EID<15:8>									EID<7:0>					
C1RXF14SID	0678	SID<10:3>									SID<2:0>		—	EXIDE	—	
C1RXF14EID	067A	EID<15:8>									EID<7:0>					
C1RXF15SID	067C	SID<10:3>									SID<2:0>		—	EXIDE	—	
C1RXF15EID	067E	EID<15:8>									EID<7:0>					

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

**TABLE 4-40: ANALOG COMPARATOR CONTROL REGISTER MAP**

File Name	ADR	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
CMPCON1	0540	CMPON	—	CMPSIDL	—	—	—	—	DACOE	INSEL<1:0>	EXTREF	—	CMPSTAT	—	—
CMPDAC1	0542	—	—	-	—	—	—	CMREF<9:0>							
CMPCON2	0544	CMPON	—	CMPSIDL	—	—	—	—	DACOE	INSEL<1:0>	EXTREF	—	CMPSTAT	—	—
CMPDAC2	0546	—	—	-	—	—	—	CMREF<9:0>							
CMPCON3	0548	CMPON	—	CMPSIDL	—	—	—	—	DACOE	INSEL<1:0>	EXTREF	—	CMPSTAT	—	—
CMPDAC3	054A	—	—	-	—	—	—	CMREF<9:0>							
CMPCON4	054C	CMPON	—	CMPSIDL	—	—	—	—	DACOE	INSEL<1:0>	EXTREF	—	CMPSTAT	—	—
CMPDAC4	054E	—	—	—	—	—	—	CMREF<9:0>							

**TABLE 4-41: PORTA REGISTER MAP FOR dsPIC33FJ32GS610 AND dsPIC33FJ64GS610 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
TRISA	02C0	TRISA15	TRISA14	—	—	—	TRISA10	TRISA9	—	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2
PORTA	02C2	RA15	RA14	—	—	—	RA10	RA9	—	RA7	RA6	RA5	RA4	RA3	RA2
LATA	02C4	LATA15	LATA14	—	—	—	LATA10	LATA9	—	LATA7	LATA6	LATA5	LATA4	LATA3	LATA2
ODCA	02C6	ODCA15	ODCA14	—	—	—	ODCA10	ODCA9	—	—	—	ODCA5	ODCA4	—	—

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

**TABLE 4-42: PORTA REGISTER MAP FOR dsPIC33FJ32GS608 AND dsPIC33FJ64GS608 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
TRISA	02C0	TRISA15	TRISA14	—	—	—	TRISA10	TRISA9	—	—	—	—	—	—	—
PORTA	02C2	RA15	RA14	—	—	—	RA10	RA9	—	—	—	—	—	—	—
LATA	02C4	LATA15	LATA14	—	—	—	LATA10	LATA9	—	—	—	—	—	—	—
ODCA	02C6	ODCA15	ODCA14	—	—	—	ODCA10	ODCA9	—	—	—	—	—	—	—

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

**TABLE 4-43: PORTB 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
TRISB	02C8	TRISB15	TRISB14	TRISB13	TRISB12	TRISB11	TRISB10	TRISB9	TRISB8	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2
PORTB	02CA	RB15	RB14	RB13	RB12	RB11	RB10	RB9	RB8	RB7	RB6	RB5	RB4	RB3	RB2
LATB	02CC	LATB15	LATB14	LATB13	LATB12	LATB11	LATB10	LATB9	LATB8	LATB7	LATB6	LATB5	LATB4	LATB3	LATB2

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

**TABLE 4-44: PORTC REGISTER MAP FOR dsPIC33FJ32GS610 AND dsPIC33FJ64GS610 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
TRISC	02D0	TRISC15	TRISC14	TRISC13	TRISC12	—	—	—	—	—	—	—	TRISC4	TRISC3	TRISC2
PORTC	02D2	RC15	RC14	RC13	RC12	—	—	—	—	—	—	—	RC4	RC3	RC2
LATC	02D4	LATC15	LATC14	LATC13	LATC12	—	—	—	—	—	—	—	LATC4	LATC3	LATC2

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

**TABLE 4-45: PORTC REGISTER MAP FOR dsPIC33FJ32GS608 AND dsPIC33FJ64GS608 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
TRISC	02D0	TRISC15	TRISC14	TRISC13	TRISC12	—	—	—	—	—	—	—	—	—	TRISC2
PORTC	02D2	RC15	RC14	RC13	RC12	—	—	—	—	—	—	—	—	—	RC2
LATC	02D4	LATC15	LATC14	LATC13	LATC12	—	—	—	—	—	—	—	—	—	LATC2

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

**TABLE 4-46: PORTC REGISTER MAP FOR dsPIC33FJ32GS406/606 AND dsPIC33FJ64GS406/606 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
TRISC	02D0	TRISC15	TRISC14	TRISC13	TRISC12	—	—	—	—	—	—	—	—	—	—
PORTC	02D2	RC15	RC14	RC13	RC12	—	—	—	—	—	—	—	—	—	—
LATC	02D4	LATC15	LATC14	LATC13	LATC12	—	—	—	—	—	—	—	—	—	—

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

**TABLE 4-47: PORTD REGISTER MAP FOR dsPIC33FJ32GS608/610 AND dsPIC33FJ64GS608/610 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
TRISD	02D8	TRISD15	TRISD14	TRISD13	TRISD12	TRISD11	TRISD10	TRISD9	TRISD8	TRISD7	TRISD6	TRISD5	TRISD4	TRISD3	TRISD2
PORTD	02DA	RD15	RD14	RD13	RD12	RD11	RD10	RD9	RD8	RD7	RD6	RD5	RD4	RD3	RD2
LATD	02DC	LATD15	LATD14	LATD13	LATD12	LATD11	LATD10	LATD9	LATD8	LATD7	LATD6	LATD5	LATD4	LATD3	LATD2
ODCD	02DE	ODCD15	ODCD14	ODCD13	ODCD12	ODCD11	ODCD10	ODCD9	ODCD8	ODCD7	ODCD6	ODCD5	ODCD4	ODCD3	ODCD2

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

**TABLE 4-48: PORTD REGISTER MAP FOR dsPIC33FJ32GS406/606 AND dsPIC33FJ64GS406/606 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
TRISD	02D8	—	—	—	—	TRISD11	TRISD10	TRISD9	TRISD8	TRISD7	TRISD6	TRISD5	TRISD4	TRISD3	TRISD2
PORTD	02DA	—	—	—	—	RD11	RD10	RD9	RD8	RD7	RD6	RD5	RD4	RD3	RD2
LATD	02DC	—	—	—	—	LATD11	LATD10	LATD9	LATD8	LATD7	LATD6	LATD5	LATD4	LATD3	LATD2
ODCD	02DE	—	—	—	—	ODCD11	ODCD10	ODCD9	ODCD8	ODCD7	ODCD6	ODCD5	ODCD4	ODCD3	ODCD2

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

**TABLE 4-49: PORTE REGISTER MAP FOR dsPIC33FJ32GS608/610 AND dsPIC33FJ64GS608/610 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
TRISE	02E0	—	—	—	—	—	—	TRISE9	TRISE8	TRISE7	TRISE6	TRISE5	TRISE4	TRISE3	TRISE2
PORTE	02E2	—	—	—	—	—	—	RE9	RE8	RE7	RE6	RE5	RE4	RE3	RE2
LATE	02E4	—	—	—	—	—	—	LATE9	LATE8	LATE7	LATE6	LATE5	LATE4	LATE3	LATE2
ODCE	02E6	—	—	—	—	—	—	—	—	ODCE7	ODCE6	ODCE5	ODCE4	ODCE3	ODCE2

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

**TABLE 4-50: PORTE REGISTER MAP FOR dsPIC33FJ32GS406/606 AND dsPIC33FJ64GS406/606 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
TRISE	02E0	—	—	—	—	—	—	—	—	TRISE7	TRISE6	TRISE5	TRISE4	TRISE3	TRISE2
PORTE	02E2	—	—	—	—	—	—	—	—	RE7	RE6	RE5	RE4	RE3	RE2
LATE	02E4	—	—	—	—	—	—	—	—	LATE7	LATE6	LATE5	LATE4	LATE3	LATE2
ODCE	02E6	—	—	—	—	—	—	—	—	ODCE7	ODCE6	ODCE5	ODCE4	ODCE3	ODCE2

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

**TABLE 4-51: PORTF REGISTER MAP FOR dsPIC33FJ32GS610 AND dsPIC33FJ64GS610 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
TRISF	02E8	—	—	TRISF13	TRISF12	—	—	—	TRISF8	TRISF7	TRISF6	TRISF5	TRISF4	TRISF3	TRISF2
PORTF	02EA	—	—	RF13	RF12	—	—	—	RF8	RF7	RF6	RF5	RF4	RF3	RF2
LATF	02EC	—	—	LATF13	LATF12	—	—	—	LATF8	LATF7	LATF6	LATF5	LATF4	LATF3	LATF2
ODCF	02EE	—	—	ODCF13	ODCF12	—	—	—	ODCF8	ODCF7	ODCF6	—	—	ODCF3	ODCF2

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

**TABLE 4-52: PORTF REGISTER MAP FOR dsPIC33FJ32GS608 AND dsPIC33FJ64GS608 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
TRISF	02E8	—	—	—	—	—	—	—	TRISF8	TRISF7	TRISF6	TRISF5	TRISF4	TRISF3	TRISF2
PORTF	02EA	—	—	—	—	—	—	—	RF8	RF7	RF6	RF5	RF4	RF3	RF2
LATF	02EC	—	—	—	—	—	—	—	LATF8	LATF7	LATF6	LATF5	LATF4	LATF3	LATF2
ODCF	02EE	—	—	—	—	—	—	—	ODCF8	ODCF7	ODCF6	—	—	ODCF3	ODCF2

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

**TABLE 4-53: PORTF REGISTER MAP FOR dsPIC33FJ32GS406/606 AND dsPIC33FJ64GS406/606 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
TRISF	02E8	—	—	—	—	—	—	—	—	—	TRISF6	TRISF5	TRISF4	TRISF3	TRISF2
PORTF	02EA	—	—	—	—	—	—	—	—	—	RF6	RF5	RF4	RF3	RF2
LATF	02EC	—	—	—	—	—	—	—	—	—	LATF6	LATF5	LATF4	LATF3	LATF2
ODCF	02EE	—	—	—	—	—	—	—	—	—	ODCF6	—	—	ODCF3	ODCF2

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

**TABLE 4-54: PORTG REGISTER MAP FOR dsPIC33FJ32GS610 AND dsPIC33FJ64GS610 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
TRISG	02F0	TRISG15	TRISG14	TRISG13	TRISG12	—	—	TRISG9	TRISG8	TRISG7	TRISG6	—	—	TRISG3	TRISG2
PORTG	02F2	RG15	RG14	RG13	RG12	—	—	RG9	RG8	RG7	RG6	—	—	RG3	RG2
LATG	02F4	LATG15	LATG14	LATG13	LATG12	—	—	LATG9	LATG8	LATG7	LATG6	—	—	LATG3	LATG2
ODCG	02F6	ODCG15	ODCG14	ODCG13	ODCG12	—	—	ODCG9	ODCG8	ODCG7	ODCG6	—	—	ODCG3	ODCG2

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

**TABLE 4-55: PORTG REGISTER MAP FOR dsPIC33FJ32GS608 AND dsPIC33FJ64GS608 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
TRISG	02F0	—	—	—	—	—	—	TRISG9	TRISG8	TRISG7	TRISG6	—	—	TRISG3	TRISG2
PORTG	02F2	—	—	—	—	—	—	RG9	RG8	RG7	RG6	—	—	RG3	RG2
LATG	02F4	—	—	—	—	—	—	LATG9	LATG8	LATG7	LATG6	—	—	LATG3	LATG2
ODCG	02F6	—	—	—	—	—	—	ODCG9	ODCG8	ODCG7	ODCG6	—	—	ODCG3	ODCG2

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

**TABLE 4-56: PORTG REGISTER MAP FOR dsPIC33FJ32GS406/606 AND dsPIC33FJ64GS406/606 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
TRISG	02F0	—	—	—	—	—	—	TRISG9	TRISG8	TRISG7	TRISG6	—	—	TRISG3	TRISG2
PORTG	02F2	—	—	—	—	—	—	RG9	RG8	RG7	RG6	—	—	RG3	RG2
LATG	02F4	—	—	—	—	—	—	LATG9	LATG8	LATG7	LATG6	—	—	LATG3	LATG2
ODCG	02F6	—	—	—	—	—	—	ODCG9	ODCG8	ODCG7	ODCG6	—	—	ODCG3	ODCG2

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

**TABLE 4-57: SYSTEM CONTROL 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
RCON	0740	TRAPR	IOPUWR	—	—	—	—	—	VREGS	EXTR	SWR	SWDTEN	WDTO	SLEEP	IDLE
OSCCON	0742	—	COSC<2:0>			—	NOSC<2:0>			CLKLOCK	—	LOCK	—	CF	—
CLKDIV	0744	ROI	DOZE<2:0>			DOZEN	FRCDIV<2:0>			PLLPOST<1:0>		—	PLLPRE<4:0>		
PLLFBD	0746	—	—	—	—	—	—	—	PLLDIV<8:0>						
OSCTUN	0748	—	—	—	—	—	—	—	—	—	—	TUN<5:0>			
REFOCON	074E	ROON	—	ROSSLP	ROSEL	RODIV<3:0>				—	—	—	—	—	—
ACLKCON	0750	ENAPLL	APLLCK	SELACLK	—	—	APSTSCLR<2:0>			ASRCSEL	FRCSEL	—	—	—	—

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

**Note 1:** The RCON register reset values are dependent on type of reset.

**Note 2:** The OSCCON register reset values are dependent on the FOSC configuration bits, and on type of reset.

**TABLE 4-58: NVM 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
NVMCON	0760	WR	WREN	WRERR	—	—	—	—	—	—	ERASE	—	—	NVMKEY	
NVMKEY	0766	—	—	—	—	—	—	—	—	NVMKEY<7:0>					

**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 4-59: PMD REGISTER MAP FOR dsPIC33FJ64GS610 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
PMD1	0770	T5MD	T4MD	T3MD	T2MD	T1MD	QE11MD	PWMMD	—	I2C1MD	U2MD	U1MD	SPI2MD	SPI1MD	—
PMD2	0772	—	—	—	—	IC4MD	IC3MD	IC2MD	IC1MD	—	—	—	—	OC4MD	OC3MD
PMD3	0774	—	—	—	—	—	CMPMD	—	—	—	—	QE12MD	—	—	—
PMD4	0776	—	—	—	—	—	—	—	—	—	—	—	—	REFOMD	—
PMD6	077A	PWM8MD	PWM7MD	PWM6MD	PWM5MD	PWM4MD	PWM3MD	PWM2MD	PWM1MD	—	—	—	—	—	—
PMD7	077C	—	—	—	—	CMP4MD	CMP3MD	CMP2MD	CMP1MD	—	—	—	—	—	—

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

**TABLE 4-60: PMD REGISTER MAP FOR dsPIC33FJ32GS610 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
PMD1	0770	T5MD	T4MD	T3MD	T2MD	T1MD	QE11MD	PWMMD	—	I2C1MD	U2MD	U1MD	SPI2MD	SPI1MD	—
PMD2	0772	—	—	—	—	IC4MD	IC3MD	IC2MD	IC1MD	—	—	—	—	OC4MD	OC3MD
PMD3	0774	—	—	—	—	—	CMPMD	—	—	—	—	QE12MD	—	—	—
PMD4	0776	—	—	—	—	—	—	—	—	—	—	—	—	REFOMD	—
PMD6	077A	PWM8MD	PWM7MD	PWM6MD	PWM5MD	PWM4MD	PWM3MD	PWM2MD	PWM1MD	—	—	—	—	—	—
PMD7	077C	—	—	—	—	CMP4MD	CMP3MD	CMP2MD	CMP1MD	—	—	—	—	—	—

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



**TABLE 4-61: PMD REGISTER MAP FOR dsPIC33FJ64GS608 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
PMD1	0770	T5MD	T4MD	T3MD	T2MD	T1MD	QE11MD	PWMMD	—	I2C1MD	U2MD	U1MD	SPI2MD	SPI1MD	—
PMD2	0772	—	—	—	—	IC4MD	IC3MD	IC2MD	IC1MD	—	—	—	—	OC4MD	OC3MD
PMD3	0774	—	—	—	—	—	CMPMD	—	—	—	—	QEI2MD	—	—	—
PMD4	0776	—	—	—	—	—	—	—	—	—	—	—	—	REFOMD	—
PMD6	077A	PWM8MD	PWM7MD	PWM6MD	PWM5MD	PWM4MD	PWM3MD	PWM2MD	PWM1MD	—	—	—	—	—	—
PMD7	077C	—	—	—	—	CMP4MD	CMP3MD	CMP2MD	CMP1MD	—	—	—	—	—	—

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

**TABLE 4-62: PMD REGISTER MAP FOR dsPIC33FJ32GS608 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
PMD1	0770	T5MD	T4MD	T3MD	T2MD	T1MD	QE11MD	PWMMD	—	I2C1MD	U2MD	U1MD	SPI2MD	SPI1MD	—
PMD2	0772	—	—	—	—	IC4MD	IC3MD	IC2MD	IC1MD	—	—	—	—	OC4MD	OC3MD
PMD3	0774	—	—	—	—	—	CMPMD	—	—	—	—	QEI2MD	—	—	—
PMD4	0776	—	—	—	—	—	—	—	—	—	—	—	—	REFOMD	—
PMD6	077A	PWM8MD	PWM7MD	PWM6MD	PWM5MD	PWM4MD	PWM3MD	PWM2MD	PWM1MD	—	—	—	—	—	—
PMD7	077C	—	—	—	—	CMP4MD	CMP3MD	CMP2MD	CMP1MD	—	—	—	—	—	—

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

**TABLE 4-63: PMD REGISTER MAP FOR dsPIC33FJ64GS606 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
PMD1	0770	T5MD	T4MD	T3MD	T2MD	T1MD	QE11MD	PWMMD	—	I2C1MD	U2MD	U1MD	SPI2MD	SPI1MD	—
PMD2	0772	—	—	—	—	IC4MD	IC3MD	IC2MD	IC1MD	—	—	—	—	OC4MD	OC3MD
PMD3	0774	—	—	—	—	—	CMPMD	—	—	—	—	QEI2MD	—	—	—
PMD4	0776	—	—	—	—	—	—	—	—	—	—	—	—	REFOMD	—
PMD6	077A	—	—	PWM6MD	PWM5MD	PWM4MD	PWM3MD	PWM2MD	PWM1MD	—	—	—	—	—	—
PMD7	077C	—	—	—	—	CMP4MD	CMP3MD	CMP2MD	CMP1MD	—	—	—	—	—	—

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

**TABLE 4-64: PMD REGISTER MAP FOR dsPIC33FJ32GS606 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
PMD1	0770	T5MD	T4MD	T3MD	T2MD	T1MD	QE1MD	PWMMMD	—	I2C1MD	U2MD	U1MD	SPI2MD	SPI1MD	—
PMD2	0772	—	—	—	—	IC4MD	IC3MD	IC2MD	IC1MD	—	—	—	—	OC4MD	OC3MD
PMD3	0774	—	—	—	—	—	CMPMD	—	—	—	—	QEI2MD	—	—	—
PMD4	0776	—	—	—	—	—	—	—	—	—	—	—	—	REFOMD	—
PMD6	077A	—	—	PWM6MD	PWM5MD	PWM4MD	PWM3MD	PWM2MD	PWM1MD	—	—	—	—	—	—
PMD7	077C	—	—	—	—	CMP4MD	CMP3MD	CMP2MD	CMP1MD	—	—	—	—	—	—

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

**TABLE 4-65: PMD REGISTER MAP FOR dsPIC33FJ32GS406 AND dsPIC33FJ64GS406 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
PMD1	0770	T5MD	T4MD	T3MD	T2MD	T1MD	QE1MD	PWMMMD	—	I2C1MD	U2MD	U1MD	SPI2MD	SPI1MD	—
PMD2	0772	—	—	—	—	IC4MD	IC3MD	IC2MD	IC1MD	—	—	—	—	OC4MD	OC3MD
PMD3	0774	—	—	—	—	—	—	—	—	—	—	QEI2MD	—	—	—
PMD4	0776	—	—	—	—	—	—	—	—	—	—	—	—	REFOMD	—
PMD6	077A	—	—	PWM6MD	PWM5MD	PWM4MD	PWM3MD	PWM2MD	PWM1MD	—	—	—	—	—	—

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

4.2.7 SOFTWARE STACK

In addition to its use as a working register, the W15 register in the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 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 predecrements 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.

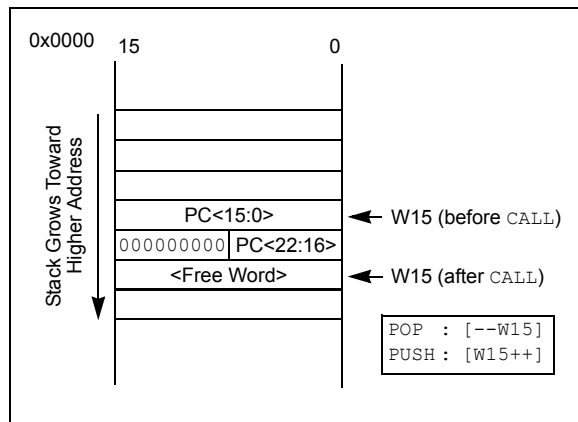
The Stack Pointer Limit register (SPLIM) associated with the Stack Pointer sets an upper address boundary for the stack. SPLIM is uninitialized at Reset. As is the case for the Stack Pointer, SPLIM<0> is forced to '0' because all stack operations must be word-aligned.

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. For example, to cause a stack error trap when the stack grows beyond address 0x1800 in RAM, initialize the SPLIM with the value 0x17FE.

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**



4.3 Instruction Addressing Modes

The addressing modes shown in Table 4-66 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 differ 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 three-operand MCU instructions are of the form:

$$\text{Operand 3} = \text{Operand 1} \langle \text{function} \rangle \text{Operand 2}$$

where Operand 1 is always a working register (that is, 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 can support different subsets of these addressing modes.

**TABLE 4-66: 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 Effective Address (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 (Register Indexed)	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 by both source and destination (but typically only used by one).

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 addressing modes given above. Individual instructions may support different subsets of these addressing modes.

### 4.3.4 MAC INSTRUCTIONS

The dual source operand DSP instructions (`CLR`, `ED`, `EDAC`, `MAC`, `MPY`, `MPY.N`, `MOV SAC` and `MSC`), also referred to as `MAC` instructions, use a simplified set of addressing modes to allow the user application to effectively manipulate the data pointers through register indirect tables.

The two-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 are always 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 available only 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 addressing modes outlined previously, 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 `NOB`, do not have any operands.

#### 4.4 Modulo Addressing

Modulo Addressing mode is a method used to provide 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 be configured to operate in only 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 that have a power-of-two length. As these buffers satisfy the start and end address criteria, they can operate in a bidirectional mode (that is, address boundary checks are 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

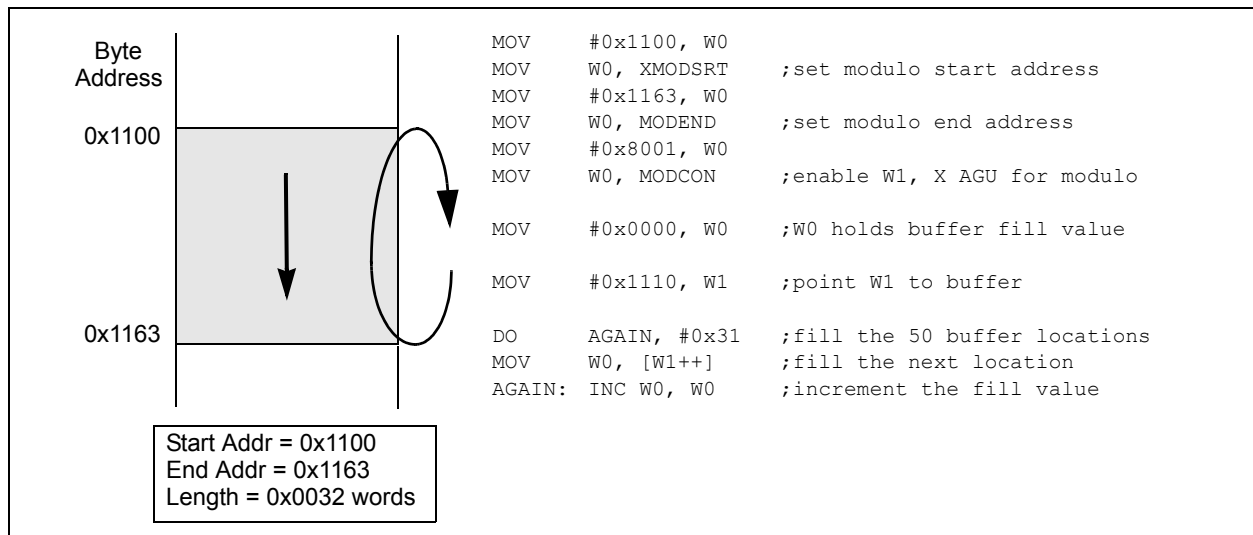
The Modulo and Bit-Reversed Addressing Control register, MODCON<15:0>, contains enable flags as well as a W register field to specify the W Address registers. The XWM and YWM fields select the registers that will operate with Modulo Addressing:

- If XWM = 15, X RAGU and X WAGU Modulo Addressing is disabled.
- 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. Address boundaries check for addresses equal to:

- Upper boundary addresses for incrementing buffers
- Lower boundary addresses for decrementing buffers

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 can, 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 (such as [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 re-ordering for radix-2 FFT algorithms. It is supported by the X AGU for data writes only.

The modifier, which can 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 in any of these situations:

- BWM bits (W register selection) in the MODCON register are any value other than '15' (the stack cannot be accessed using Bit-Reversed Addressing)
- The BREN bit is set in the XBREV register
- 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.

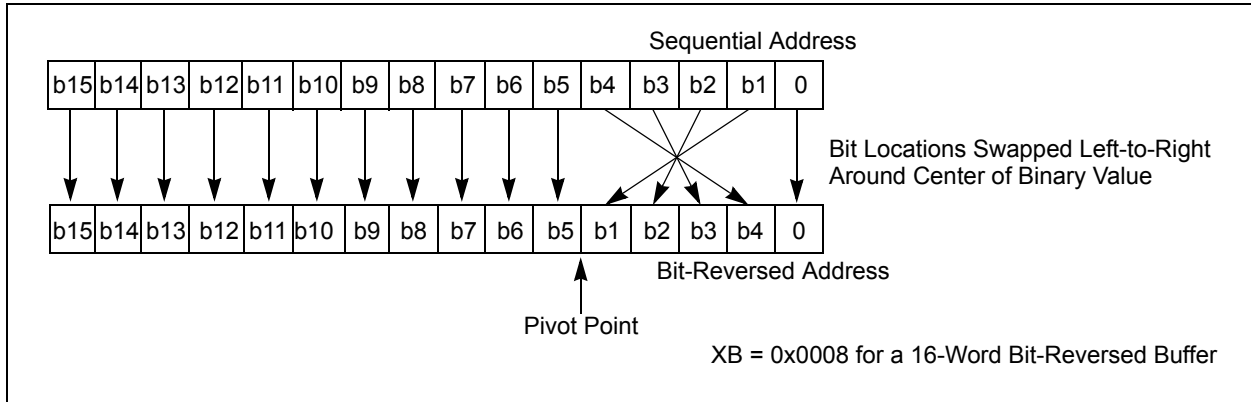
**Note:** All bit-reversed EA calculations assume word-sized data (LSb of every EA is always clear). The XB value is scaled accordingly to generate compatible (byte) addresses.

When enabled, Bit-Reversed Addressing is executed only 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, and 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. If an application attempts to do so, Bit-Reversed Addressing will assume priority when active for the X WAGU and X RAGU, 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 (XBREV<15>) bit, 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 4-67: BIT-REVERSED ADDRESS SEQUENCE (16-ENTRY)**

Normal Address					Bit-Reversed Address				
A3	A2	A1	A0	Decimal	A3	A2	A1	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 dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 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 dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 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 periodically. 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. The application 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 4-68 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, and D<15:0> refers to a data space word.

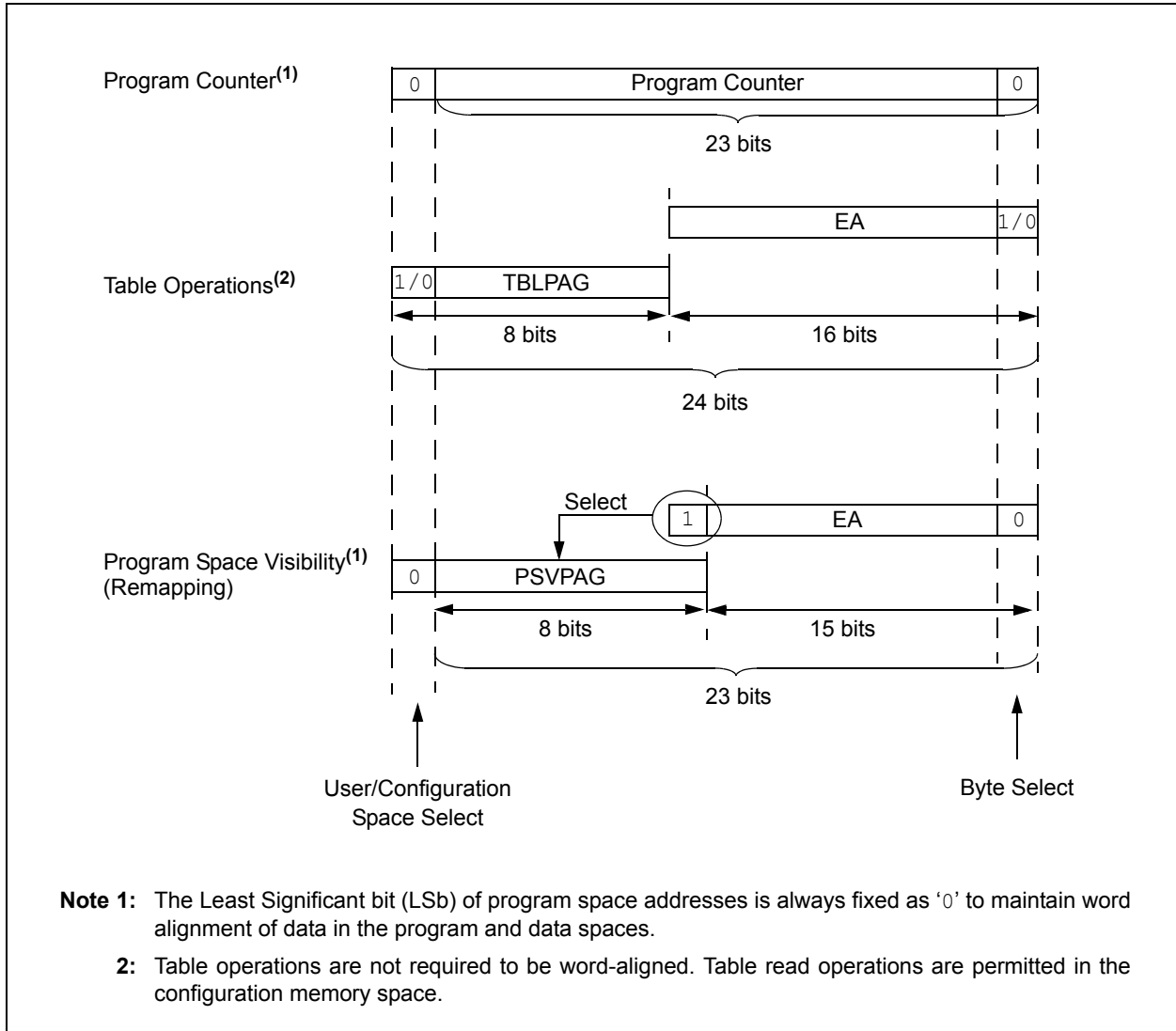
**TABLE 4-68: PROGRAM SPACE ADDRESS CONSTRUCTION**

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



FIGURE 4-9: DATA ACCESS FROM PROGRAM SPACE ADDRESS GENERATION



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-wide word address spaces, residing side by side, each with the same address range. `TBLRDL` and `TBLWTL` access the space that contains the least significant data word. `TBLRDH` and `TBLWTH` access the space that 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, this instruction maps the lower word of the program space location ( $P<15:0>$ ) to a data address ( $D<15:0>$ ).

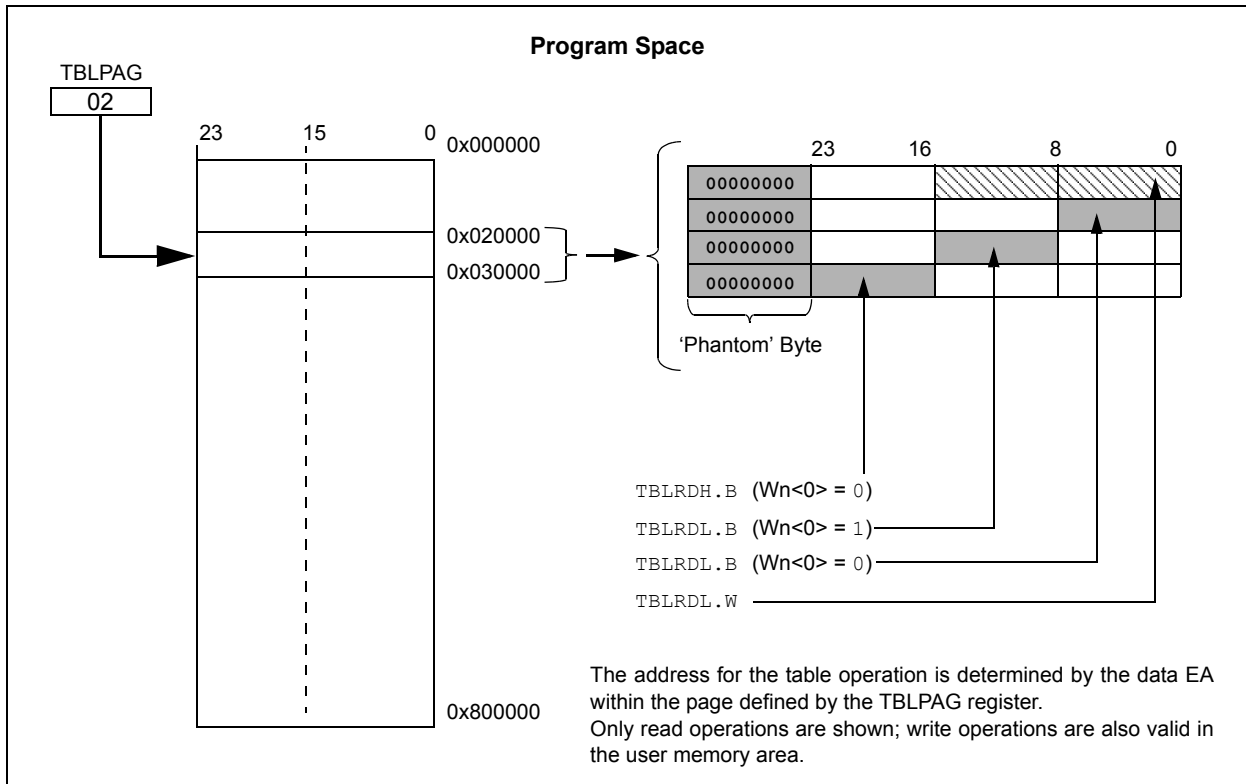
- In Byte mode, either the upper or lower byte of the lower program word is mapped to the lower byte of a data address. The upper byte is selected when Byte Select is '1'; the lower byte is selected when it is '0'.

- `TBLRDH` (Table Read High):
  - In Word mode, this instruction 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, this instruction maps the upper or lower byte of the program word to  $D<7:0>$  of the data address, in the `TBLRDL` instruction. The data is always '0' when the upper 'phantom' byte is selected (Byte Select = 1).

Similarly, 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 to stored constant data from the data space without the need to use special instructions (such as TBLRD/L/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. 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 a 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.

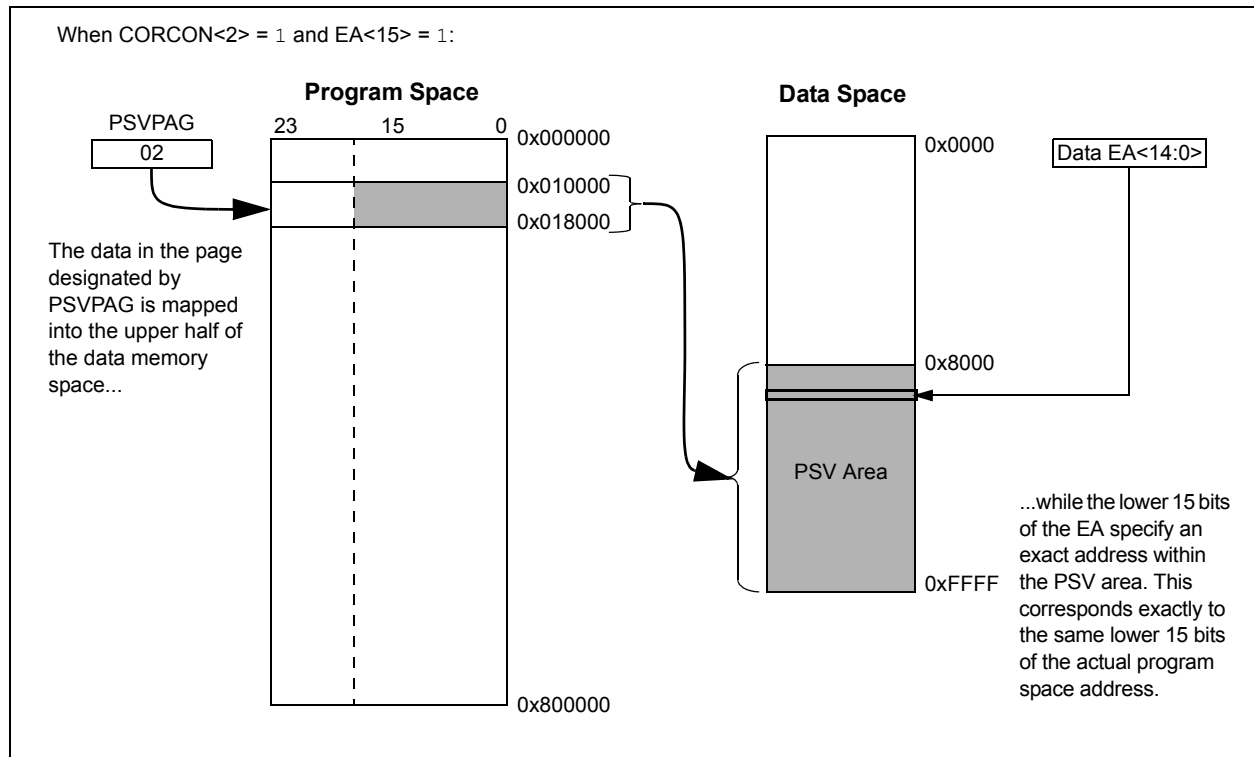
For operations that use PSV and are executed outside a REPEAT loop, the MOV and MOV.D instructions require one instruction cycle in addition to the specified execution time. All other instructions require two instruction cycles in addition to the specified execution time.

For operations that use PSV, and are executed inside a REPEAT loop, these instances 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 using PSV to access data, to execute in a single cycle.

FIGURE 4-11: PROGRAM SPACE VISIBILITY OPERATION



NOTES:

5.0 FLASH PROGRAM MEMORY

**Note 1:** This data sheet summarizes the features of the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 families of devices. 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](http://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 dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 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™)
- Run-Time Self-Programming (RTSP)

ICSP allows a dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 device to be serially programmed while in the end application circuit. This is done with two lines for programming clock and programming data (one of the alternate programming pin pairs: PGEC1/PGED1, PGEC2/PGED2 or PGEC3/

PGED3), 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 application can write program memory data, either in blocks or ‘rows’ of 64 instructions (192 bytes) at a time, or a single program memory word, and erase program memory in blocks or ‘pages’ of 512 instructions (1536 bytes) at a time.

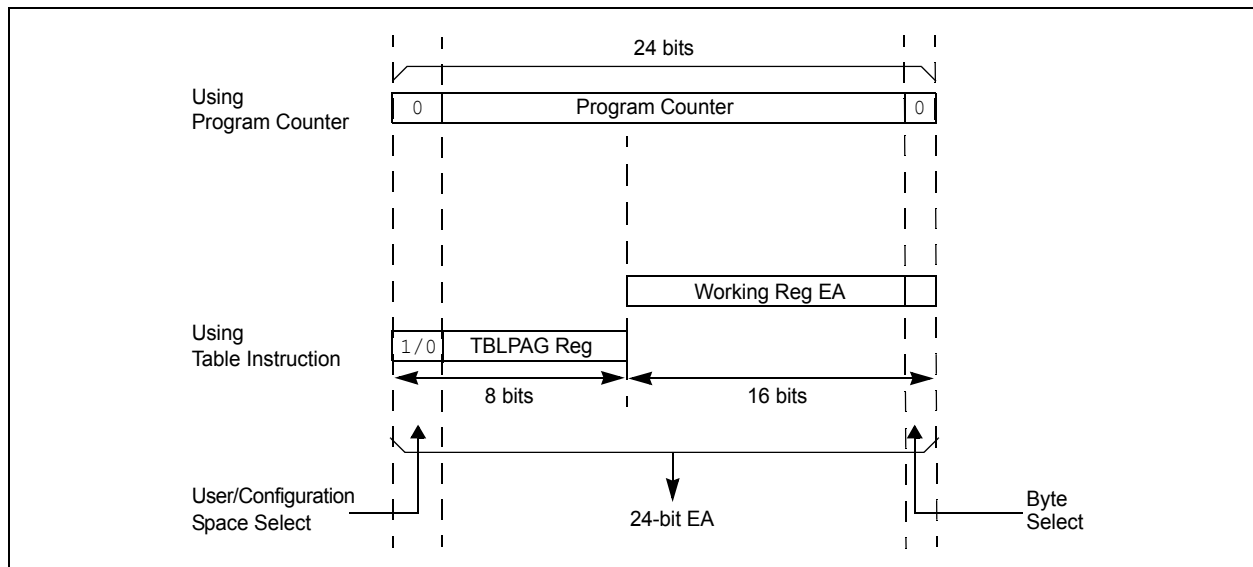
5.1 Table Instructions and Flash Programming

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

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

FIGURE 5-1: ADDRESSING FOR TABLE REGISTERS



**5.2 RTSP Operation**

The dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 Flash program memory array is organized into rows of 64 instructions or 192 bytes. RTSP allows the user application to erase a page of memory, which consists of eight rows (512 instructions) at a time, and to program one row or one word at a time. Table 27-12 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 sequentially. The instruction words loaded must always be from a group of 64 boundary.

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 27-20) 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 27-12).

**EQUATION 5-1: PROGRAMMING TIME**

$$T = \frac{11064 \text{ Cycles}}{7.37 \text{ MHz} \times (\text{FRC Accuracy})\% \times (\text{FRC Tuning})\%}$$

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 'b0000000, the minimum row write time is equal to Equation 5-2.

**EQUATION 5-2: MINIMUM ROW WRITE TIME**

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

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

**EQUATION 5-3: MAXIMUM ROW WRITE TIME**

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

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**

Two SFRs are 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 application 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

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

- bit 15      **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
  - 1101 = Erase general segment
  - 0011 = No operation
  - 0010 = Memory page erase operation
  - 0001 = No operation
  - 0000 = Erase a single Configuration register byte
  
  - If ERASE = 0:
  - 1111 = No operation
  - 1101 = No operation
  - 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.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 5-2: NVMKEY: NON-VOLATILE MEMORY KEY REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

W-0	W-0	W-0	W-0	W-0	W-0	W-0	W-0
NVMKEY<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

**Unimplemented:** Read as '0'

bit 7-0

**NVMKEY<7:0>:** Key Register bits (write-only)



## 5.4.1 PROGRAMMING ALGORITHM FOR FLASH PROGRAM MEMORY

One row of program Flash memory can be programmed at a time. To achieve this, it is necessary to erase the 8-row erase page that contains the desired row. The general process is:

1. Read eight rows of program memory (512 instructions) and store in data RAM.
2. Update the program data in RAM with the desired new data.
3. Erase the block (see [Example 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.
  - b) 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.
4. 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:
  - a) 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.
6. 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 application 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          ;
MOV    W0, NVMCON          ; Initialize NVMCON
; Init pointer to row to be ERASED
MOV    #tblpage(PROG_ADDR), W0      ;
MOV    W0, TBLPAG          ; Initialize PM Page Boundary SFR
MOV    #tbloffset(PROG_ADDR), W0    ; Initialize in-page EA[15:0] pointer
TBLWTL W0, [W0]            ; Set base address of erase block
DISI   #5                  ; Block all interrupts with priority <7
                                ; for next 5 instructions

MOV    #0x55, W0
MOV    W0, NVMKEY          ; Write the 55 key
MOV    #0xAA, W1
MOV    W1, NVMKEY          ; Write the AA key
BSET   NVMCON, #WR        ; Start the erase sequence
NOP
NOP                        ; Insert two NOPs after the erase
                                ; command is asserted
    
```

## EXAMPLE 5-2: LOADING THE WRITE BUFFERS

```
; Set up NVMCON for row programming operations
MOV    #0x4001, W0          ;
MOV    W0, NVMCON          ; Initialize NVMCON
; Set up a pointer to the first program memory location to be written
; program memory selected, and writes enabled
MOV    #0x0000, W0          ;
MOV    W0, TBLPAG          ; Initialize PM Page Boundary SFR
MOV    #0x6000, W0          ; An example program memory address
; Perform the TBLWT instructions to write the latches
; 0th_program_word
MOV    #LOW_WORD_0, W2      ;
MOV    #HIGH_BYTE_0, W3     ;
TBLWTL W2, [W0]            ; Write PM low word into program latch
TBLWTH W3, [W0++]          ; Write PM high byte into program latch
; 1st_program_word
MOV    #LOW_WORD_1, W2      ;
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
MOV    #LOW_WORD_2, W2      ;
MOV    #HIGH_BYTE_2, W3     ;
TBLWTL W2, [W0]            ; Write PM low word into program latch
TBLWTH W3, [W0++]          ; Write PM high byte into program latch
.
.
.
; 63rd_program_word
MOV    #LOW_WORD_31, W2     ;
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   #5                  ; Block all interrupts with priority <7
                          ; for next 5 instructions
MOV    #0x55, W0           ;
MOV    W0, NVMKEY          ; Write the 55 key
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
```

## 6.0 RESETS

**Note 1:** This data sheet summarizes the features of the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 families of devices. 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](http://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,  $\overline{\text{SYSRST}}$ . The following is a list of device Reset sources:

- POR: Power-on Reset
- BOR: Brown-out Reset
- $\overline{\text{MCLR}}$ : Master Clear Pin Reset
- SWR: Software **RESET** Instruction
- WDTO: Watchdog Timer Reset
- TRAPR: Trap Conflict Reset
- IOPUWR: Illegal Condition Device Reset
  - Illegal Opcode Reset
  - Uninitialized W Register Reset
  - Security Reset

A simplified block diagram of the Reset module is shown in [Figure 6-1](#).

Any active source of reset will make the  $\overline{\text{SYSRST}}$  signal active. On system Reset, some of the registers associated with the CPU and peripherals are forced to a known Reset state and some are unaffected.

**Note:** Refer to the specific peripheral section or **Section 3.0 “CPU”** of this data sheet for register Reset states.

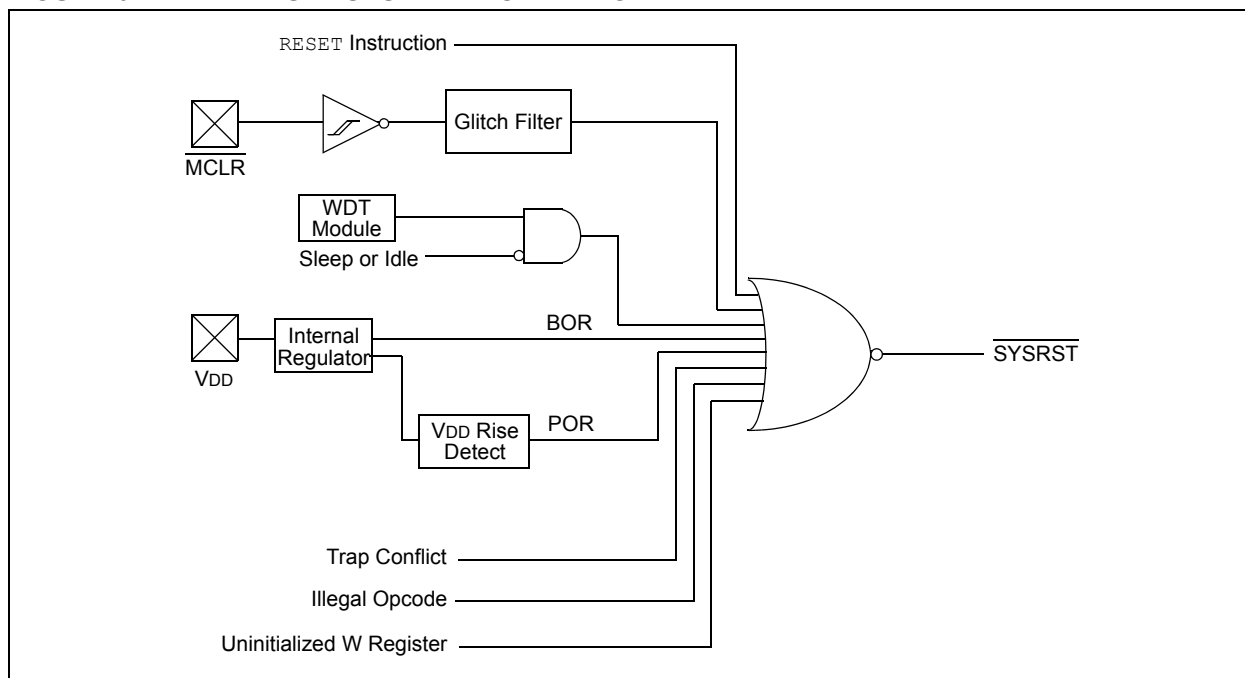
All types of device Reset sets a corresponding status bit in the RCON register to indicate the type of Reset (see [Register 6-1](#)).

A POR clears all the bits, except for the POR bit (RCON<0>), that are set. The user application 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 is meaningful.

**FIGURE 6-1: RESET SYSTEM BLOCK DIAGRAM**



# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

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

R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0	R/W-0
TRAPR	IOPUWR	—	—	—	—	—	VREGS
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                      W = Writable bit                      U = Unimplemented bit, read as '0'  
 -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
- bit 14            **IOPUWR:** Illegal Opcode or Uninitialized W Access Reset Flag bit  
                   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'
- bit 8            **VREGS:** Voltage Regulator Standby During Sleep bit  
                   1 = Voltage regulator is active during Sleep  
                   0 = Voltage regulator goes into Standby mode during Sleep
- bit 7            **EXTR:** External Reset Pin ( $\overline{\text{MCLR}}$ ) bit  
                   1 = A Master Clear (pin) Reset has occurred  
                   0 = A Master Clear (pin) Reset has not occurred
- bit 6            **SWR:** Software Reset Flag (Instruction) bit  
                   1 = A `RESET` instruction has been executed  
                   0 = A `RESET` instruction has not been executed
- bit 5            **SWDTEN:** Software Enable/Disable of WDT bit<sup>(2)</sup>  
                   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
- bit 1            **BOR:** Brown-out Reset Flag bit  
                   1 = A Brown-out Reset has occurred  
                   0 = 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 can be set or cleared in software. Setting one of these bits in software does not cause a device Reset.

**2:** If the `SWDTEN` Configuration bit is '1' (unprogrammed), the WDT is always enabled, regardless of the `SWDTEN` bit setting.

## 6.1 System Reset

The dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 families of devices have two types of Reset:

- Cold Reset
- Warm Reset

A cold Reset is the result of a Power-on Reset (POR) or a Brown-out Reset (BOR). On a cold Reset, the FNOSC Configuration bits in the FOSC Configuration register select the device clock source.

A warm Reset is the result of all the other Reset sources, including the RESET instruction. On warm Reset, the device will continue to operate from the current clock source as indicated by the Current Oscillator Selection (COSC<2:0>) bits in the Oscillator Control (OSCCON<14:12>) register.

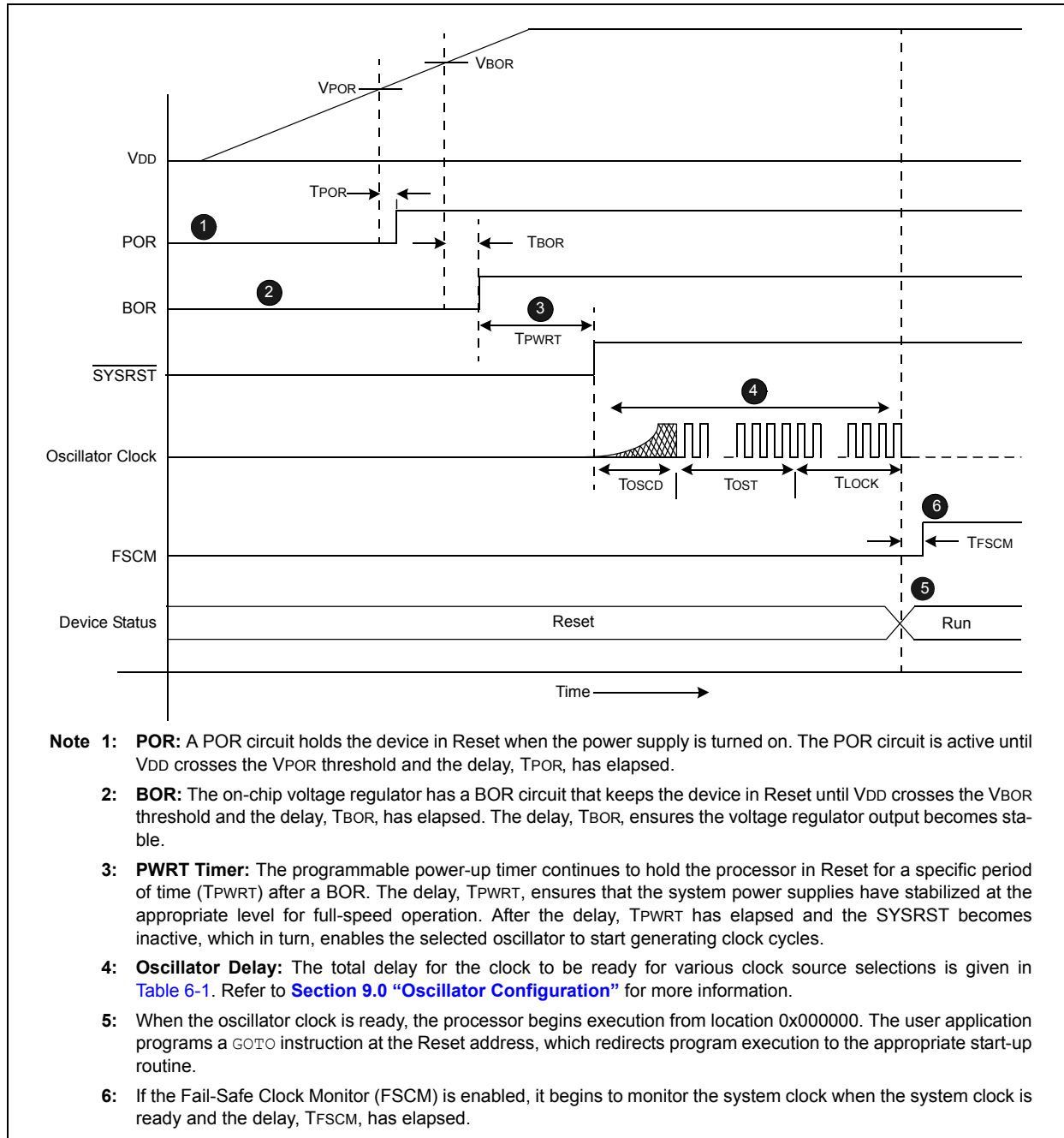
The device is kept in a Reset state until the system power supplies have stabilized at appropriate levels and the oscillator clock is ready. The sequence in which this occurs is described in [Figure 6-2](#).

**TABLE 6-1: OSCILLATOR DELAY**

Oscillator Mode	Oscillator Start-up Delay	Oscillator Start-up Timer	PLL Lock Time	Total Delay
FRC, FRCDIV16, FRCDIVN	T <sub>OSCD</sub> <sup>(1)</sup>	—	—	T <sub>OSCD</sub> <sup>(1)</sup>
FRCPLL	T <sub>OSCD</sub> <sup>(1)</sup>	—	T <sub>LOCK</sub> <sup>(3)</sup>	T <sub>OSCD</sub> + T <sub>LOCK</sub> <sup>(1,3)</sup>
XT	T <sub>OSCD</sub> <sup>(1)</sup>	T <sub>OST</sub> <sup>(2)</sup>	—	T <sub>OSCD</sub> + T <sub>OST</sub> <sup>(1,2)</sup>
HS	T <sub>OSCD</sub> <sup>(1)</sup>	T <sub>OST</sub> <sup>(2)</sup>	—	T <sub>OSCD</sub> + T <sub>OST</sub> <sup>(1,2)</sup>
EC	—	—	—	—
XTPLL	T <sub>OSCD</sub> <sup>(1)</sup>	T <sub>OST</sub> <sup>(2)</sup>	T <sub>LOCK</sub> <sup>(3)</sup>	T <sub>OSCD</sub> + T <sub>OST</sub> + T <sub>LOCK</sub> <sup>(1,2,3)</sup>
HSPLL	T <sub>OSCD</sub> <sup>(1)</sup>	T <sub>OST</sub> <sup>(2)</sup>	T <sub>LOCK</sub> <sup>(3)</sup>	T <sub>OSCD</sub> + T <sub>OST</sub> + T <sub>LOCK</sub> <sup>(1,2,3)</sup>
ECPLL	—	—	T <sub>LOCK</sub> <sup>(3)</sup>	T <sub>LOCK</sub> <sup>(3)</sup>
LPRC	T <sub>OSCD</sub> <sup>(1)</sup>	—	—	T <sub>OSCD</sub> <sup>(1)</sup>

- Note 1:** T<sub>OSCD</sub> = Oscillator start-up delay (1.1 μs max for FRC, 70 μs max for LPRC). Crystal oscillator start-up times vary with crystal characteristics, load capacitance, etc.
- 2:** T<sub>OST</sub> = Oscillator start-up timer delay (1024 oscillator clock period). For example, T<sub>OST</sub> = 102.4 μs for a 10 MHz crystal and T<sub>OST</sub> = 32 ms for a 32 kHz crystal.
- 3:** T<sub>LOCK</sub> = PLL lock time (1.5 ms nominal) if PLL is enabled.

FIGURE 6-2: SYSTEM RESET TIMING



**Note:** When the device exits the Reset condition (begins normal operation), the device operating parameters (voltage, frequency, temperature, etc.) must be within their operating ranges; otherwise, the device may not function correctly. The user application must ensure that the delay between the time power is first applied, and the time SYSRST becomes inactive, is long enough to get all operating parameters within specification.

### 6.2 Power-on Reset (POR)

A Power-on Reset (POR) circuit ensures the device is reset from power-on. The POR circuit is active until VDD crosses the VPOR threshold and the delay, TPOR, has elapsed. The delay, TPOR, ensures the internal device bias circuits become stable.

The device supply voltage characteristics must meet the specified starting voltage and rise rate requirements to generate the POR. Refer to [Section 27.0 “Electrical Characteristics”](#) for details.

The POR Status (POR) bit in the Reset Control (RCON<0>) register is set to indicate the Power-on Reset.

### 6.3 Brown-out Reset (BOR) and Power-up Timer (PWRT)

The on-chip regulator has a Brown-out Reset (BOR) circuit that resets the device when the VDD is too low ( $V_{DD} < V_{BOR}$ ) for proper device operation. The BOR circuit keeps the device in Reset until VDD crosses the

$V_{BOR}$  threshold and the delay, TBOR, has elapsed. The delay, TBOR, ensures the voltage regulator output becomes stable.

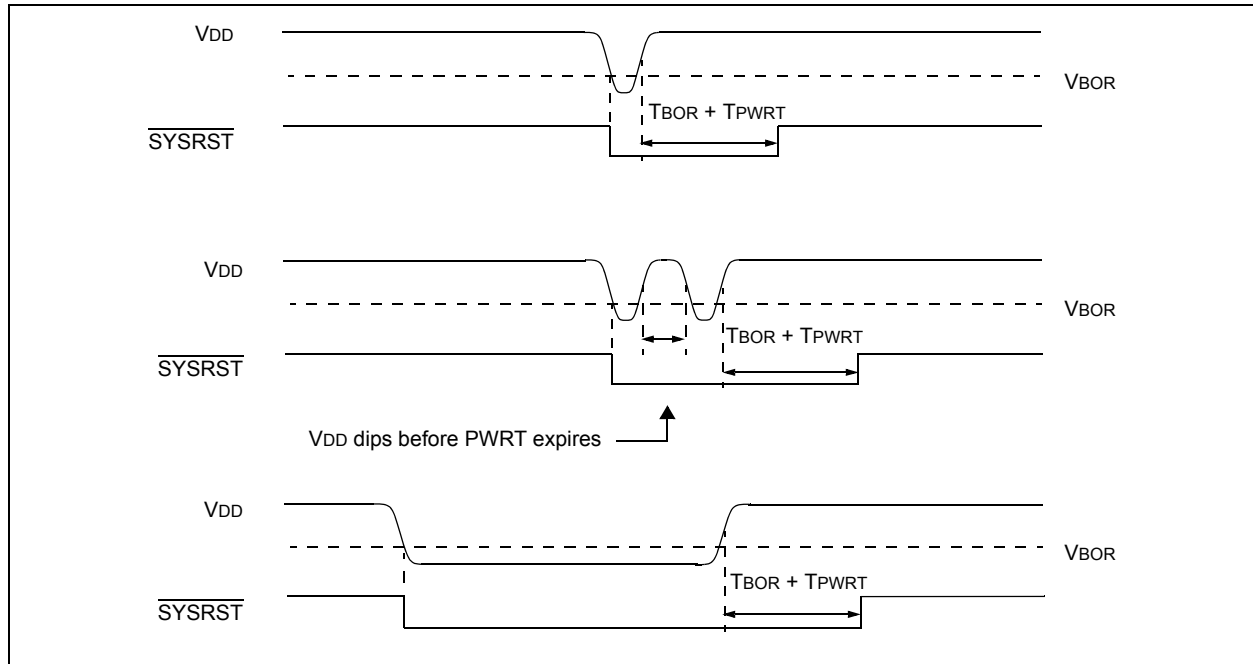
The BOR Status (BOR) bit in the Reset Control (RCON<1>) register is set to indicate the Brown-out Reset.

The device will not run at full speed after a BOR as the VDD should rise to acceptable levels for full-speed operation. The PWRT provides power-up time delay (TPWRT) to ensure that the system power supplies have stabilized at the appropriate levels for full-speed operation before the SYSRST is released.

The power-up timer delay (TPWRT) is programmed by the Power-on Reset Timer Value Select (FPWRT<2:0>) bits in the POR Configuration (FPOR<2:0>) register, which provides eight settings (from 0 ms to 128 ms). Refer to [Section 24.0 “Special Features”](#) for further details.

[Figure 6-3](#) shows the typical brown-out scenarios. The reset delay (TBOR + TPWRT) is initiated each time VDD rises above the  $V_{BOR}$  trip point

**FIGURE 6-3: BROWN-OUT SITUATIONS**



## 6.4 External Reset (EXTR)

The external Reset is generated by driving the  $\overline{\text{MCLR}}$  pin low. The  $\overline{\text{MCLR}}$  pin is a Schmitt Trigger input with an additional glitch filter. Reset pulses that are longer than the minimum pulse width will generate a Reset. Refer to [Section 27.0 “Electrical Characteristics”](#) for minimum pulse width specifications. The external Reset ( $\overline{\text{MCLR}}$ ) pin (EXTR) bit in the Reset Control (RCON) register is set to indicate the  $\overline{\text{MCLR}}$  Reset.

### 6.4.1 EXTERNAL SUPERVISORY CIRCUIT

Many systems have external supervisory circuits that generate Reset signals to reset multiple devices in the system. This external Reset signal can be directly connected to the  $\overline{\text{MCLR}}$  pin to reset the device when the rest of system is reset.

### 6.4.2 INTERNAL SUPERVISORY CIRCUIT

When using the internal power supervisory circuit to reset the device, the external Reset pin ( $\overline{\text{MCLR}}$ ) should be tied directly or resistively to VDD. In this case, the  $\overline{\text{MCLR}}$  pin will not be used to generate a Reset. The external Reset pin ( $\overline{\text{MCLR}}$ ) does not have an internal pull-up and must not be left unconnected.

## 6.5 Software RESET Instruction (SWR)

Whenever the `RESET` instruction is executed, the device will assert `SYSRST`, placing the device in a special Reset state. This Reset state will not re-initialize the clock. The clock source in effect prior to the `RESET` instruction will remain. `SYSRST` is released at the next instruction cycle and the Reset vector fetch will commence.

The Software Reset (SWR) flag (instruction) in the Reset Control (RCON<6>) register is set to indicate the software Reset.

## 6.6 Watchdog Time-out Reset (WDTO)

Whenever a Watchdog time-out occurs, the device will asynchronously assert `SYSRST`. The clock source will remain unchanged. A WDT time-out during Sleep or Idle mode will wake-up the processor, but will not reset the processor.

The Watchdog Timer Time-out (WDTO) flag in the Reset Control (RCON<4>) register is set to indicate the Watchdog Reset. Refer to [Section 24.4 “Watchdog Timer \(WDT\)”](#) for more information on Watchdog Reset.

## 6.7 Trap Conflict Reset

If a lower priority hard trap occurs while a higher priority trap is being processed, a hard Trap Conflict Reset occurs. The hard traps include exceptions of pri-

ority level 13 through level 15, inclusive. The address error (level 13) and oscillator error (level 14) traps fall into this category.

The Trap Reset (TRAPR) flag in the Reset Control (RCON<15>) register is set to indicate the Trap Conflict Reset. Refer to [Section 7.0 “Interrupt Controller”](#) for more information on Trap Conflict Resets.

## 6.8 Illegal Condition Device Reset

An illegal condition device Reset occurs due to the following sources:

- Illegal Opcode Reset
- Uninitialized W Register Reset
- Security Reset

The Illegal Opcode or Uninitialized W Access Reset (IOPUWR) flag in the Reset Control (RCON<14>) register is set to indicate the illegal condition device Reset.

### 6.8.1 ILLEGAL OPCODE RESET

A device Reset is generated if the device attempts to execute an illegal opcode value that is fetched from program memory.

The Illegal Opcode Reset function can prevent the device from executing program memory sections that are used to store constant data. To take advantage of the Illegal Opcode Reset, use only the lower 16 bits of each program memory section to store the data values. The upper 8 bits should be programmed with 3Fh, which is an illegal opcode value.

### 6.8.2 UNINITIALIZED W REGISTER RESET

Any attempt to use the uninitialized W register as an Address Pointer will Reset the device. The W register array (with the exception of W15) is cleared during all Resets and is considered uninitialized until written to.

### 6.8.3 SECURITY RESET

If a Program Flow Change (PFC) or Vector Flow Change (VFC) targets a restricted location in a protected segment (boot and secure segment), that operation will cause a Security Reset.

The PFC occurs when the program counter is reloaded as a result of a call, jump, computed jump, return, return from subroutine or other form of branch instruction.

The VFC occurs when the program counter is reloaded with an interrupt or trap vector.

Refer to [Section 24.8 “Code Protection and CodeGuard™ Security”](#) for more information on Security Reset.



**6.9 Using the RCON Status Bits**

Table 6-2 provides a summary of the Reset flag bit operation.

The user application can read the Reset Control (RCON) register after any device Reset to determine the cause of the Reset.

**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.

**TABLE 6-2: RESET FLAG BIT OPERATION**

Flag Bit	Set by:	Cleared by:
<b>TRAPR</b> (RCON<15>)	Trap conflict event	POR,BOR
<b>IOPWR</b> (RCON<14>)	Illegal opcode or uninitialized W register access or Security Reset	POR,BOR
<b>EXTR</b> (RCON<7>)	MCLR Reset	POR
<b>SWR</b> (RCON<6>)	RESET instruction	POR,BOR
<b>WDTO</b> (RCON<4>)	WDT time-out	PWRSV instruction, CLRWDT instruction, POR,BOR
<b>SLEEP</b> (RCON<3>)	PWRSV #SLEEP instruction	POR,BOR
<b>IDLE</b> (RCON<2>)	PWRSV #IDLE instruction	POR,BOR
<b>BOR</b> (RCON<1>)	POR, BOR	—
<b>POR</b> (RCON<0>)	POR	—

**Note:** All Reset flag bits can be set or cleared by user software.

NOTES:

## 7.0 INTERRUPT CONTROLLER

**Note 1:** This data sheet summarizes the features of the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 families of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 47. “Interrupts (Part V)”** (DS70597) in the “*dsPIC33F/PIC24H Family Reference Manual*”, which is available from the Microchip web site ([www.microchip.com](http://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 dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 interrupt controller reduces the numerous peripheral interrupt request signals to a single interrupt request signal to the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 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. Lower addresses generally have a higher natural priority. For example, the interrupt associated with vector 0 will take priority over interrupts at any other vector address.

The dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 devices implement up to 71 unique interrupts and five non-maskable traps. These are summarized in [Table 7-1](#).

#### 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.

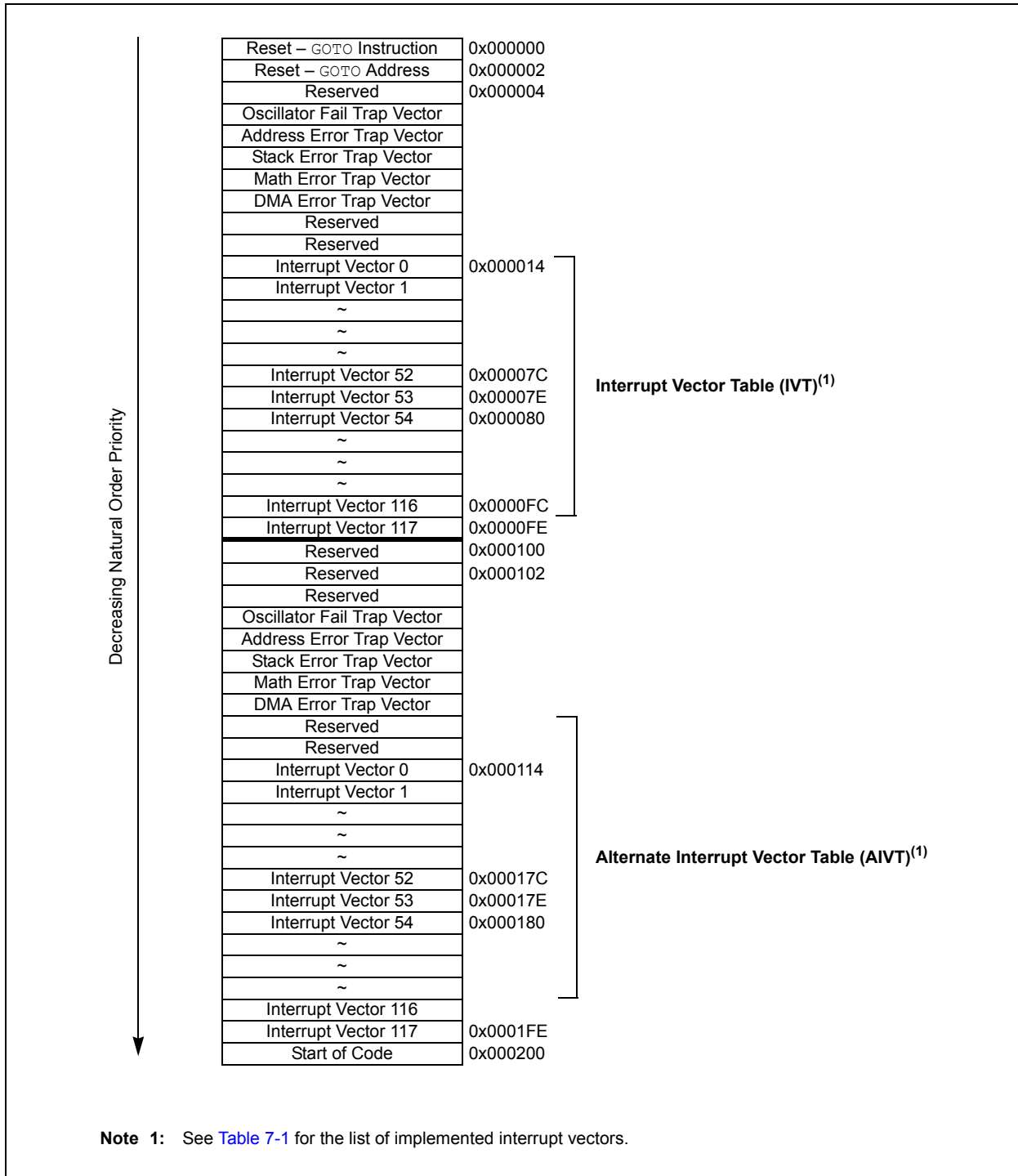
The AIVT supports debugging by providing a means to switch between an application and a support environment without requiring the interrupt vectors to be reprogrammed. This feature also enables switching between applications for evaluation of different software algorithms at run time. If the AIVT is 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 dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 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. A GOTO instruction at the Reset address can redirect 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.

**FIGURE 7-1: dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 INTERRUPT VECTOR TABLE**



**TABLE 7-1: INTERRUPT VECTORS**

Vector Number	Interrupt Request (IQR)	IVT Address	AIVT Address	Interrupt Source
Highest Natural Order Priority				
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 Fault
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	ADC – ADC Group Convert Done
22	14	0x000030	0x000130	DMA1 – DMA Channel 1
23	15	0x000032	0x000132	Reserved
24	16	0x000034	0x000134	SI2C1 – I2C1 Slave Event
25	17	0x000036	0x000136	MI2C1 – I2C1 Master Event
26	18	0x000038	0x000138	CMP1 – Analog Comparator 1 Interrupt
27	19	0x00003A	0x00013A	CN – Input Change Notification Interrupt
28	20	0x00003C	0x00013C	INT1 – External Interrupt 1
29-31	21-23	0x00003E- 0x000042	0x00013E- 0x000142	Reserved
32	24	0x000044	0x000144	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	SPI2 – SPI2 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
47-56	39-48	0x000062- 0x000074	0x000162- 0x000174	Reserved
57	49	0x000076	0x000176	SI2C2 – I2C2 Slave Events
58	50	0x000078	0x000178	MI2C2 – I2C2 Master Events
59-60	51-52	0x00007A- 0x00007C	0x00017A- 0x00017C	Reserved
61	53	0x00007E	0x00017E	INT3 – External Interrupt 3
62	54	0x000080	0x000180	INT4 – External Interrupt 4

**TABLE 7-1: INTERRUPT VECTORS (CONTINUED)**

Vector Number	Interrupt Request (IQR)	IVT Address	AIVT Address	Interrupt Source
63-64	55-56	0x000082-0x000084	0x000182-0x000184	Reserved
65	57	0x000086	0x000186	PWM PSEM Special Event Match
66	58	0x000088	0x000188	QE1 – Position Counter Compare
67-72	59-64	0x00008A-0x000094	0x00018A-0x000194	Reserved
73	65	0x000096	0x000196	U1E – UART1 Error Interrupt
74	66	0x000098	0x000198	U2E – UART2 Error Interrupt
75-77	67-69	0x00009A-0x00009E	0x00019A-0x00019E	Reserved
78	70	0x0000A0	0x0001A0	C1TX – ECAN1 Transmit Data Request
79	71	0x0000A2	0x0001A2	Reserved
80	72	0x0000A4	0x0001A4	Reserved
81	73	0x0000A6	0x0001A6	PWM Secondary Special Event Match
82	74	0x0000A8	0x0001A8	Reserved
83	75	0x0000AA	0x0001AA	QE2 – Position Counter Compare
84-88	76-80	0x0000AC-0x0000B4	0x0001AC-0x0001B4	Reserved
89	81	0x0000B6	0x0001B6	ADC Pair 8 Conversion Done
90	82	0x0000B8	0x0001B8	ADC Pair 9 Conversion Done
91	83	0x0000BA	0x0001BA	ADC Pair 10 Conversion Done
92	84	0x0000BC	0x0001BC	ADC Pair 11 Conversion Done
93	85	0x0000BE	0x0001BE	ADC Pair 12 Conversion Done
94-101	86-93	0x0000C0-0x0000CE	0x0001C0-0x0001CE	Reserved
102	94	0x0000D0	0x0001D0	PWM1 – PWM1 Interrupt
103	95	0x0000D2	0x0001D2	PWM2 – PWM2 Interrupt
104	96	0x0000D4	0x0001D4	PWM3 – PWM3 Interrupt
105	97	0x0000D6	0x0001D6	PWM4 – PWM4 Interrupt
106	98	0x0000D8	0x0001D8	PWM5 – PWM5 Interrupt
107	99	0x0000DA	0x0001DA	PWM6 – PWM6 Interrupt
108	100	0x0000DC	0x0001DC	PWM7 – PWM7 Interrupt
109	101	0x0000DE	0x0001DE	PWM8 – PWM8 Interrupt
110	102	0x0000E0	0x0001E0	PWM9 – PWM9 Interrupt
111	103	0x0000E2	0x0001E2	CMP2 – Analog Comparator 2
112	104	0x0000E4	0x0001E4	CMP3 – Analog Comparator 3
113	105	0x0000E6	0x0001E6	CMP4 – Analog Comparator 4
114-117	106-109	0x0000E8-0x0000EE	0x0001E8-0x0001EE	Reserved
118	110	0x0000F0	0x0001F0	ADC Pair 0 Convert Done
119	111	0x0000F2	0x0001F2	ADC Pair 1 Convert Done
120	112	0x0000F4	0x0001F4	ADC Pair 2 Convert Done
121	113	0x0000F6	0x0001F6	ADC Pair 3 Convert Done
122	114	0x0000F8	0x0001F8	ADC Pair 4 Convert Done
123	115	0x0000FA	0x0001FA	ADC Pair 5 Convert Done
124	116	0x0000FC	0x0001FC	ADC Pair 6 Convert Done
125	117	0x0000FE	0x0001FE	ADC Pair 7 Convert Done
Lowest Natural Order Priority				

## 7.3 Interrupt Control and Status Registers

The dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 devices implement 27 registers for the interrupt controller:

- INTCON1
- INTCON2
- IFSx
- IECx
- IPCx
- INTTREG

### 7.3.1 INTCON1 AND INTCON2

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

### 7.3.2 IFSx

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

### 7.3.3 IECx

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

### 7.3.4 IPCx

The IPCx 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.

### 7.3.5 INTTREG

The INTTREG register contains the associated interrupt vector number and the new CPU Interrupt priority Level, which are latched into the Vector Number (VECNUM<6:0>) and Interrupt Level (ILR<3:0>) bit fields in the INTTREG register. The new Interrupt Priority Level is the priority of the pending interrupt.

The interrupt sources are assigned to the IFSx, IECx and IPCx registers in the 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 is found in IEC0<0> and the INT0IP bits are found in the first position of IPC0 (IPC0<2:0>).

### 7.3.6 STATUS/CONTROL REGISTERS

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>, 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 7-46](#) in the following pages.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## 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	OB	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	C
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](#).

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

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](#).

- 2: The IPL3 bit is concatenated with the IPL<2:0> bits (SR<7:5>) to form the CPU Interrupt Priority Level.



# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## 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  
-n = Value at POR

W = Writable bit  
'1' = Bit is set

U = Unimplemented bit, read as '0'  
'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     **OVATE:** Accumulator A Overflow Trap Enable bit  
1 = Trap overflow of Accumulator A  
0 = Trap disabled
- bit 9      **OVBTE:** 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      **DIV0ERR:** 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  
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

## 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 occurred  
          0 = 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 vector table  
                   0 = 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 edge  
                   0 = Interrupt on positive edge
- bit 3            **INT3EP:** External Interrupt 3 Edge Detect Polarity Select bit  
                   1 = Interrupt on negative edge  
                   0 = Interrupt on positive edge
- bit 2            **INT2EP:** External Interrupt 2 Edge Detect Polarity Select bit  
                   1 = Interrupt on negative edge  
                   0 = Interrupt on positive edge
- bit 1            **INT1EP:** External Interrupt 1 Edge Detect Polarity Select bit  
                   1 = Interrupt on negative edge  
                   0 = Interrupt on positive edge
- bit 0            **INT0EP:** External Interrupt 0 Edge Detect Polarity Select bit  
                   1 = Interrupt on negative edge  
                   0 = Interrupt on positive edge

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## 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	ADIF	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	DMA0IF	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 occurred  
0 = Interrupt request has not occurred
- bit 13      **ADIF:** ADC Group Conversion Complete Interrupt Flag Status bit  
1 = Interrupt request has occurred  
0 = Interrupt request has not occurred
- bit 12      **U1TXIF:** UART1 Transmitter Interrupt Flag Status bit  
1 = Interrupt request has occurred  
0 = Interrupt request has not occurred
- bit 11      **U1RXIF:** UART1 Receiver Interrupt Flag Status bit  
1 = Interrupt request has occurred  
0 = Interrupt request has not occurred
- bit 10      **SPI1IF:** SPI1 Event Interrupt Flag Status bit  
1 = Interrupt request has occurred  
0 = 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 occurred  
0 = Interrupt request has not occurred
- bit 7        **T2IF:** Timer2 Interrupt Flag Status bit  
1 = Interrupt request has occurred  
0 = Interrupt request has not occurred
- bit 6        **OC2IF:** Output Compare Channel 2 Interrupt Flag Status bit  
1 = Interrupt request has occurred  
0 = Interrupt request has not occurred
- bit 5        **IC2IF:** Input Capture Channel 2 Interrupt Flag Status bit  
1 = Interrupt request has occurred  
0 = Interrupt request has not occurred
- bit 4        **DMA0IF:** DMA Channel 0 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  
1 = Interrupt request has occurred  
0 = Interrupt request has not occurred

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

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## 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	DMA2IF
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	INT1IF	CNIF	AC1IF	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 12            **U2TXIF:** UART2 Transmitter Interrupt Flag Status bit  
                   1 = Interrupt request has occurred  
                   0 = Interrupt request has not occurred
- bit 11            **U2RXIF:** UART2 Receiver Interrupt Flag Status bit  
                   1 = Interrupt request has occurred  
                   0 = Interrupt request has not occurred
- bit 13            **INT2IF:** External Interrupt 2 Flag Status bit  
                   1 = Interrupt request has occurred  
                   0 = Interrupt request has not occurred
- bit 12            **T5IF:** Timer5 Interrupt Flag Status bit  
                   1 = Interrupt request has occurred  
                   0 = Interrupt request has not occurred
- bit 11            **T4IF:** Timer4 Interrupt Flag Status bit  
                   1 = Interrupt request has occurred  
                   0 = Interrupt request has not occurred
- bit 10            **OC4IF:** Output Compare Channel 4 Interrupt Flag Status bit  
                   1 = Interrupt request has occurred  
                   0 = Interrupt request has not occurred
- bit 9             **OC3IF:** Output Compare Channel 3 Interrupt Flag Status bit  
                   1 = Interrupt request has occurred  
                   0 = Interrupt request has not occurred
- bit 8             **DMA2IF:** DMA Channel 2 Data Transfer Complete Interrupt Flag Status bit  
                   1 = Interrupt request has occurred  
                   0 = Interrupt request has not occurred
- bit 7-5           **Unimplemented:** Read as '0'
- bit 4             **INT1IF:** External Interrupt 1 Flag Status bit  
                   1 = Interrupt request has occurred  
                   0 = Interrupt request has not occurred
- bit 3             **CNIF:** Input Change Notification Interrupt Flag Status bit  
                   1 = Interrupt request has occurred  
                   0 = Interrupt request has not occurred
- bit 2             **AC1IF:** Analog Comparator 1 Interrupt Flag Status bit  
                   1 = Interrupt request has occurred  
                   0 = Interrupt request has not occurred
- bit 1             **MI2C1IF:** I2C1 Master Events Interrupt Flag Status bit  
                   1 = Interrupt request has occurred  
                   0 = Interrupt request has not occurred
- bit 0             **SI2C1IF:** I2C1 Slave Events Interrupt Flag Status bit  
                   1 = Interrupt request has occurred  
                   0 = Interrupt request has not occurred

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

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	IC4IF	IC3IF	DMA3IF	C1IF <sup>(1)</sup>	C1EIF <sup>(1)</sup>	SPI2IF	SPI2EIF
bit 7							bit 0

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

- bit 15-7      **Unimplemented:** Read as '0'
- bit 6        **IC4IF:** Input Capture Channel 4 Interrupt Flag Status bit
  - 1 = Interrupt request has occurred
  - 0 = Interrupt request has not occurred
- bit 5        **IC3IF:** Input Capture Channel 3 Interrupt Flag Status bit
  - 1 = Interrupt request has occurred
  - 0 = Interrupt request has not occurred
- bit 4        **DMA3IF:** DMA Channel 3 Data Transfer Complete Interrupt Flag Status bit
  - 1 = Interrupt request has occurred
  - 0 = Interrupt request has not occurred
- bit 3        **C1IF:** ECAN1 Event Interrupt Flag Status bit<sup>(1)</sup>
  - 1 = Interrupt request has occurred
  - 0 = Interrupt request has not occurred
- bit 2        **C1EIF:** ECAN1 External Event Interrupt Flag Status bit<sup>(1)</sup>
  - 1 = Interrupt request has occurred
  - 0 = Interrupt request has not occurred
- bit 1        **SPI2IF:** SPI2 Event Interrupt Flag Status bit
  - 1 = Interrupt request has occurred
  - 0 = Interrupt request has not occurred
- bit 0        **SPI2EIF:** SPI2 Error Interrupt Flag Status bit
  - 1 = Interrupt request has occurred
  - 0 = Interrupt request has not occurred

**Note 1:** Interrupts disabled on devices without ECAN™ modules.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

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

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	U-0
—	—	—	—	—	QE11IF	PSEMIF	—
bit 15							bit 8

U-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	U-0
—	INT4IF	INT3IF	—	—	MI2C2IF	SI2C2IF	—
bit 7							bit 0

### Legend:

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

- bit 15-11      **Unimplemented:** Read as '0'
- bit 10        **QE11IF:** QE11 Event Interrupt Flag Status bit  
                  1 = Interrupt request has occurred  
                  0 = Interrupt request has not occurred
- bit 9         **PSEMIF:** PWM Special Event Match Interrupt Flag Status bit  
                  1 = Interrupt request has occurred  
                  0 = Interrupt request has not occurred
- bit 8-7      **Unimplemented:** Read as '0'
- bit 6         **INT4IF:** External Interrupt 4 Flag Status bit  
                  1 = Interrupt request has occurred  
                  0 = 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 4-3      **Unimplemented:** Read as '0'
- bit 2         **MI2C2IF:** I2C2 Master Events Interrupt Flag Status bit  
                  1 = Interrupt request has occurred  
                  0 = Interrupt request has not occurred
- bit 1         **SI2C2IF:** I2C2 Slave Events Interrupt Flag Status bit  
                  1 = Interrupt request has occurred  
                  0 = Interrupt request has not occurred
- bit 0         **Unimplemented:** Read as '0'



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

U-0	U-0	U-0	U-0	R/W-0	U-0	R/W-0	U-0
—	—	—	—	QE12IF	—	PSESMIF	—
bit 15							bit 8

U-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	U-0
—	C1TXIF <sup>(1)</sup>	—	—	—	U2EIF	U1EIF	—
bit 7							bit 0

<b>Legend:</b>			
R = Readable bit	W = Writable bit	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       **QE12IF:** QE12 Event Interrupt Flag Status bit  
               1 = Interrupt request has occurred  
               0 = Interrupt request has not occurred
- bit 10       **Unimplemented:** Read as '0'
- bit 9         **PSESMIF:** PWM Special Event Secondary Match Interrupt Flag Status bit  
               1 = Interrupt request has occurred  
               0 = Interrupt request has not occurred
- bit 8-7      **Unimplemented:** Read as '0'
- bit 6         **C1TXIF:** ECAN1 Transmit Data Request Interrupt Flag Status bit<sup>(1)</sup>  
               1 = Interrupt request has occurred  
               0 = Interrupt request has not occurred
- bit 5-3      **Unimplemented:** Read as '0'
- bit 2         **U2EIF:** UART2 Error Interrupt Flag Status bit  
               1 = Interrupt request has occurred  
               0 = Interrupt request has not occurred
- bit 1         **U1EIF:** UART1 Error Interrupt Flag Status bit  
               1 = Interrupt request has occurred  
               0 = Interrupt request has not occurred
- bit 0         **Unimplemented:** Read as '0'

**Note 1:** Interrupts disabled on devices without ECAN™ modules.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 7-10: IFS5: INTERRUPT FLAG STATUS REGISTER 5

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0
PWM2IF	PWM1IF	ADCP12IF	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0
—	—	—	ADCP11IF	ADCP10IF	ADCP9IF	ADCP8IF	—
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            **PWM2IF:** PWM2 Interrupt Flag Status bit  
                   1 = Interrupt request has occurred  
                   0 = Interrupt request has not occurred
- bit 14            **PWM1IF:** PWM1 Interrupt Flag Status bit  
                   1 = Interrupt request has occurred  
                   0 = Interrupt request has not occurred
- bit 13            **ADCP12IF:** ADC Pair 12 Conversion Done Interrupt Flag Status bit  
                   1 = Interrupt request has occurred  
                   0 = Interrupt request has not occurred
- bit 12-5        **Unimplemented:** Read as '0'
- bit 4             **ADCP11IF:** ADC Pair 11 Conversion Done Interrupt Flag Status bit  
                   1 = Interrupt request has occurred  
                   0 = Interrupt request has not occurred
- bit 3             **ADCP10IF:** ADC Pair 10 Conversion Done Interrupt Flag Status bit  
                   1 = Interrupt request has occurred  
                   0 = Interrupt request has not occurred
- bit 2             **ADCP9IF:** ADC Pair 9 Conversion Done Interrupt Flag Status bit  
                   1 = Interrupt request has occurred  
                   0 = Interrupt request has not occurred
- bit 1             **ADCP8IF:** ADC Pair 8 Conversion Done Interrupt Flag Status bit  
                   1 = Interrupt request has occurred  
                   0 = Interrupt request has not occurred
- bit 0             **Unimplemented:** Read as '0'

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 7-11: IFS6: INTERRUPT FLAG STATUS REGISTER 6

R/W-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
ADCP1IF	ADCP0IF	—	—	—	—	AC4IF	AC3IF
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
AC2IF	PWM9IF	PWM8IF	PWM7IF	PWM6IF	PWM5IF	PWM4IF	PWM3IF
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        **ADCP1IF:** ADC Pair 1 Conversion Done Interrupt Flag Status bit  
                  1 = Interrupt request has occurred  
                  0 = Interrupt request has not occurred
- bit 14        **ADCP0IF:** ADC Pair 0 Conversion Done Interrupt Flag Status bit  
                  1 = Interrupt request has occurred  
                  0 = Interrupt request has not occurred
- bit 13-10     **Unimplemented:** Read as '0'
- bit 9         **AC4IF:** Analog Comparator 4 Interrupt Flag Status bit  
                  1 = Interrupt request has occurred  
                  0 = Interrupt request has not occurred
- bit 8         **AC3IF:** Analog Comparator 3 Interrupt Flag Status bit  
                  1 = Interrupt request has occurred  
                  0 = Interrupt request has not occurred
- bit 7         **AC2IF:** Analog Comparator 2 Interrupt Flag Status bit  
                  1 = Interrupt request has occurred  
                  0 = Interrupt request has not occurred
- bit 6         **PWM9IF:** PWM9 Interrupt Flag Status bit  
                  1 = Interrupt request has occurred  
                  0 = Interrupt request has not occurred
- bit 5         **PWM8IF:** PWM8 Interrupt Flag Status bit  
                  1 = Interrupt request has occurred  
                  0 = Interrupt request has not occurred
- bit 4         **PWM7IF:** PWM7 Interrupt Flag Status bit  
                  1 = Interrupt request has occurred  
                  0 = Interrupt request has not occurred
- bit 3         **PWM6IF:** PWM6 Interrupt Flag Status bit  
                  1 = Interrupt request has occurred  
                  0 = Interrupt request has not occurred
- bit 2         **PWM5IF:** PWM5 Interrupt Flag Status bit  
                  1 = Interrupt request has occurred  
                  0 = Interrupt request has not occurred
- bit 1         **PWM4IF:** PWM4 Interrupt Flag Status bit  
                  1 = Interrupt request has occurred  
                  0 = Interrupt request has not occurred
- bit 0         **PWM3IF:** PWM3 Interrupt Flag Status bit  
                  1 = Interrupt request has occurred  
                  0 = Interrupt request has not occurred

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

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

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	ADCP7IF	ADCP6IF	ADCP5IF	ADCP4IF	ADCP3IF	ADCP2IF
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        **ADCP7IF:** ADC Pair 7 Conversion Done Interrupt Flag Status bit  
               1 = Interrupt request has occurred  
               0 = Interrupt request has not occurred
- bit 4        **ADCP6IF:** ADC Pair 6 Conversion Done Interrupt Flag Status bit  
               1 = Interrupt request has occurred  
               0 = Interrupt request has not occurred
- bit 3        **ADCP5IF:** ADC Pair 5 Conversion Done Interrupt Flag Status bit  
               1 = Interrupt request has occurred  
               0 = Interrupt request has not occurred
- bit 2        **ADCP4IF:** ADC Pair 4 Conversion Done Interrupt Flag Status bit  
               1 = Interrupt request has occurred  
               0 = Interrupt request has not occurred
- bit 1        **ADCP3IF:** ADC Pair 3 Conversion Done Interrupt Flag Status bit  
               1 = Interrupt request has occurred  
               0 = Interrupt request has not occurred
- bit 0        **ADCP2IF:** ADC Pair 2 Conversion Done Interrupt Flag Status bit  
               1 = Interrupt request has occurred  
               0 = Interrupt request has not occurred

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 7-13: IEC0: INTERRUPT ENABLE CONTROL REGISTER 0

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	DMA1IE	ADIE	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:

R = Readable bit	W = Writable bit	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 enabled  
0 = Interrupt request not enabled
- bit 13      **ADIE:** ADC1 Conversion Complete Interrupt Enable bit  
1 = Interrupt request enabled  
0 = Interrupt request not enabled
- bit 12      **U1TXIE:** UART1 Transmitter Interrupt Enable bit  
1 = Interrupt request enabled  
0 = Interrupt request not enabled
- bit 11      **U1RXIE:** UART1 Receiver Interrupt Enable bit  
1 = Interrupt request enabled  
0 = Interrupt request not enabled
- bit 10      **SPI1IE:** SPI1 Event Interrupt Enable bit  
1 = Interrupt request enabled  
0 = Interrupt request not enabled
- bit 9        **SPI1EIE:** SPI1 Event Interrupt Enable bit  
1 = Interrupt request enabled  
0 = Interrupt request not enabled
- bit 8        **T3IE:** Timer3 Interrupt Enable bit  
1 = Interrupt request enabled  
0 = Interrupt request not enabled
- bit 7        **T2IE:** Timer2 Interrupt Enable bit  
1 = Interrupt request enabled  
0 = Interrupt request not enabled
- bit 6        **OC2IE:** Output Compare Channel 2 Interrupt Enable bit  
1 = Interrupt request enabled  
0 = Interrupt request not enabled
- bit 5        **IC2IE:** Input Capture Channel 2 Interrupt Enable bit  
1 = Interrupt request enabled  
0 = Interrupt request not enabled
- bit 4        **DMA0IE:** DMA Channel 0 Data Transfer Complete Interrupt Enable bit  
1 = Interrupt request enabled  
0 = Interrupt request not enabled
- bit 3        **T1IE:** Timer1 Interrupt Enable bit  
1 = Interrupt request enabled  
0 = Interrupt request not enabled

## REGISTER 7-13: IEC0: INTERRUPT ENABLE CONTROL REGISTER 0 (CONTINUED)

- bit 2      **OC1IE:** Output Compare Channel 1 Interrupt Enable bit  
            1 = Interrupt request enabled  
            0 = Interrupt request not enabled
- bit 1      **IC1IE:** Input Capture Channel 1 Interrupt Enable bit  
            1 = Interrupt request enabled  
            0 = Interrupt request not enabled
- bit 0      **INT0IE:** External Interrupt 0 Enable bit  
            1 = Interrupt request enabled  
            0 = Interrupt request not enabled

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 7-14: IEC1: INTERRUPT ENABLE CONTROL REGISTER 1

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

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	INT1IE	CNIE	AC1IE	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 12            **U2TXIE:** UART2 Transmitter Interrupt Enable bit  
1 = Interrupt request enabled  
0 = Interrupt request not enabled
- bit 11            **U2RXIE:** UART2 Receiver Interrupt Enable bit  
1 = Interrupt request enabled  
0 = Interrupt request not enabled
- bit 13            **INT2IE:** External Interrupt 2 Enable bit  
1 = Interrupt request enabled  
0 = 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 enabled  
0 = Interrupt request not enabled
- bit 9             **OC3IE:** Output Compare Channel 3 Interrupt Enable bit  
1 = Interrupt request enabled  
0 = Interrupt request not enabled
- bit 8             **DMA2IE:** DMA Channel 2 Data Transfer Complete Interrupt Enable bit  
1 = Interrupt request enabled  
0 = Interrupt request not enabled
- bit 7-5           **Unimplemented:** Read as '0'
- bit 4             **INT1IE:** External Interrupt 1 Enable bit  
1 = Interrupt request enabled  
0 = Interrupt request not enabled
- bit 3             **CNIE:** Input Change Notification Interrupt Enable bit  
1 = Interrupt request enabled  
0 = Interrupt request not enabled
- bit 2             **AC1IE:** Analog Comparator 1 Interrupt Enable bit  
1 = Interrupt request enabled  
0 = Interrupt request not enabled
- bit 1             **MI2C1IE:** I2C1 Master Events Interrupt Enable bit  
1 = Interrupt request enabled  
0 = Interrupt request not enabled
- bit 0             **SI2C1IE:** I2C1 Slave Events Interrupt Enable bit  
1 = Interrupt request enabled  
0 = Interrupt request not enabled

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 7-15: IEC2: INTERRUPT ENABLE 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	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	IC4IE	IC3IE	DMA3IE	C1IE <sup>(1)</sup>	C1RXIE <sup>(1)</sup>	SPI2IE	SPI2EIE
bit 7						bit 0	

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

- bit 15-7      **Unimplemented:** Read as '0'
- bit 6        **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 has enabled
- bit 3        **C1IE:** ECAN1 Event Interrupt Enable bit<sup>(1)</sup>  
1 = Interrupt request enabled  
0 = Interrupt request not enabled
- bit 2        **C1RXIE:** ECAN1 Receive Data Ready Interrupt Enable bit<sup>(1)</sup>  
1 = Interrupt request enabled  
0 = Interrupt request not enabled
- bit 1        **SPI2IE:** SPI2 Event Interrupt Enable bit  
1 = Interrupt request enabled  
0 = Interrupt request not enabled
- bit 0        **SPI2EIE:** SPI2 Error Interrupt Enable bit  
1 = Interrupt request enabled  
0 = Interrupt request not enabled

**Note 1:** Interrupts disabled on devices without ECAN™ modules.



**REGISTER 7-16: IEC3: INTERRUPT ENABLE CONTROL REGISTER 3**

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	U-0
—	—	—	—	—	QE11IE	PSEMIE	—
bit 15						bit 8	

U-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	U-0
—	INT4IE	INT3IE	—	—	MI2C2IE	SI2C2IE	—
bit 7						bit 0	

**Legend:**

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

- bit 15-11      **Unimplemented:** Read as '0'
- bit 10      **QE11IE:** QE11 Event Interrupt Enable bit  
 1 = Interrupt request enabled  
 0 = Interrupt request not enabled
- bit 9      **PSEMIE:** PWM Special Event Match Interrupt Enable bit  
 1 = Interrupt request enabled  
 0 = Interrupt request not enabled
- bit 8-7      **Unimplemented:** Read as '0'
- bit 6      **INT4IE:** External Interrupt 4 Enable bit  
 1 = Interrupt request enabled  
 0 = Interrupt request not enabled
- bit 6      **INT3IE:** External Interrupt 3 Enable bit  
 1 = Interrupt request enabled  
 0 = Interrupt request not enabled
- bit 4-3      **Unimplemented:** Read as '0'
- bit 2      **MI2C2IE:** I2C2 Master Events Interrupt Enable bit  
 1 = Interrupt request enabled  
 0 = Interrupt request not enabled
- bit 1      **SI2C2IE:** I2C2 Slave Events Interrupt Enable bit  
 1 = Interrupt request enabled  
 0 = Interrupt request not enabled
- bit 0      **Unimplemented:** Read as '0'

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 7-17: IEC4: INTERRUPT ENABLE CONTROL REGISTER 4

U-0	U-0	U-0	U-0	R/W-0	U-0	R/W-0	U-0
—	—	—	—	QEI2IE	—	PSESMIE	—
bit 15							bit 8

U-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	U-0
—	C1TXIE <sup>(1)</sup>	—	—	—	U2EIE	U1EIE	—
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 **QEI2IE:** QEI2 Event Interrupt Enable bit

1 = Interrupt request enabled

0 = Interrupt request not enabled

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

bit 9 **PSESMIE:** PWM Special Event Secondary Match Error Interrupt Enable bit

1 = Interrupt request enabled

0 = Interrupt request not enabled

bit 8-7 **Unimplemented:** Read as '0'

bit 6 **C1TXIE:** ECAN1 Transmit Data Request Interrupt Enable bit<sup>(1)</sup>

1 = Interrupt request occurred

0 = Interrupt request not occurred

bit 5-3 **Unimplemented:** Read as '0'

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

1 = Interrupt request enabled

0 = Interrupt request not enabled

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

1 = Interrupt request enabled

0 = Interrupt request not enabled

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

**Note 1:** Interrupts disabled on devices without ECAN™ modules.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 7-18: IEC5: INTERRUPT ENABLE CONTROL REGISTER 5

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0
PWM2IE	PWM1IE	ADCP12IE	—	—	—	—	—
bit 15						bit 8	

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0
—	—	—	ADCP11IE	ADCP10IE	ADCP9IE	ADCP8IE	—
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      **PWM2IE:** PWM2 Interrupt Enable bit<sup>(1)</sup>  
1 = Interrupt request is enabled  
0 = Interrupt request is not enabled
- bit 14      **PWM1IE:** PWM1 Interrupt Enable bit  
1 = Interrupt request is enabled  
0 = Interrupt request is not enabled
- bit 13      **ADCP12IE:** ADC Pair 12 Conversion Done Interrupt Enable bit  
1 = Interrupt request is enabled  
0 = Interrupt request is not enabled
- bit 12-5    **Unimplemented:** Read as '0'
- bit 4        **ADCP11IE:** ADC Pair 11 Conversion Done Interrupt Enable bit  
1 = Interrupt request is enabled  
0 = Interrupt request is not enabled
- bit 3        **ADCP10IE:** ADC Pair 10 Conversion Done Interrupt Enable bit  
1 = Interrupt request is enabled  
0 = Interrupt request is not enabled
- bit 2        **ADCP9IE:** ADC Pair 9 Conversion Done Interrupt Enable bit  
1 = Interrupt request is enabled  
0 = Interrupt request is not enabled
- bit 1        **ADCP8IE:** ADC Pair 8 Conversion Done Interrupt Enable bit  
1 = Interrupt request is enabled  
0 = Interrupt request is not enabled
- bit 0        **Unimplemented:** Read as '0'

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 7-19: IEC6: INTERRUPT ENABLE CONTROL REGISTER 6

R/W-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
ADCP1IE	ADCP0IE	—	—	—	—	AC4IE	AC3IE
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
AC2IE	PWM9IE	PWM8IE	PWM7IE	PWM6IE	PWM5IE	PWM4IE	PWM3IE
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            **ADCP1IE:** ADC Pair 1 Conversion Done Interrupt Enable bit

1 = Interrupt request is enabled  
0 = Interrupt request is not enabled

bit 14            **ADCP0IE:** ADC Pair 0 Conversion Done Interrupt Enable bit

1 = Interrupt request is enabled  
0 = Interrupt request is not enabled

bit 13-10        **Unimplemented:** Read as '0'

bit 9             **AC4IE:** Analog Comparator 4 Interrupt Enable bit

1 = Interrupt request is enabled  
0 = Interrupt request is not enabled

bit 8             **AC3IE:** Analog Comparator 3 Interrupt Enable bit

1 = Interrupt request is enabled  
0 = Interrupt request is not enabled

bit 7             **AC2IE:** Analog Comparator 2 Interrupt Enable bit

1 = Interrupt request is enabled  
0 = Interrupt request is not enabled

bit 6             **PWM9IE:** PWM9 Interrupt Enable bit

1 = Interrupt request is enabled  
0 = Interrupt request is not enabled

bit 5             **PWM8IE:** PWM8 Interrupt Enable bit

1 = Interrupt request is enabled  
0 = Interrupt request is not enabled

bit 4             **PWM7IE:** PWM7 Interrupt Enable bit

1 = Interrupt request is enabled  
0 = Interrupt request is not enabled

bit 3             **PWM6IE:** PWM6 Interrupt Enable bit

1 = Interrupt request is enabled  
0 = Interrupt request is not enabled

bit 2             **PWM5IE:** PWM5 Interrupt Enable bit

1 = Interrupt request is enabled  
0 = Interrupt request is not enabled

bit 1             **PWM4IE:** PWM4 Interrupt Enable bit

1 = Interrupt request is enabled  
0 = Interrupt request is not enabled

bit 0             **PWM3IE:** PWM3 Interrupt Enable bit

1 = Interrupt request is enabled  
0 = Interrupt request is not enabled

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

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	ADCP7IE	ADCP6IE	ADCP5IE	ADCP4IE	ADCP3IE	ADCP2IE
bit 7							bit 0

<b>Legend:</b>			
R = Readable bit	W = Writable bit	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        **ADCP7IE:** ADC Pair 7 Conversion Done Interrupt Enable bit  
                  1 = Interrupt request is enabled  
                  0 = Interrupt request is not enabled
- bit 4        **ADCP6IE:** ADC Pair 6 Conversion Done Interrupt Enable bit  
                  1 = Interrupt request is enabled  
                  0 = Interrupt request is not enabled
- bit 3        **ADCP5IE:** ADC Pair 5 Conversion Done Interrupt Enable bit  
                  1 = Interrupt request is enabled  
                  0 = Interrupt request is not enabled
- bit 2        **ADCP4IE:** ADC Pair 4 Conversion Done Interrupt Enable bit  
                  1 = Interrupt request is enabled  
                  0 = Interrupt request is not enabled
- bit 1        **ADCP3IE:** ADC Pair 3 Conversion Done Interrupt Enable bit  
                  1 = Interrupt request is enabled  
                  0 = Interrupt request is not enabled
- bit 0        **ADCP2IE:** ADC Pair 2 Conversion Done Interrupt Enable bit  
                  1 = Interrupt request is enabled  
                  0 = Interrupt request is not enabled

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 7-21: IPC0: INTERRUPT PRIORITY CONTROL REGISTER 0

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	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)
  - 
  - 
  - 
  - 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
  - 000 = Interrupt source is disabled

## REGISTER 7-22: IPC1: INTERRUPT PRIORITY CONTROL REGISTER 1

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	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 3-0 **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

000 = Interrupt source is disabled

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 7-23: IPC2: INTERRUPT PRIORITY CONTROL REGISTER 2

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	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
- 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
  - 000 = Interrupt source is disabled



# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 7-24: IPC3: INTERRUPT PRIORITY CONTROL REGISTER 3

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	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
—	ADIP<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 **ADIP<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

000 = Interrupt source is disabled

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 7-25: IPC4: INTERRUPT PRIORITY CONTROL REGISTER 4

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	CNIP<2:0>			—	AC1IP<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      **Unimplemented:** Read as '0'
- bit 10-8    **AC1IP<2:0>:** Analog Comparator 1 Interrupt Priority bits
  - 111 = Interrupt is priority 7 (highest priority interrupt)
  - 
  - 
  - 
  - 001 = Interrupt is priority 1
  - 000 = Interrupt source is disabled
- bit 7      **Unimplemented:** Read as '0'
- bit 6-4    **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
  - 000 = Interrupt source is disabled

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 7-26: IPC5: INTERRUPT PRIORITY CONTROL REGISTER 5

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15						bit 8	

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	INT1IP<2:0>		
bit 7					bit 0		

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 15-3

**Unimplemented:** Read as '0'

bit 2-0

**INT1IP<2:0>:** External Interrupt 1 Priority bits

111 = Interrupt is priority 7 (highest priority interrupt)

•

•

•

001 = Interrupt is priority 1

000 = Interrupt source is disabled

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 7-27: IPC6: INTERRUPT PRIORITY CONTROL REGISTER 6

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	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
  - 000 = Interrupt source is disabled

## REGISTER 7-28: IPC7: INTERRUPT PRIORITY CONTROL REGISTER 7

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	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			

<b>Legend:</b>			
R = Readable bit	W = Writable bit	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
  - 000 = Interrupt source is disabled

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 7-29: IPC8: INTERRUPT PRIORITY CONTROL REGISTER 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	C1IP<2:0> <sup>(1)</sup>			—	C1RXIP<2:0> <sup>(1)</sup>		
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<sup>(1)</sup>
  - 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<sup>(1)</sup>
  - 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
  - 000 = Interrupt source is disabled

**Note 1:** Interrupts disabled on devices without ECAN™ modules.

**REGISTER 7-30: IPC9: INTERRUPT PRIORITY CONTROL REGISTER 9**

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-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			

<b>Legend:</b>			
R = Readable bit	W = Writable bit	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      **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
  - 000 = Interrupt source is disabled

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 7-31: IPC12: INTERRUPT PRIORITY CONTROL REGISTER 12

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	MI2C2IP<2:0>		
bit 15					bit 8		

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	SI2C2IP<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 **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 3-0 **Unimplemented:** Read as '0'



## REGISTER 7-32: IPC13: INTERRUPT PRIORITY CONTROL REGISTER 13

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	INT4IP<2:0>		
bit 15					bit 8		

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	INT3IP<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 **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 3-0 **Unimplemented:** Read as '0'

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 7-33: IPC14: INTERRUPT PRIORITY CONTROL REGISTER 14

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	QE11IP<2:0>		
bit 15					bit 8		

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	PSEMIP<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 **QE11IP<2:0>:** QE11 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 **PSEMIP<2:0>:** PWM Special Event Match 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'

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 7-34: 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	U-0	U-0	U-0
—	U1EIP<2:0>			—	—	—	—
bit 7				bit 0			

<b>Legend:</b>			
R = Readable bit	W = Writable bit	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 3-0     **Unimplemented:** Read as '0'

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 7-35: IPC17: INTERRUPT PRIORITY CONTROL REGISTER 17

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	C1TXIP<2:0> <sup>(1)</sup>		
bit 15					bit 8		

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7						bit 0	

### Legend:

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

bit 15-11      **Unimplemented:** Read as '0'

bit 10-8      **C1TXIP<2:0>:** ECAN1 Transmit Data Request Interrupt Priority bits<sup>(1)</sup>  
                  111 = Interrupt is priority 7 (highest priority interrupt)  
                  •  
                  •  
                  •  
                  001 = Interrupt is priority 1  
                  000 = Interrupt source is disabled

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

**Note 1:** Interrupts disabled on devices without ECAN™ modules.

**REGISTER 7-36: IPC18: INTERRUPT PRIORITY CONTROL REGISTER 18**

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	QEI2IP<2:0>			—	—	—	—
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	PSESMIP<2:0>			—	—	—	—
bit 7							bit 0

<b>Legend:</b>			
R = Readable bit	W = Writable bit	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    **QEI2IP<2:0>:** QEI2 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    **PSESMIP<2:0>:** PWM Special Event Secondary Match 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'

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 7-37: IPC20: INTERRUPT PRIORITY CONTROL REGISTER 20

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	ADCP10IP<2:0>			—	ADCP9IP<2:0>		
bit 15				bit 8			

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	ADCP8IP<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    **ADCP10IP<2:0>:** ADC Pair 10 Conversion Done Interrupt 1 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    **ADCP9IP<2:0>:** ADC Pair 9 Conversion Done Interrupt 1 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     **ADCP8IP<2:0>:** ADC Pair 8 Conversion Done Interrupt 1 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'

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 7-38: IPC21: INTERRUPT PRIORITY CONTROL REGISTER 21

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15						bit 8	

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	ADCP12IP<2:0>			—	ADCP11IP<2:0>		
bit 7				bit 0			

### Legend:

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

bit 15-7                      **Unimplemented:** Read as '0'

bit 6-4                      **ADCP12IP<2:0>:** ADC Pair 12 Conversion Done Interrupt 1 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                      **ADCP11IP<2:0>:** ADC Pair 11 Conversion Done Interrupt 1 Priority bits

111 = Interrupt is priority 7 (highest priority interrupt)

•

•

•

001 = Interrupt is priority 1

000 = Interrupt source is disabled

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 7-39: IPC23: INTERRUPT PRIORITY CONTROL REGISTER 23

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	PWM2IP<2:0>			—	PWM1IP<2:0>		
bit 15				bit 8			

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7							bit 0

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

- bit 15      **Unimplemented:** Read as '0'
- bit 14-12    **PWM2IP<2:0>:** PWM2 Interrupt Priority bits  
 111 = Interrupt is priority 7 (highest priority)  
 •  
 •  
 •  
 001 = Interrupt is priority 1  
 000 = Interrupt source is disabled
- bit 11      **Unimplemented:** Read as '0'
- bit 10-8    **PWM1IP<2:0>:** PWM1 Interrupt Priority bits  
 111 = Interrupt is priority 7 (highest priority)  
 •  
 •  
 •  
 001 = Interrupt is priority 1  
 000 = Interrupt source is disabled
- bit 7-0     **Unimplemented:** Read as '0'



# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 7-40: IPC24: INTERRUPT PRIORITY CONTROL REGISTER 24

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	PWM6IP<2:0>			—	PWM5IP<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
—	PWM4IP<2:0>			—	PWM3IP<2:0>		
bit 7				bit 0			

### Legend:

R = Readable bit  
-n = Value at POR

W = Writable bit  
'1' = Bit is set

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

- bit 15      **Unimplemented:** Read as '0'
- bit 14-12    **PWM6IP<2:0>:** PWM6 Interrupt Priority bits  
                  111 = Interrupt is priority 7 (highest priority)  
                  •  
                  •  
                  •  
                  001 = Interrupt is priority 1  
                  000 = Interrupt source is disabled
- bit 11      **Unimplemented:** Read as '0'
- bit 10-8     **PWM5IP<2:0>:** PWM5 Interrupt Priority bits  
                  111 = Interrupt is priority 7 (highest priority)  
                  •  
                  •  
                  •  
                  001 = Interrupt is priority 1  
                  000 = Interrupt source is disabled
- bit 7        **Unimplemented:** Read as '0'
- bit 6-4     **PWM4IP<2:0>:** PWM4 Interrupt Priority bits  
                  111 = Interrupt is priority 7 (highest priority)  
                  •  
                  •  
                  •  
                  001 = Interrupt is priority 1  
                  000 = Interrupt source is disabled
- bit 3        **Unimplemented:** Read as '0'
- bit 2-0     **PWM3IP<2:0>:** PWM3 Interrupt Priority bits  
                  111 = Interrupt is priority 7 (highest priority)  
                  •  
                  •  
                  •  
                  001 = Interrupt is priority 1  
                  000 = Interrupt source is disabled

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 7-41: IPC25: INTERRUPT PRIORITY CONTROL REGISTER 25

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	AC2IP<2:0>			—	PWM9IP<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
—	PWM8IP<2:0>			—	PWM7IP<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       **AC2IP<2:0>:** Analog Comparator 2 Interrupt Priority bits  
                   111 = Interrupt is priority 7 (highest priority)  
                   •  
                   •  
                   •  
                   001 = Interrupt is priority 1  
                   000 = Interrupt source is disabled
- bit 11           **Unimplemented:** Read as '0'
- bit 10-8        **PWM9IP<2:0>:** PWM9 Interrupt Priority bits  
                   111 = Interrupt is priority 7 (highest priority)  
                   •  
                   •  
                   •  
                   001 = Interrupt is priority 1  
                   000 = Interrupt source is disabled
- bit 7            **Unimplemented:** Read as '0'
- bit 6-4         **PWM8IP<2:0>:** PWM8 Interrupt Priority bits  
                   111 = Interrupt is priority 7 (highest priority)  
                   •  
                   •  
                   •  
                   001 = Interrupt is priority 1  
                   000 = Interrupt source is disabled
- bit 3            **Unimplemented:** Read as '0'
- bit 2-0         **PWM7IP<2:0>:** PWM7 Interrupt Priority bits  
                   111 = Interrupt is priority 7 (highest priority)  
                   •  
                   •  
                   •  
                   001 = Interrupt is priority 1  
                   000 = Interrupt source is disabled

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 7-42: IPC26: INTERRUPT PRIORITY CONTROL REGISTER 26

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15						bit 8	

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	AC4IP<2:0>			—	AC3IP<2:0>		
bit 7				bit 0			

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 15-7 **Unimplemented:** Read as '0'

bit 6-4 **AC4IP<2:0>:** Analog Comparator 4 Interrupt Priority bits

111 = Interrupt is priority 7 (highest priority)

- 
- 
- 

001 = Interrupt is priority 1

000 = Interrupt source is disabled

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

bit 2-0 **AC3IP<2:0>:** Analog Comparator 3 Interrupt Priority bits

111 = Interrupt is priority 7 (highest priority)

- 
- 
- 

001 = Interrupt is priority 1

000 = Interrupt source is disabled

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 7-43: IPC27: INTERRUPT PRIORITY CONTROL REGISTER 27

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	ADCP1IP<2:0>			—	ADCP0IP<2:0>		
bit 15				bit 8			

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7							bit 0

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

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

bit 14-12 **ADCP1IP<2:0>:** ADC Pair 1 Conversion Done Interrupt Priority bits

111 = Interrupt is priority 7 (highest priority interrupt)

•

•

•

001 = Interrupt is priority 1

000 = Interrupt source is disabled

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

bit 10-8 **ADCP0IP<2:0>:** ADC Pair 0 Conversion Done Interrupt Priority bits

111 = Interrupt is priority 7 (highest priority interrupt)

•

•

•

001 = Interrupt is priority 1

000 = Interrupt source is disabled

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

**REGISTER 7-44: IPC28: INTERRUPT PRIORITY CONTROL REGISTER 28**

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	ADCP5IP<2:0>			—	ADCP4IP<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
—	ADCP3IP<2:0>			—	ADCP2IP<2:0>		
bit 7				bit 0			

<b>Legend:</b>			
R = Readable bit	W = Writable bit	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    **ADCP5IP<2:0>:** ADC Pair 5 Conversion Done Interrupt Priority bits
  - 111 = Interrupt is priority 7 (highest priority interrupt)
  - 
  - 
  - 
  - 001 = Interrupt is priority 1
  - 000 = Interrupt source is disabled
- bit 11      **Unimplemented:** Read as '0'
- bit 10-8    **ADCP4IP<2:0>:** ADC Pair 4 Conversion Done 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    **ADCP3IP<2:0>:** ADC Pair 3 Conversion Done 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    **ADCP2IP<2:0>:** ADC Pair 2 Conversion Done Interrupt Priority bits
  - 111 = Interrupt is priority 7 (highest priority interrupt)
  - 
  - 
  - 
  - 001 = Interrupt is priority 1
  - 000 = Interrupt source is disabled

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 7-45: IPC29: INTERRUPT PRIORITY CONTROL REGISTER 29

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	ADCP7IP<2:0>			—	ADCP6IP<2:0>		
bit 7							bit 0

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 15-7 **Unimplemented:** Read as '0'

bit 6-4 **ADCP7IP<2:0>:** ADC Pair 7 Conversion Done Interrupt 1 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 **ADCP6IP<2:0>:** ADC Pair 6 Conversion Done Interrupt 1 Priority bits

111 = Interrupt is priority 7 (highest priority interrupt)

•

•

•

001 = Interrupt is priority 1

000 = Interrupt source is disabled

## REGISTER 7-46: 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

<b>Legend:</b>			
R = Readable bit	W = Writable bit	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

Complete the following steps to configure an interrupt source at initialization:

1. Set the NSTDIS bit (INTCON1<15>) if nested interrupts are not desired.
2. 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 can 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.
4. 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 used to declare an ISR and initialize the IVT with the correct vector address depends on the programming language (C or assembler) and the language development toolsuite used to develop the application.

In general, the user application must clear the interrupt flag in the appropriate IFSx register for the source of interrupt that the ISR handles. Otherwise, program will re-enter the ISR 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

The following steps outline the procedure to disable all user interrupts:

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 EOh with SRL.

To enable user interrupts, the `POP` instruction can be used to restore the previous SR value.

**Note:** Only user interrupts with a priority level of 7 or lower 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.



## 8.0 DIRECT MEMORY ACCESS (DMA)

**Note 1:** This data sheet summarizes the features of the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 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](http://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.

**Note:** The DMA module is not available on dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406 devices.

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

**TABLE 8-1: DMA CHANNEL TO PERIPHERAL ASSOCIATIONS**

Peripheral to DMA Association	DMAxREQ Register IRQSEL<6:0> Bits	DMAxPAD Register Values to Read from Peripheral	DMAxPAD Register Values to Write to Peripheral
INT0 – External Interrupt 0	0000000	—	—
IC1 – Input Capture 1	0000001	0x0140 (IC1BUF)	—
IC2 – Input Capture 2	0000101	0x0144 (IC2BUF)	—
IC3 – Input Capture 3	0100101	0x0148 (IC3BUF)	—
IC4 – Input Capture 4	0100110	0x014C (IC4BUF)	—
OC1 – Output Compare 1 Data	0000010	—	0x0182 (OC1R)
OC1 – Output Compare 1 Secondary Data	0000010	—	0x0180 (OC1RS)
OC2 – Output Compare 2 Data	0000110	—	0x0188 (OC2R)
OC2 – Output Compare 2 Secondary Data	0000110	—	0x0186 (OC2RS)
OC3 – Output Compare 3 Data	0011001	—	0x018E (OC3R)
OC3 – Output Compare 3 Secondary Data	0011001	—	0x018C (OC3RS)
OC4 – Output Compare 4 Data	0011010	—	0x0194 (OC4R)
OC4 – Output Compare 4 Secondary Data	0011010	—	0x0192 (OC4RS)
TMR2 – Timer2	0000111	—	—
TMR3 – Timer3	0001000	—	—
TMR4 – Timer4	0011011	—	—
TMR5 – Timer5	0011100	—	—
SPI1 – Transfer Done	0001010	0x0248 (SPI1BUF)	0x0248 (SPI1BUF)
SPI2 – Transfer Done	0100001	0x0268 (SPI2BUF)	0x0268 (SPI2BUF)
UART1RX – UART1 Receiver	0001011	0x0226 (U1RXREG)	—
UART1TX – UART1 Transmitter	0001100	—	0x0224 (U1TXREG)
UART2RX – UART2 Receiver	0011110	0x0236 (U2RXREG)	—
UART2TX – UART2 Transmitter	0011111	—	0x0234 (U2TXREG)
ECAN1 – RX Data Ready	0100010	0x0640 (C1RXD)	—
ECAN1 – TX Data Request	1000110	—	0x0642 (C1TXD)

The DMA controller features four 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

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.

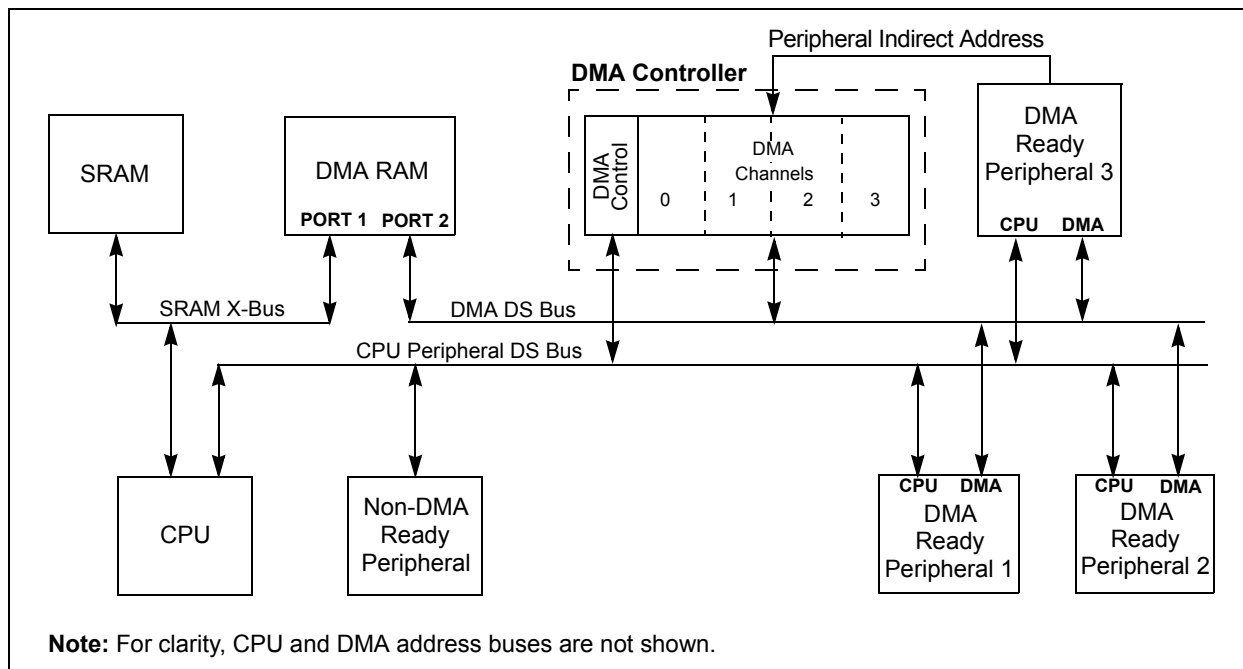
## 8.1 DMAC Registers

Each DMAC Channel  $x$  ( $x = 0, 1, 2, \text{ or } 3$ ) 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.

**FIGURE 8-1: TOP LEVEL SYSTEM ARCHITECTURE USING A DEDICATED TRANSACTION BUS**



## REGISTER 8-1: DMA<sub>x</sub>CON: 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
—	—	AMODE<1:0>		—	—	MODE<1:0>	
bit 7							bit 0

<b>Legend:</b>			
R = Readable bit	W = Writable bit	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      **Unimplemented:** Read as '0'
- bit 1-0      **MODE<1:0>:** DMA Channel Operating Mode Select bits  
             11 = One-Shot, Ping-Pong modes enabled (one block transfer from/to each DMA RAM buffer)  
             10 = Continuous, Ping-Pong modes enabled  
             01 = One-Shot, Ping-Pong modes disabled  
             00 = Continuous, Ping-Pong modes disabled

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

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

R/W-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
FORCE <sup>(1)</sup>	—	—	—	—	—	—	—
bit 15							bit 8

U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	IRQSEL<6:0> <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                      **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>  
                                     0000000-1111111 = 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

<b>Legend:</b>							
R = Readable bit	W = Writable bit	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

<b>Legend:</b>							
R = Readable bit	W = Writable bit	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.

**2:** See [Table 8-1](#) for a complete list of peripheral addresses.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## 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> <sup>(2)</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
CNT<7:0>							
bit 7						bit 0	

### Legend:

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

bit 15-10      **Unimplemented:** Read as '0'

bit 9-0      **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**

U-0	U-0	U-0	U-0	R/C-0	R/C-0	R/C-0	R/C-0
—	—	—	—	PWCOL3	PWCOL2	PWCOL1	PWCOL0
bit 15				bit 8			

U-0	U-0	U-0	U-0	R/C-0	R/C-0	R/C-0	R/C-0
—	—	—	—	XWCOL3	XWCOL2	XWCOL1	XWCOL0
bit 7				bit 0			

<b>Legend:</b>							
R = Readable bit	W = Writable bit	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       **PWCOL3:** Channel 3 Peripheral Write Collision Flag bit  
               1 = Write collision detected  
               0 = No write collision detected
- bit 10       **PWCOL2:** Channel 2 Peripheral Write Collision Flag bit  
               1 = Write collision detected  
               0 = No write collision detected
- 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  
               1 = Write collision detected  
               0 = No write collision detected
- bit 7-4      **Unimplemented:** Read as '0'
- bit 3         **XWCOL3:** Channel 3 DMA RAM Write Collision Flag bit  
               1 = Write collision detected  
               0 = No write collision detected
- bit 2         **XWCOL2:** Channel 2 DMA RAM Write Collision Flag bit  
               1 = Write collision detected  
               0 = No write collision detected
- bit 1         **XWCOL1:** Channel 1 DMA RAM Write Collision Flag bit  
               1 = Write collision detected  
               0 = No write collision detected
- bit 0         **XWCOL0:** Channel 0 DMA RAM Write Collision Flag bit  
               1 = Write collision detected  
               0 = No write collision detected

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## 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			

U-0	U-0	U-0	U-0	R-0	R-0	R-0	R-0
—	—	—	—	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 = Reserved
  - 
  - 
  - 
  - 0100 = Reserved
  - 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 7-4      **Unimplemented:** Read as '0'
- 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



# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## 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

NOTES:

## 9.0 OSCILLATOR CONFIGURATION

**Note 1:** This data sheet summarizes the features of the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 families of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 42. “Oscillator (Part IV)”** (DS70307) in the “*dsPIC33F/PIC24H Family Reference Manual*”, which is available from the Microchip web site ([www.microchip.com](http://www.microchip.com)).

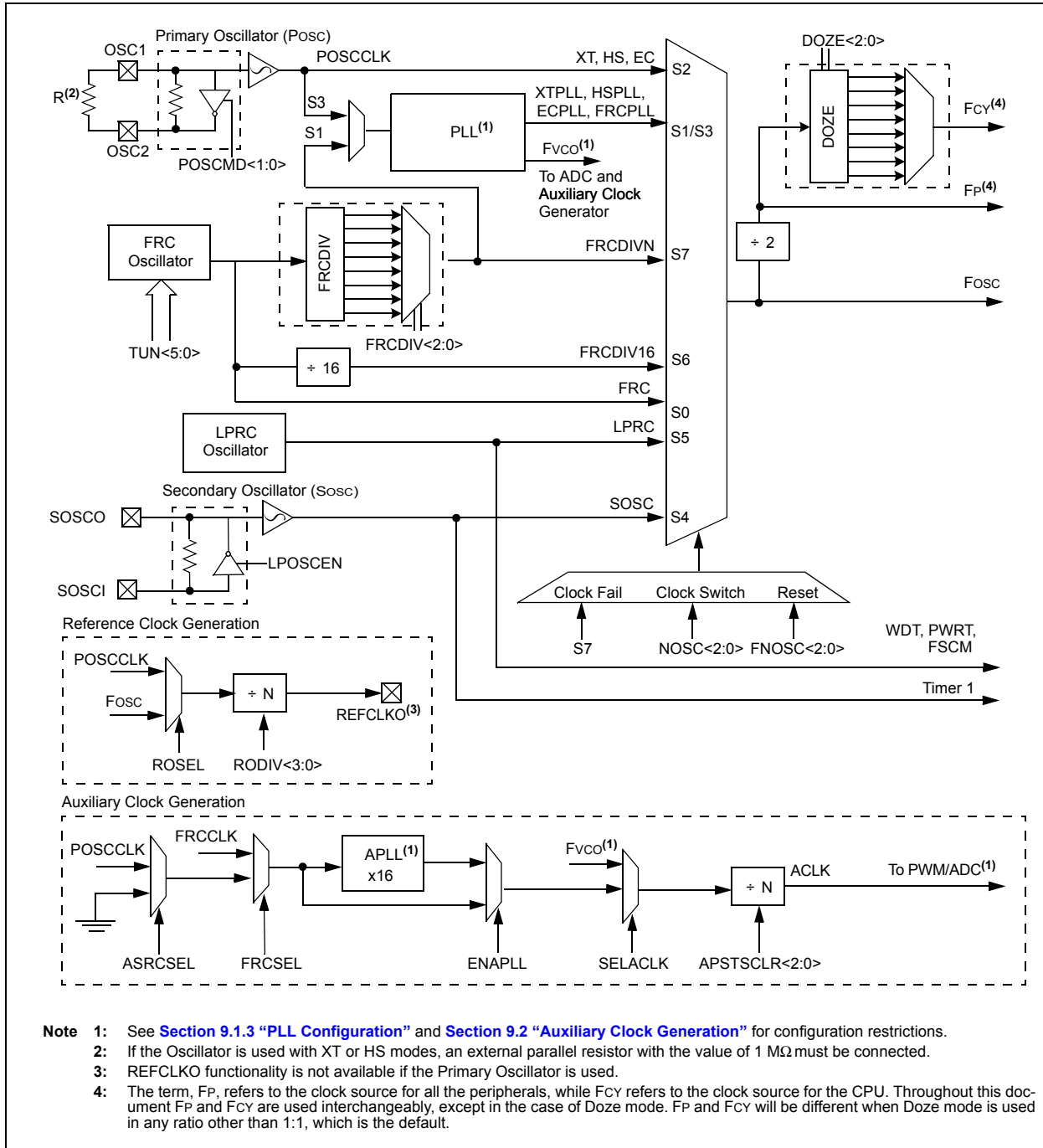
**2:** Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 “Memory Organization”** in this data sheet for device-specific register and bit information.

The oscillator system provides:

- External and internal oscillator options as clock sources
- An on-chip Phase-Locked Loop (PLL) to scale the internal operating frequency to the required system clock frequency
- An internal FRC oscillator that 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.
- Auxiliary PLL for ADC and PWM

A simplified diagram of the oscillator system is shown in [Figure 9-1](#).

FIGURE 9-1: OSCILLATOR SYSTEM DIAGRAM



## 9.1 CPU Clocking System

The dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 devices provide six system clock options:

- Fast RC (FRC) Oscillator
- FRC Oscillator with PLL
- Primary (XT, HS, or EC) Oscillator
- Primary Oscillator with PLL
- Low-Power RC (LPRC) Oscillator
- FRC Oscillator with Postscaler
- Secondary (LP) Oscillator

### 9.1.1 SYSTEM CLOCK SOURCES

The Fast RC (FRC) internal oscillator runs at a nominal frequency of 7.37 MHz. 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> (CLKDIV<10:8>) bits.

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 50 MHz. The crystal is connected to the OSC1 and OSC2 pins
- EC (External Clock): The 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 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 27-20](#)) and the value of the FRC Oscillator Tuning register (see [Register 9-4](#)).

### 9.1.2 SYSTEM CLOCK SELECTION

The oscillator source 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 24.1 “Configuration Bits”](#) for further details.) The Initial Oscillator Selection Configuration bits, FNOSEL<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 among 12 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 50 MIPS are supported by the device architecture.

Instruction execution speed or device operating frequency, FCY, is given by [Equation 9-1](#).

#### EQUATION 9-1: DEVICE OPERATING FREQUENCY

$$FCY = FOSC/2$$

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

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

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

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

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 significant 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 in the range of 0.8 MHz to 8 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 100 MHz, which generates device operating speeds of 6.25-50 MIPS.

For a primary oscillator or FRC oscillator, output 'FIN', the PLL output 'Fosc' is given by Equation 9-2.

EQUATION 9-2: Fosc CALCULATION

$$F_{OSC} = F_{IN} * \left( \frac{M}{N1 * N2} \right)$$

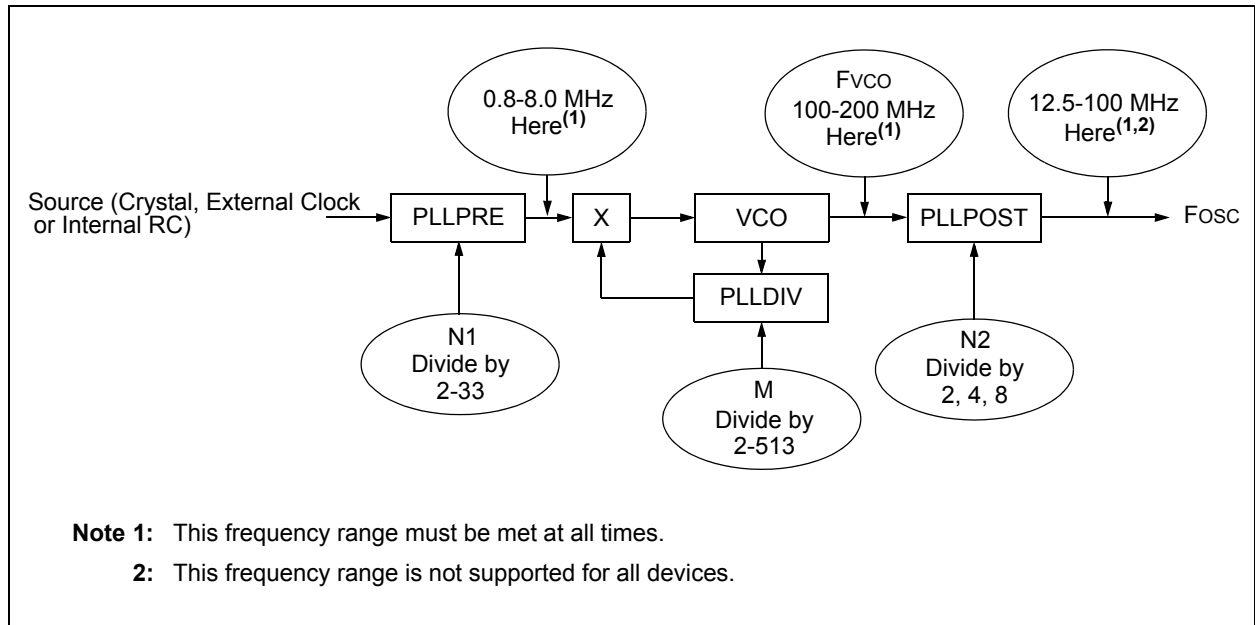
For example, suppose a 10 MHz crystal is being used with the selected oscillator mode of XT with PLL (see Equation 9-3).

- 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> = 0x26, then M = 40. This yields a VCO output of 5 x 40 = 200 MHz, which is within the 100-200 MHz ranged needed.
- If PLLPOST<1:0> = 0, then N2 = 2. This provides a Fosc of 200/2 = 100 MHz. The resultant device operating speed is 100/2 = 40 MIPS.

EQUATION 9-3: XT WITH PLL MODE EXAMPLE

$$F_{CY} = \frac{F_{OSC}}{2} = \frac{1}{2} \left( \frac{10000000 * 40}{2 * 2} \right) = 50 \text{ MIPS}$$

FIGURE 9-2: PLL BLOCK DIAGRAM



## 9.2 Auxiliary Clock Generation

The auxiliary clock generation is used for a peripherals that need to operate at a frequency unrelated to the system clock such as a PWM or ADC.

The primary oscillator and internal FRC oscillator sources can be used with an auxiliary PLL to obtain the auxiliary clock. The auxiliary PLL has a fixed 16x multiplication factor.

The auxiliary clock has the following configuration restrictions:

- For proper PWM operation, auxiliary clock generation must be configured for 120 MHz (see parameter OS56 in [Table 27-18](#) in [Section 27.0 “Electrical Characteristics”](#)). If a slower frequency is desired, the PWM Input Clock Prescaler (Divider) Select bits (PCLKDIV<2:0>) should be used.
- To achieve 1.04 ns PWM resolution, the auxiliary clock must use the 16x auxiliary PLL (APLL). All other clock sources will have a minimum PWM resolution of 8 ns.
- If the primary PLL is used as a source for the auxiliary clock, the primary PLL should be configured up to a maximum operation of 30 MIPS or less.

## 9.3 Reference Clock Generation

The reference clock output logic provides the user with the ability to output a clock signal based on the system clock or the crystal oscillator on a device pin. The user application can specify a wide range of clock scaling prior to outputting the reference clock.

## 9.4 Oscillator Control Registers

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

U-0	R-y	R-y	R-y	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	U-0	R/W-0
CLKLOCK	—	LOCK	—	CF	—	—	OSWEN
bit 7							bit 0

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

bit 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 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 PLL
- 000 = Fast RC oscillator (FRC)

bit 7 **CLKLOCK:** Clock Lock Enable bit

If clock switching is enabled and FSCM is disabled, (FOSC<FCKSM> = 0b01):

- 1 = Clock switching is disabled, system clock source is locked
- 0 = Clock switching is enabled, system clock source can be modified by clock switching

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

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

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 42. "Oscillator (Part IV)"** (DS70307) in the *"dsPIC33F/PIC24H Family Reference Manual"* for details.

**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.



# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 9-2: CLKDIV: CLOCK DIVISOR REGISTER

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
PLLPOST<1:0>		—	PLLPRE<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                      **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  
 111 = Fcy/128  
 110 = Fcy/64  
 101 = Fcy/32  
 100 = Fcy/16  
 011 = Fcy/8 (default)  
 010 = Fcy/4  
 001 = Fcy/2  
 000 = Fcy/1
- 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  
 111 = FRC divide by 256  
 110 = FRC divide by 64  
 101 = FRC divide by 32  
 100 = FRC divide by 16  
 011 = FRC divide by 8  
 010 = FRC divide by 4  
 001 = FRC divide by 2  
 000 = FRC divide by 1 (default)
- bit 7-6                      **PLLPOST<1:0>:** PLL VCO Output Divider Select bits (also denoted as 'N2', PLL postscaler)  
 11 = Output/8  
 10 = Reserved  
 01 = Output/4 (default)  
 00 = Output/2
- 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.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 9-3: PLLFBD: PLL FEEDBACK DIVISOR REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	PLLDIV<8>
bit 15							bit 8

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)

000000000 = 2

000000001 = 3

000000010 = 4

•

•

•

000110000 = 50 (default)

•

•

•

111111111 = 513

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 9-4: OSCTUN: OSCILLATOR TUNING REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	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.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 9-5: ACLKCON: AUXILIARY CLOCK DIVISOR CONTROL REGISTER

R/W-0	R-0	R/W-1	U-0	U-0	R/W-1	R/W-1	R/W-1
ENAPLL	APLLCK	SELACLK	—	—	APSTSCLR<2:0>		
bit 15							bit 0

R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0	U-0
ASRCSEL	FRCSEL	—	—	—	—	—	—
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            **ENAPLL:** Auxiliary PLL Enable bit  
                   1 = APLL is enabled  
                   0 = APLL is disabled
- bit 14            **APLLCK:** APLL Locked Status bit (read-only)  
                   1 = Indicates that auxiliary PLL is in lock  
                   0 = Indicates that auxiliary PLL is not in lock
- bit 13            **SELACLK:** Select Auxiliary Clock Source for Auxiliary Clock Divider bit  
                   1 = Auxiliary Oscillators provides the source clock for auxiliary clock divider  
                   0 = Primary PLL (FVCO) provides the source clock for auxiliary clock divider
- bit 12-11        **Unimplemented:** Read as '0'
- bit 10-8         **APSTSCLR<2:0>:** Auxiliary Clock Output Divider bits  
                   111 = Divided by 1  
                   110 = Divided by 2  
                   101 = Divided by 4  
                   100 = Divided by 8  
                   011 = Divided by 16  
                   010 = Divided by 32  
                   001 = Divided by 64  
                   000 = Divided by 256
- bit 7             **ASRCSEL:** Select Reference Clock Source for Auxiliary Clock bit  
                   1 = Primary oscillator is the clock source  
                   0 = No clock input is selected
- bit 6             **FRCSEL:** Select Reference Clock Source for Auxiliary PLL bit  
                   1 = Select FRC clock for auxiliary PLL  
                   0 = Input clock source is determined by ASRCSEL bit setting
- bit 5-0          **Unimplemented:** Read as '0'

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 9-6: REFOCON: REFERENCE OSCILLATOR CONTROL REGISTER

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ROON	—	ROSSLP	ROSEL	RODIV<3:0> <sup>(1)</sup>			
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  
-n = Value at POR

W = Writable bit  
'1' = Bit is set

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

- bit 15      **ROON:** Reference Oscillator Output Enable bit  
             1 = Reference oscillator output enabled on REFCLK0 pin  
             0 = Reference oscillator output disabled
- bit 14      **Unimplemented:** Read as '0'
- bit 13      **ROSSLP:** Reference Oscillator Run in Sleep bit  
             1 = Reference oscillator output continues to run in Sleep  
             0 = Reference oscillator output is disabled in Sleep
- bit 12      **ROSEL:** Reference Oscillator Source Select bit  
             1 = Oscillator crystal used as the reference clock  
             0 = System clock used as the reference clock
- bit 11-8    **RODIV<3:0>:** Reference Oscillator Divider bits<sup>(1)</sup>  
             1111 = Reference clock divided by 32,768  
             1110 = Reference clock divided by 16,384  
             1101 = Reference clock divided by 8,192  
             1100 = Reference clock divided by 4,096  
             1011 = Reference clock divided by 2,048  
             1010 = Reference clock divided by 1,024  
             1001 = Reference clock divided by 512  
             1000 = Reference clock divided by 256  
             0111 = Reference clock divided by 128  
             0110 = Reference clock divided by 64  
             0101 = Reference clock divided by 32  
             0100 = Reference clock divided by 16  
             0011 = Reference clock divided by 8  
             0010 = Reference clock divided by 4  
             0001 = Reference clock divided by 2  
             0000 = Reference clock
- bit 7-0     **Unimplemented:** Read as '0'

**Note 1:** The reference oscillator output must be disabled (ROON = 0) before writing to these bits.

## 9.5 Clock Switching Operation

Applications are free to switch among any of the four clock sources (primary, LP, FRC and LPRC) under software control at any time. To limit the possible side effects of this flexibility, dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 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 among the different primary submodes without reprogramming the device.

### 9.5.1 ENABLING CLOCK SWITCHING

To enable clock switching, the FCKSM1 Configuration bit in the Configuration register must be programmed to '0'. (Refer to [Section 24.1 "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.

### 9.5.2 OSCILLATOR SWITCHING SEQUENCE

To perform a clock switch, the following basic sequence is required:

1. 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.
3. Write the appropriate value to the NOSC control bits (OSCCON<10:8>) for the new oscillator source.
4. Perform the unlock sequence to allow a write to the OSCCON register low byte.
5. Set the OSWEN bit (OSCCON<0>) to initiate the oscillator switch.

Once the basic sequence is completed, the system clock hardware responds automatically as follows:

1. The clock switching hardware compares the COSC Status bits with the new value of the NOSC control bits. If they are the same, the clock switch is a redundant operation. In this case, the OSWEN bit is cleared automatically and the clock switch is aborted.

2. If a valid clock switch has been initiated, the LOCK (OSCCON<5>) and the CF (OSCCON<3>) Status bits are cleared.
3. 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).
4. The hardware waits for 10 clock cycles from the new clock source and then performs the clock switch.
5. 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.
6. The old clock source is turned off at this time, with the exception of LPRC (if WDT or FSCM are 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 42. "Oscillator \(Part IV\)"](#) (DS70307) in the *dsPIC33F/PIC24H Family Reference Manual* for details.

## 9.6 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.

## 10.0 POWER-SAVING FEATURES

**Note 1:** This data sheet summarizes the features of the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 families of devices. 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](http://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 dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 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. 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

The 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”**.

#### 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 Instruction-Based Power-Saving Modes

The 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 assembler 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

The following occur in Sleep mode:

- 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, 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. This includes the items such as the input change notification on the I/O ports or peripherals that use an external clock input.
- Any peripheral that requires the system clock source for its operation is disabled.

The device will wake-up from Sleep mode on any of these events:

- Any interrupt source that is individually enabled
- Any form of device Reset
- A WDT time-out

On wake-up from Sleep mode, the processor restarts with the same clock source that was active when Sleep mode was entered.

## 10.2.2 IDLE MODE

The following occur in Idle mode:

- 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 10.5 “Peripheral Module Disable”](#)).
- If the WDT or FSCM is enabled, the LPRC also remains active.

The device will wake-up from Idle mode on any of these events:

- Any interrupt that is individually enabled
- Any device Reset
- A WDT time-out

On wake-up from Idle mode, the clock is reapplied to the CPU and instruction execution will begin (2-4 clock cycles later), starting with the instruction following the `PWRSVAV` 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 `PWRSVAV` 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

The preferred strategies for reducing power consumption are changing clock speed and invoking one of the power-saving modes. In some circumstances, this may not be 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 can introduce communication errors, while using a power-saving mode can 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.

Programs can 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. An automatic return to full-speed CPU operation on interrupts can be 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 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 PWM Power-Saving Features

Typically, many applications need either a high resolution duty cycle or phase offset (for fixed frequency operation) or a high resolution PWM period for variable frequency modes of operation (such as Resonant mode). Very few applications require both high resolution modes simultaneously.

The HRPDIS and the HRDDIS bits in the AUXCONX registers permit the user to disable the circuitry associated with the high resolution duty cycle and PWM period to reduce the operating current of the device.

If the HRDDIS bit is set, the circuitry associated with the high resolution duty cycle, phase offset, and dead time for the respective PWM generator is disabled. If the HRPDIS bit is set, the circuitry associated with the high resolution PWM period for the respective PWM generator is disabled.

When the HRPDIS bit is set, the smallest unit of measure for the PWM period is 8.32 ns.

If the HRDDIS bit is set, the smallest unit of measure for the PWM duty cycle, phase offset and dead time is 8.32 ns.



## 10.5 Peripheral Module Disable

The Peripheral Module Disable (PMD) registers provide a method to disable a peripheral module by stopping all clock sources supplied to that module. When a peripheral is disabled using the appropriate PMD control bit, the peripheral is in a minimum power consumption state. The control and STATUS registers associated with the peripheral are also disabled, so writes to those registers will have no effect and read values will be invalid.

A peripheral module is enabled only if both the associated bit in the PMD register is cleared and the peripheral is supported by the specific dsPIC® DSC variant. If the peripheral is present in the device, it is enabled in the PMD register by default.

<p><b>Note:</b> If a PMD bit is set, the corresponding module is disabled after a delay of one instruction cycle. Similarly, if a PMD bit is cleared, the corresponding module is enabled after a delay of one instruction cycle (assuming the module control registers are already configured to enable module operation).</p>
---

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## 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	QE11MD	PWMMD <sup>(1)</sup>	—
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
I2C1MD	U2MD	U1MD	SPI2MD	SPI1MD	—	C1MD	ADCMD
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      **T5MD:** Timer5 Module Disable bit  
 1 = Timer5 module is disabled  
 0 = Timer5 module is enabled
- bit 14      **T4MD:** Timer4 Module Disable bit  
 1 = Timer4 module is disabled  
 0 = Timer4 module is enabled
- bit 13      **T3MD:** Timer3 Module Disable bit  
 1 = Timer3 module is disabled  
 0 = Timer3 module is enabled
- bit 12      **T2MD:** Timer2 Module Disable bit  
 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      **QE11MD:** QE11 Module Disable bit  
 1 = QE11 module is disabled  
 0 = QE11 module is enabled
- bit 9        **PWMMD:** PWM Module Disable bit<sup>(1)</sup>  
 1 = PWM module is disabled  
 0 = PWM module is enabled
- bit 8        **Unimplemented:** Read as '0'
- bit 7        **I2C1MD:** I2C1 Module Disable bit  
 1 = I2C1 module is disabled  
 0 = I2C1 module is enabled
- 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:** Once the PWM module is re-enabled (PWMMD is set to '1' and then set to '0'), all PWM registers must be reinitialized.

## REGISTER 10-1: PMD1: PERIPHERAL MODULE DISABLE CONTROL REGISTER 1 (CONTINUED)

bit 3	<b>SPI1MD:</b> SPI1 Module Disable bit 1 = SPI1 module is disabled 0 = SPI1 module is enabled
bit 2	<b>Unimplemented:</b> Read as '0'
bit 1	<b>C1MD:</b> ECAN1 Module Disable bit 1 = ECAN1 module is disabled 0 = ECAN1 module is enabled
bit 0	<b>ADCMD:</b> ADC Module Disable bit 1 = ADC module is disabled 0 = ADC module is enabled

**Note 1:** Once the PWM module is re-enabled (PWMMD is set to '1' and then set to '0'), all PWM registers must be reinitialized.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 10-2: PMD2: PERIPHERAL MODULE DISABLE CONTROL REGISTER 2

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	IC4MD	IC3MD	IC2MD	IC1MD
bit 15							bit 8

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	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 15-12     **Unimplemented:** Read as '0'
- bit 11       **IC4MD:** Input Capture 4 Module Disable bit  
               1 = Input Capture 4 module is disabled  
               0 = Input Capture 4 module is enabled
- bit 19       **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 7-4      **Unimplemented:** Read as '0'
- 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**

U-0	U-0	U-0	U-0	U-0	R/W-0	U-0	U-0
—	—	—	—	—	CMPMD	—	—
bit 15						bit 8	
U-0	U-0	R/W-0	U-0	U-0	U-0	R/W-0	U-0
—	—	QE12MD	—	—	—	I2C2MD	—
bit 7						bit 0	

<b>Legend:</b>			
R = Readable bit	W = Writable bit	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      **CMPMD:** Analog Comparator Module Disable bit  
                  1 = Analog Comparator module is disabled  
                  0 = Analog Comparator module is enabled
- bit 9-6     **Unimplemented:** Read as '0'
- bit 5        **QE12MD:** QE12 Module Disable bit  
                  1 = QE12 module is disabled  
                  0 = QE12 module is enabled
- bit 4-2     **Unimplemented:** Read as '0'
- bit 1        **I2C2MD:** I2C2 Module Disable bit  
                  1 = I2C2 module is disabled  
                  0 = I2C2 module is enabled
- bit 0        **Unimplemented:** Read as '0'

**REGISTER 10-4: PMD4: PERIPHERAL MODULE DISABLE 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	R/W-0	U-0	U-0	U-0
—	—	—	—	REFOMD	—	—	—
bit 7						bit 0	

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

- bit 15-4    **Unimplemented:** Read as '0'
- bit 3        **REFOMD:** Reference Clock Generator Module Disable bit  
                  1 = Reference clock generator module is disabled  
                  0 = Reference clock generator module is enabled
- bit 2-0     **Unimplemented:** Read as '0'

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 10-5: PMD6: PERIPHERAL MODULE DISABLE CONTROL REGISTER 6

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PWM8MD	PWM7MD	PWM6MD	PWM5MD	PWM4MD	PWM3MD	PWM2MD	PWM1MD
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 **PWM8MD:** PWM Generator 8 Module Disable bit

1 = PWM Generator 8 module is disabled

0 = PWM Generator 8 module is enabled

bit 14 **PWM7MD:** PWM Generator 7 Module Disable bit

1 = PWM Generator 7 module is disabled

0 = PWM Generator 7 module is enabled

bit 13 **PWM6MD:** PWM Generator 6 Module Disable bit

1 = PWM Generator 6 module is disabled

0 = PWM Generator 6 module is enabled

bit 12 **PWM5MD:** PWM Generator 5 Module Disable bit

1 = PWM Generator 5 module is disabled

0 = PWM Generator 5 module is enabled

bit 11 **PWM4MD:** PWM Generator 4 Module Disable bit

1 = PWM Generator 4 module is disabled

0 = PWM Generator 4 module is enabled

bit 10 **PWM3MD:** PWM Generator 3 Module Disable bit

1 = PWM Generator 3 module is disabled

0 = PWM Generator 3 module is enabled

bit 9 **PWM2MD:** PWM Generator 2 Module Disable bit

1 = PWM Generator 2 module is disabled

0 = PWM Generator 2 module is enabled

bit 8 **PWM1MD:** PWM Generator 1 Module Disable bit

1 = PWM Generator 1 module is disabled

0 = PWM Generator 1 module is enabled

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

## REGISTER 10-6: PMD7: PERIPHERAL MODULE DISABLE CONTROL REGISTER 7

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	CMP4MD	CMP3MD	CMP2MD	CMP1MD
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	PWM9MD
bit 7							bit 0

<b>Legend:</b>			
R = Readable bit	W = Writable bit	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       **CMP4MD:** Analog Comparator 4 Module Disable bit  
               1 = Analog Comparator 4 module is disabled  
               0 = Analog Comparator 4 module is enabled
- bit 10       **CMP3MD:** Analog Comparator 3 Module Disable bit  
               1 = Analog Comparator 3 module is disabled  
               0 = Analog Comparator 3 module is enabled
- bit 9         **CMP2MD:** Analog Comparator 2 Module Disable bit  
               1 = Analog Comparator 2 module is disabled  
               0 = Analog Comparator 2 module is enabled
- bit 8         **CMP1MD:** Analog Comparator 1 Module Disable bit  
               1 = Analog Comparator 1 module is disabled  
               0 = Analog Comparator 1 module is enabled
- bit 7-1      **Unimplemented:** Read as '0'
- bit 0         **PWM9MD:** PWM Generator 9 Module Disable bit  
               1 = PWM Generator 9 module is disabled  
               0 = PWM Generator 9 module is enabled

NOTES:



## 11.0 I/O PORTS

**Note 1:** This data sheet summarizes the features of the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 families of devices. 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](http://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/CLKI) are shared among 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

Generally a parallel I/O port that shares a pin with a peripheral is subservient to the peripheral. The peripheral's output buffer data and control signals are provided to a pair of multiplexers. The multiplexers select whether the peripheral or the associated port

has ownership of the output data and control signals of the I/O pin. The logic also prevents "loop through", in which a port's digital output can drive the input of a peripheral that shares the same pin. **Figure 11-1** shows how ports are shared with other peripherals and the associated I/O pin to which they are connected.

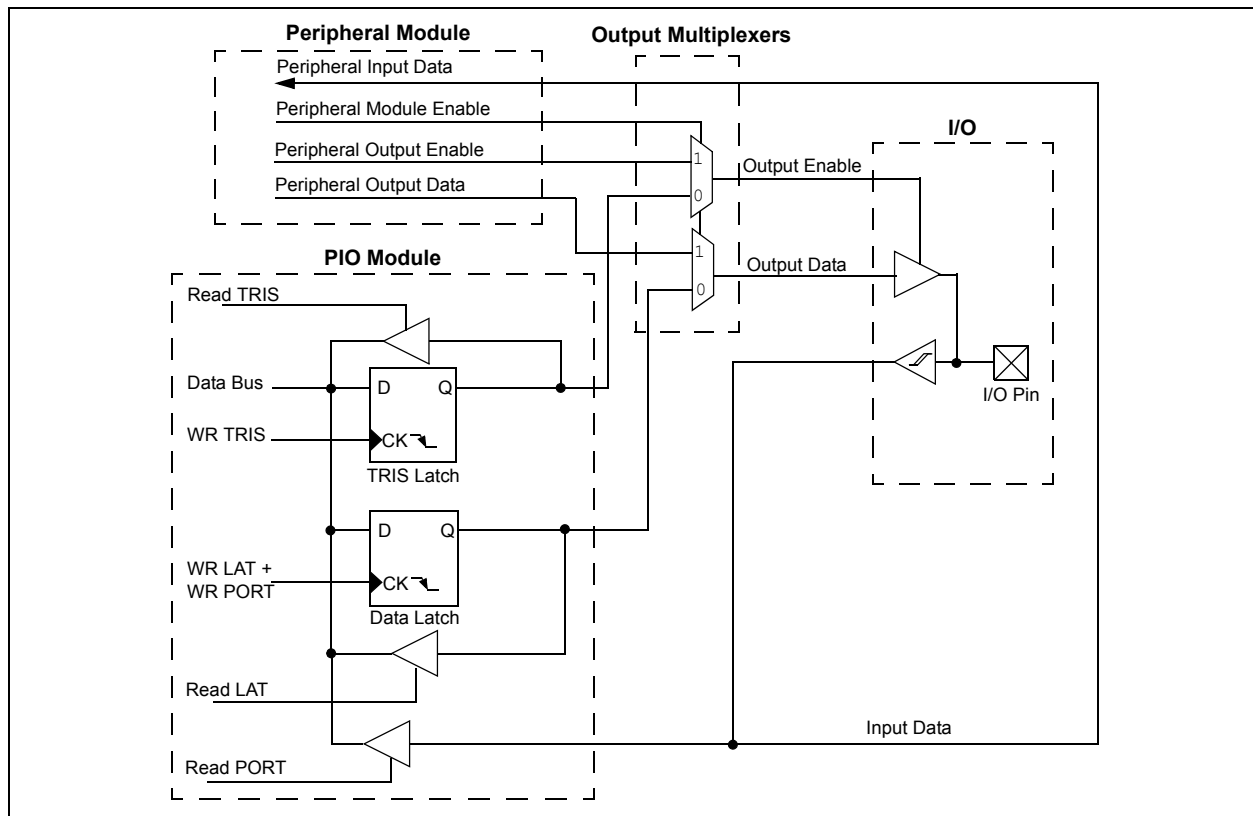
When a peripheral is enabled and the peripheral is actively driving an associated pin, the use of the pin as a general purpose output pin is disabled. The I/O pin can be read, but the output driver for the parallel port bit is disabled. If a peripheral is enabled, but the peripheral is not actively driving a pin, that pin can 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 '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 pin 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.

**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 digital-only 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 V<sub>DD</sub> (for example, 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 V<sub>IH</sub> specification.

Refer to “[Pin Diagrams](#)” for the available pins and their functionality.

## 11.3 Configuring Analog Port Pins

The ADPCFG and TRIS registers control the operation of the analog-to-digital port pins. The port pins that are to function as analog inputs must have their corresponding TRIS bit set (input). If the TRIS bit is cleared (output), the digital output level (V<sub>OH</sub> or V<sub>OL</sub>) will be converted.

The ADPCFG and ADPCFG2 registers have a default value of 0x000; therefore, all pins that share ANx functions are analog (not digital) by default.

When the PORT register is read, 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 defined as a digital input (including the ANx pins) can cause the input buffer to consume current that exceeds the device specifications.

## 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. An example is shown in [Example 11-1](#).

## 11.5 Input Change Notification

The input change notification function of the I/O ports allows the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 devices to generate interrupt requests to the processor in response to a Change-Of-State (COS) on selected input pins. This feature can detect input Change-Of-States even in Sleep mode, when the clocks are disabled. Depending on the device pin count, up to 30 external signals (CNx pin) can be selected (enabled) for generating an interrupt request on a Change-Of-State.

Four control registers are associated with the CN module. The CNEN1 and CNEN2 registers contain the interrupt enable control bits for each of the CN input pins. Setting any of these bits enables a CN interrupt for the corresponding pins.

Each CN pin also has a weak pull-up connected to it. The pull-ups act as a current source connected to the pin, and eliminate the need for external resistors when the push button or keypad devices are connected. The pull-ups are enabled separately using the CNPU1 and CNPU2 registers, which contain the control bits 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 when the port pin is configured as a digital output.

### EQUATION 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
```

12.0 TIMER1

**Note 1:** This data sheet summarizes the features of the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 families of devices. 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](http://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 a time counter for the Real-Time Clock (RTC), or operate as a free-running interval timer/counter.

The Timer1 module has the following unique features over other timers:

- Can be operated from the low-power 32.767 kHz crystal oscillator available on the device
- Can be operated in Asynchronous Counter mode from an external clock source.
- The external clock input (T1CK) can optionally be synchronized to the internal device clock and the clock synchronization is performed after the prescaler.

The unique features of Timer1 allow it to be used for Real-Time Clock (RTC) applications. A block diagram of Timer1 is shown in [Figure 12-1](#).

The Timer1 module can operate in one of the following modes:

- Timer mode
- Gated Timer mode
- Synchronous Counter mode
- Asynchronous Counter mode

In Timer and Gated Timer modes, the input clock is derived from the internal instruction cycle clock (FCY). In Synchronous and Asynchronous Counter modes, the input clock is derived from the external clock input at the T1CK pin.

The Timer modes are determined by the following bits:

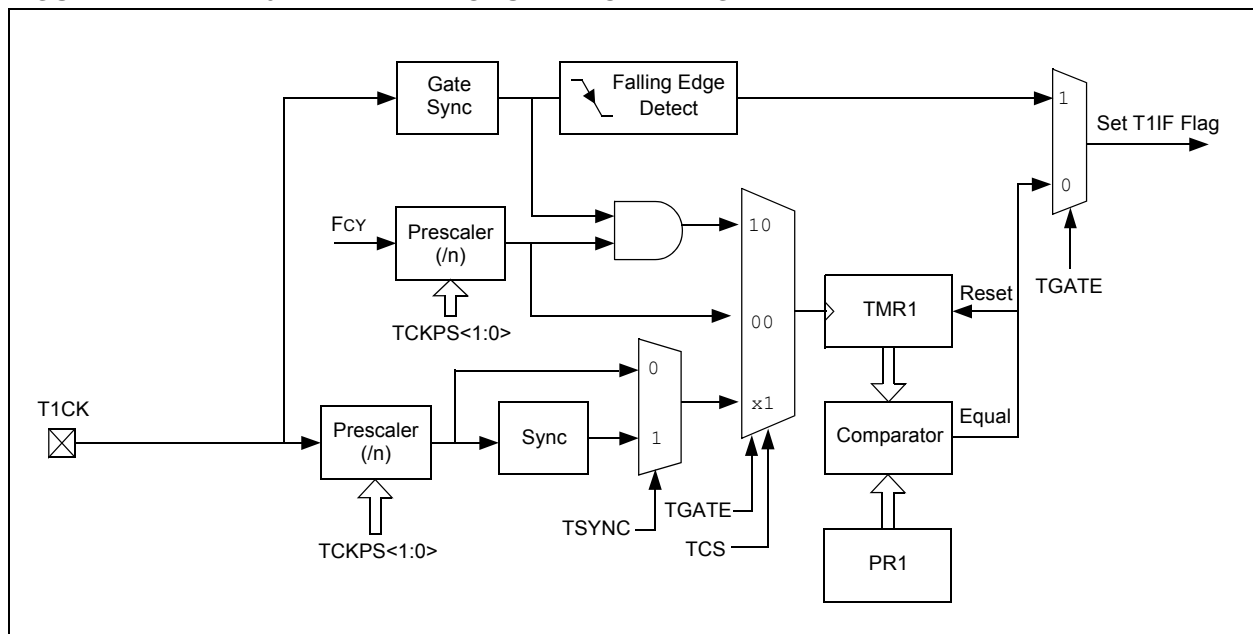
- Timer Clock Source Control bit: TCS (T1CON<1>)
- Timer Synchronization Control bit: TSYNC (T1CON<2>)
- Timer Gate Control bit: TGATE (T1CON<6>)

The timer control bit settings for different operating modes are given in the [Table 12-1](#).

**TABLE 12-1: TIMER MODE SETTINGS**

Mode	TCS	TGATE	TSYNC
Timer	0	0	x
Gated Timer	0	1	x
Synchronous Counter	1	x	1
Asynchronous Counter	1	x	0

**FIGURE 12-1: 16-BIT TIMER1 MODULE BLOCK DIAGRAM**



# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## 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 TCS = 1:  
             This bit is ignored.  
             When TCS = 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'

### 13.0 TIMER2/3/4/5 FEATURES

**Note 1:** This data sheet summarizes the features of the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 families of devices. 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](http://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.

- A Type B timer can be concatenated with a Type C timer to form a 32-bit timer
- External clock input (TxCK) is always synchronized to the internal device clock and the clock synchronization is performed after the prescaler.

Figure 13-1 shows a block diagram of the Type B timer. Timer3 and Timer5 are Type C timers that offer the following major features:

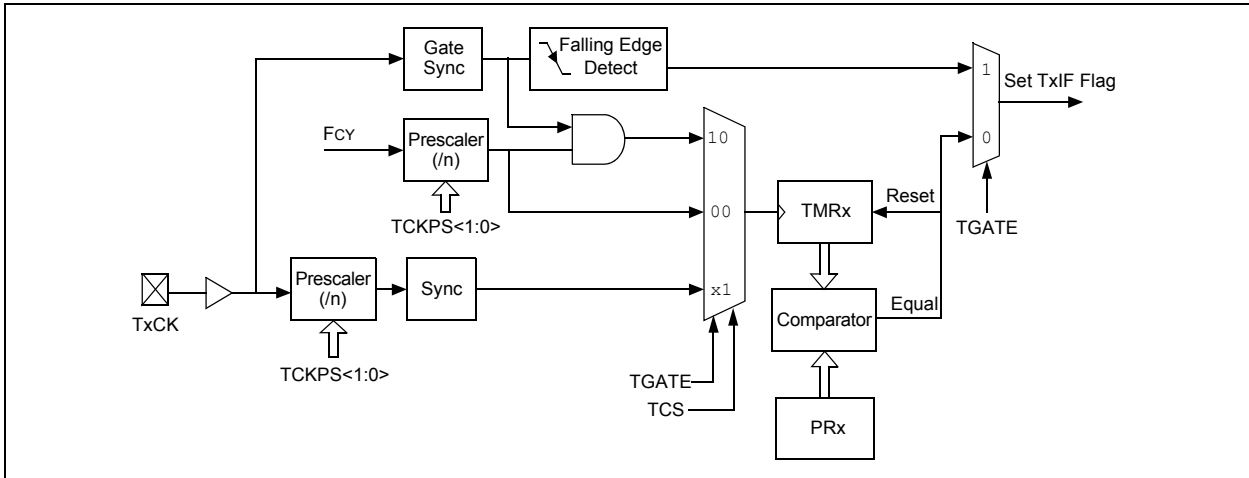
- A Type C timer can be concatenated with a Type B timer to form a 32-bit timer
- At least one Type C timer has the ability to trigger an analog-to-digital conversion.
- The external clock input (TxCK) is always synchronized to the internal device clock and the clock synchronization is performed before the prescaler

A block diagram of the Type C timer is shown in Figure 13-2.

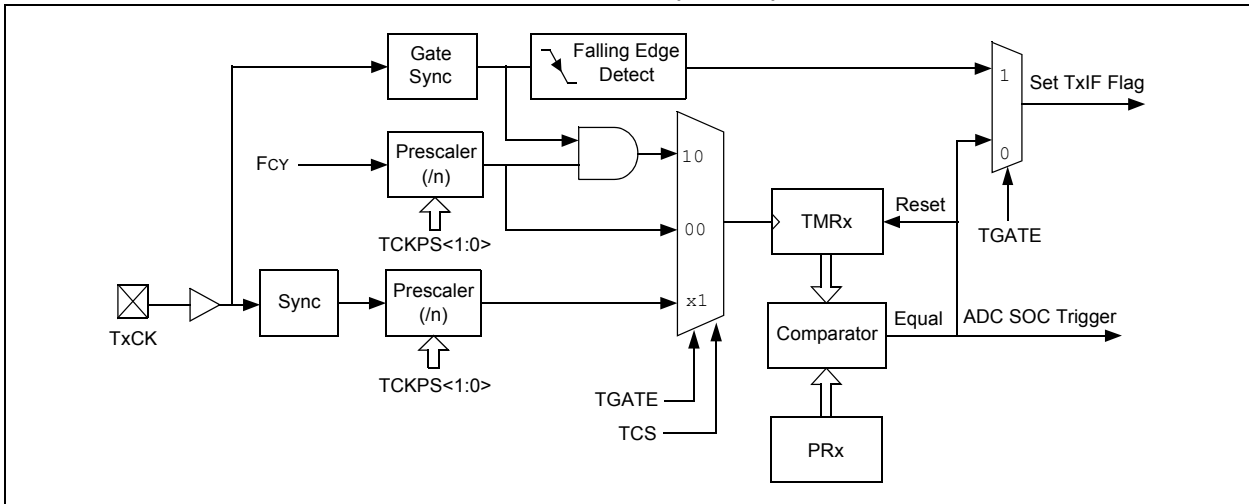
**Note:** Timer3 is not available on all devices.

Timer2 and Timer4 are Type B timers that offer the following major features:

**FIGURE 13-1: TYPE B TIMER BLOCK DIAGRAM (x = 2, 4)**



**FIGURE 13-2: TYPE C TIMER BLOCK DIAGRAM (x = 3, 5)**



The Timer2/3/4/5 modules can operate in one of the following modes:

- Timer mode
- Gated Timer mode
- Synchronous Counter mode

In Timer and Gated Timer modes, the input clock is derived from the internal instruction cycle clock (FCY). In Synchronous Counter mode, the input clock is derived from the external clock input at the TxCK pin.

The timer modes are determined by the following bits:

- TCS (TxCON<1>): Timer Clock Source Control bit
- TGATE (TxCON<6>): Timer Gate Control bit

Timer control bit settings for different operating modes are given in the [Table 13-1](#).

**TABLE 13-1: TIMER MODE SETTINGS**

Mode	TCS	TGATE
Timer	0	0
Gated Timer	0	1
Synchronous Counter	1	x

## 13.1 16-bit Operation

To configure any of the timers for individual 16-bit operation:

1. Clear the T32 bit corresponding to that timer.
2. Select the timer prescaler ratio using the TCKPS<1:0> bits.
3. 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.

## 13.2 32-bit Operation

A 32-bit timer module can be formed by combining a Type B and a Type C 16-bit timer module. For 32-bit timer operation, the T32 control bit in the Type B Timer Control (TxCON<3>) register must be set. The Type C timer holds the most significant word (msw) and the Type B timer holds the least significant word (lsw) for 32-bit operation.

When configured for 32-bit operation, only the Type B Timer Control (TxCON) register bits are required for setup and control while the Type C Timer Control register bits are ignored (except the TSIDL bit).

For interrupt control, the combined 32-bit timer uses the interrupt enable, interrupt flag and interrupt priority control bits of the Type C timer. The interrupt control and status bits for the Type B timer are ignored during 32-bit timer operation.

The timers that can be combined to form a 32-bit timer are listed in [Table 13-2](#).

**TABLE 13-2: 32-BIT TIMER**

Type B Timer (lsw)	Type C Timer (msw)
Timer2	Timer3
Timer4	Timer5

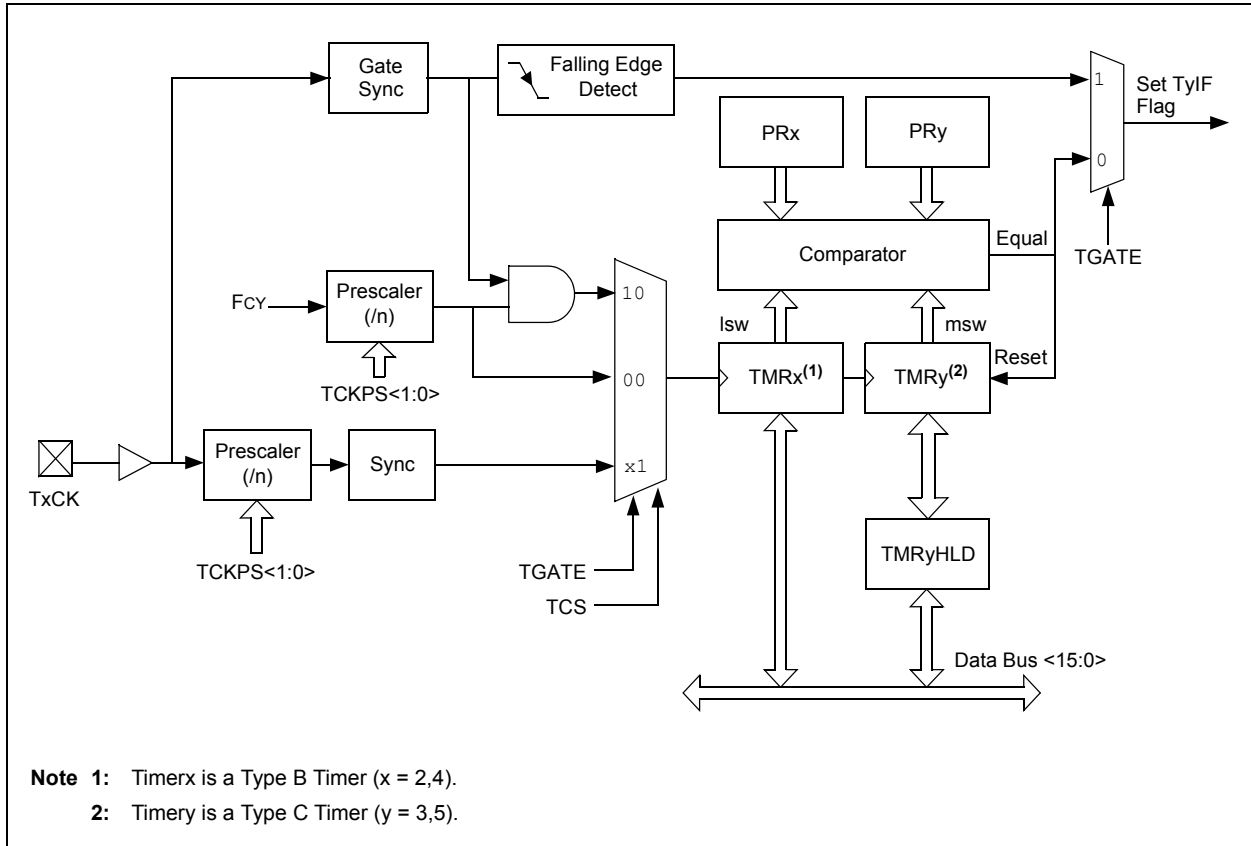
A block diagram representation of the 32-bit timer module is shown in [Figure 13-3](#). The 32-timer module can operate in one of the following modes:

- Timer mode
- Gated Timer mode
- Synchronous Counter mode

To configure the timer features for 32-bit operation:

1. Set the T32 control bit.
2. Select the prescaler ratio for Timer2 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 contains the most significant word of the value, while PR2 contains the least significant word.
5. If interrupts are required, set the interrupt enable bit, T3IE. Use the priority bits, T3IP<2:0>, to set the interrupt priority. While Timer2 controls the timer, the interrupt appears as a Timer3 interrupt.
6. Set the corresponding TON bit.

FIGURE 13-3: 32-BIT TIMER BLOCK DIAGRAM



# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 13-1: TxCON: TIMER CONTROL REGISTER (x = 2,4)

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<1:0>		T32	—	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:** Timerx On bit  
When T32 = 1 (in 32-bit Timer mode):  
 1 = Starts 32-bit TMRx:TMRy timer pair  
 0 = Stops 32-bit TMRx:TMRy timer pair  
When T32 = 0 (in 16-bit Timer mode):  
 1 = Starts 16-bit timer  
 0 = Stops 16-bit timer
- bit 14      **Unimplemented:** Read as '0'
- bit 13      **TSIDL:** Stop in Idle Mode bit  
 1 = Discontinue timer operation when device enters Idle mode  
 0 = Continue timer 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 enabled  
 0 = Gated time accumulation disabled
- bit 5-4     **TCKPS<1:0>:** Timerx 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
- bit 3        **T32:** 32-bit Timerx Mode Select bit  
 1 = TMRx and TMRy form a 32-bit timer  
 0 = TMRx and TMRy form separate 16-bit timer
- bit 2        **Unimplemented:** Read as '0'
- bit 1        **TCS:** Timerx Clock Source Select bit  
 1 = External clock from TxCK pin  
 0 = Internal clock (Fosc/2)
- bit 0        **Unimplemented:** Read as '0'



# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

**REGISTER 13-2: TyCON: TIMER CONTROL REGISTER (y = 3,5)**

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
TON <sup>(2)</sup>	—	TSIDL <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	U-0
—	TGATE <sup>(2)</sup>	TCKPS<1:0> <sup>(2)</sup>		—	—	TCS <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            **TON:** Timery On bit<sup>(2)</sup>  
                   1 = Starts 16-bit Timery  
                   0 = Stops 16-bit Timery
- bit 14            **Unimplemented:** Read as '0'
- bit 13            **TSIDL:** Stop in Idle Mode bit<sup>(1)</sup>  
                   1 = Discontinue timer operation when device enters Idle mode  
                   0 = Continue timer operation in Idle mode
- bit 12-7        **Unimplemented:** Read as '0'
- bit 6            **TGATE:** Timery Gated Time Accumulation Enable bit<sup>(2)</sup>  
                   When TCS = 1:  
                   This bit is ignored.  
                   When TCS = 0:  
                   1 = Gated time accumulation enabled  
                   0 = Gated time accumulation disabled
- bit 5-4        **TCKPS<1:0>:** Timery Input Clock Prescale Select bits<sup>(2)</sup>  
                   11 = 1:256 prescale value  
                   10 = 1:64 prescale value  
                   01 = 1:8 prescale value  
                   00 = 1:1 prescale value
- bit 3-2        **Unimplemented:** Read as '0'
- bit 1            **TCS:** Timery Clock Source Select bit<sup>(2)</sup>  
                   1 = External clock from TxCK pin  
                   0 = Internal clock (FOSC/2)
- bit 0            **Unimplemented:** Read as '0'

- Note 1:** 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.
- 2:** When the 32-bit timer operation is enabled (T32 = 1) in the Timer Control (TxCON<3>) register, these bits have no effect.

NOTES:

14.0 INPUT CAPTURE

**Note 1:** This data sheet summarizes the features of the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 families of devices. 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](http://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 dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 devices support up to two 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:

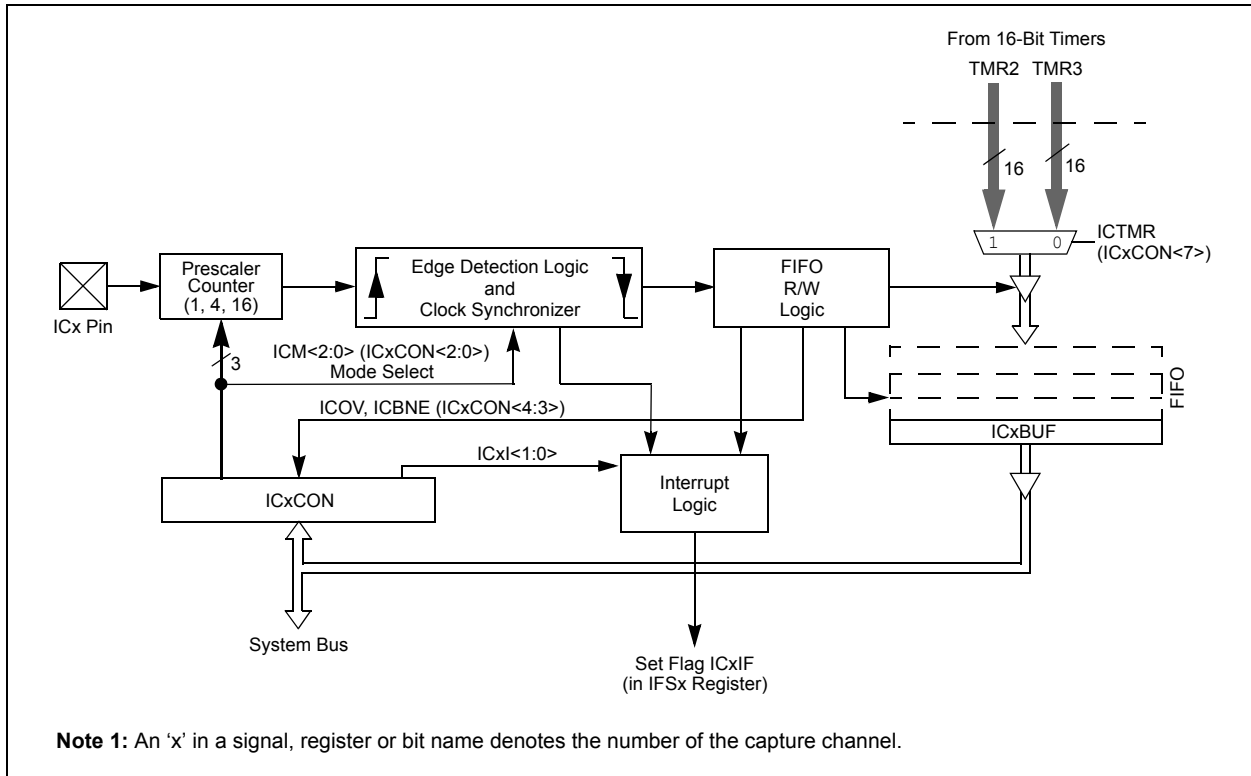
- Simple Capture Event modes:
  - Capture timer value on every falling edge of input at ICx pin
  - Capture timer value on every rising edge of input at ICx pin
- Capture timer value on every edge (rising and falling)
- Prescaler Capture Event modes:
  - Capture timer value on every 4th rising edge of input at ICx pin
  - Capture timer value on every 16th rising edge of input at ICx pin

Each input capture channel can select one of the 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:

- 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
- Use of input capture to provide additional sources of external interrupts

FIGURE 14-1: INPUT CAPTURE BLOCK DIAGRAM



## 14.1 Input Capture Registers

### REGISTER 14-1: ICxCON: INPUT CAPTURE x CONTROL REGISTER (x = 1, 2)

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	ICI<1:0>		ICOV	ICBNE	ICM<2:0>		
bit 7						bit 0	

<b>Legend:</b>	HC = Hardware Clearable bit
R = Readable bit	W = Writable bit
-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 halts in CPU Idle mode
  - 0 = Input capture module continues to operate in CPU Idle mode
- bit 12-8     **Unimplemented:** Read as '0'
- bit 7        **ICTMR:** Input Capture Timer Select bits
  - 1 = TMR2 contents are captured on capture event
  - 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 occurred
  - 0 = 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 edge
  - 100 = Capture mode, every 4th rising edge
  - 011 = Capture mode, every rising edge
  - 010 = 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 dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 families of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 13. “Output Compare”** (DS70209) in the “dsPIC33F/PIC24H Family Reference Manual”, which is available from the Microchip web site ([www.microchip.com](http://www.microchip.com)).

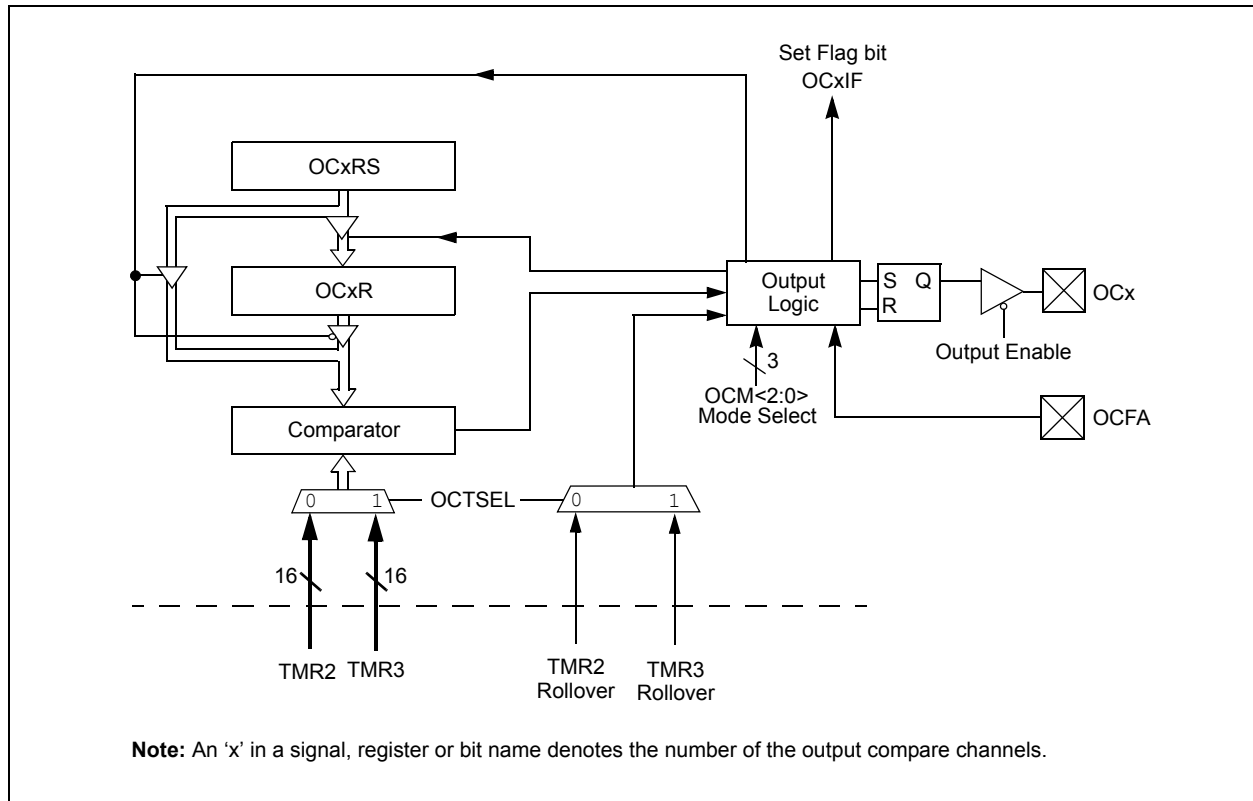
**2:** Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 “Memory Organization”** in this data sheet for device-specific register and bit information.

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 (OCM<2:0>) bits in the Output Compare Control (OCxCON<2:0>) register. 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

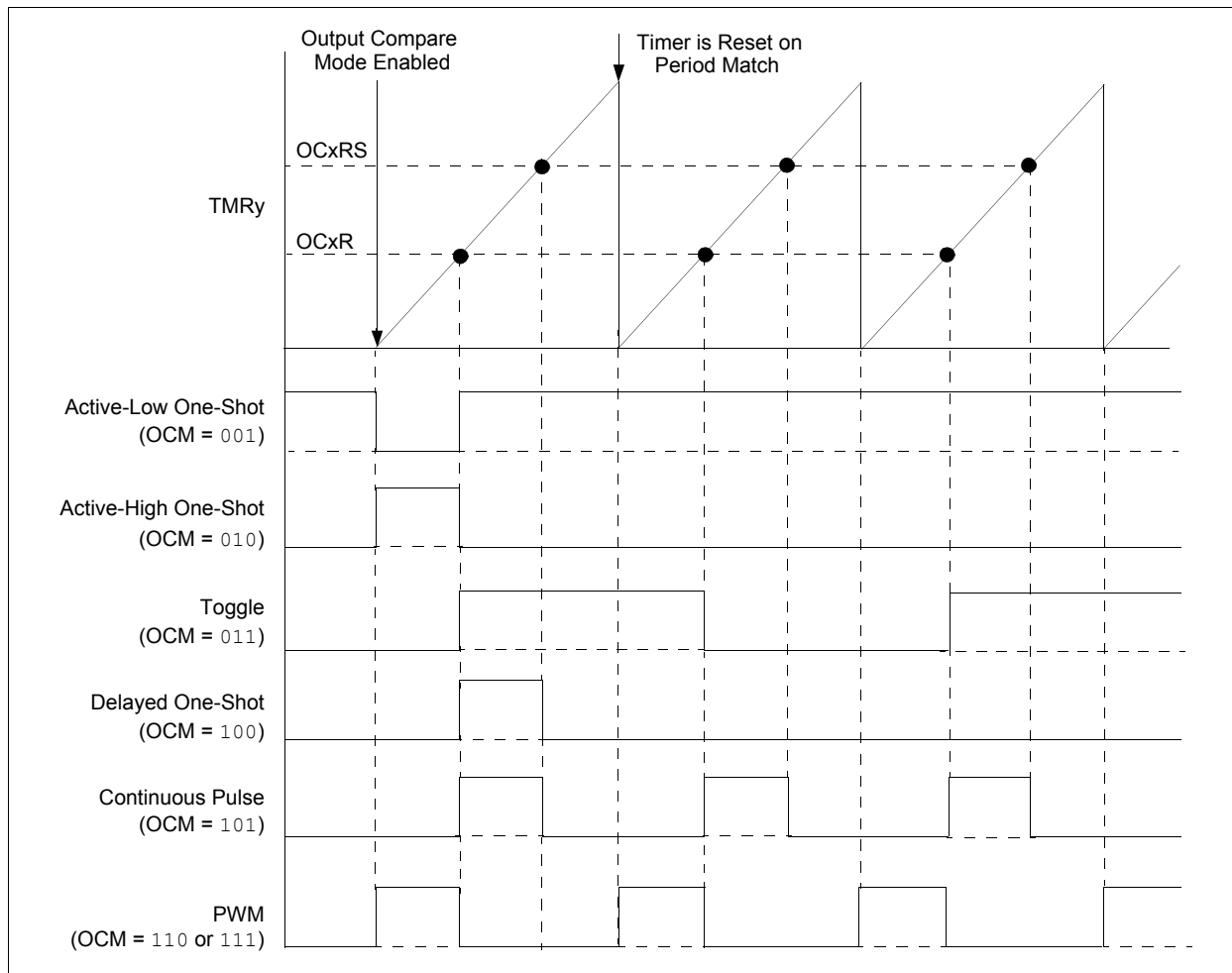
application must disable the associated timer when writing to the Output Compare Control registers to avoid malfunctions.

**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**



# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## 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	

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

- bit 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

NOTES:



## 16.0 HIGH-SPEED PWM

**Note 1:** This data sheet summarizes the features of the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 families of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 43. “High-Speed PWM”** (DS70323) in the “*dsPIC33F/PIC24H Family Reference Manual*”, which is available from the Microchip web site ([www.microchip.com](http://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 High-Speed PWM module on the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 devices supports a wide variety of PWM modes and output formats. This PWM module is ideal for power conversion applications, such as:

- AC/DC Converters
- DC/DC Converters
- Power Factor Correction
- Uninterruptible Power Supply (UPS)
- Inverters
- Battery Chargers
- Digital Lighting

### 16.1 Features Overview

The High-Speed PWM module incorporates the following features:

- Two master time base modules
- Up to nine PWM generators with up to 18 outputs
- Two PWM outputs per PWM generator
- Individual time base and duty cycle for each PWM output
- Duty cycle, dead time, phase shift, and frequency resolution of 1.04 ns
- Independent fault and current-limit inputs for eight PWM Outputs
- Redundant output
- True Independent output
- Center-Aligned PWM mode
- Output override control
- Chop mode (also known as Gated mode)
- Special Event Trigger
- Prescaler for input clock

- Dual trigger from PWM to Analog-to-Digital Converter (ADC) per PWM period
- PWMxL and PWMxH output pin swapping
- Independent PWM frequency, duty cycle, and phase shift changes
- Current compensation
- Enhanced Leading-Edge Blanking (LEB) functionality
- PWM Capture functionality

**Note:** Duty cycle, dead-time, phase shift and frequency resolution is 8.32 ns in Center-Aligned PWM mode.

Figure 16-1 conceptualizes the PWM module in a simplified block diagram. Figure 16-2 illustrates how the module hardware is partitioned for each PWM output pair for the Complementary PWM mode.

The PWM module contains nine PWM generators. The module has up to 18 PWM output pins: PWM1H/PWM1L through PWM9H/PWM9L. For complementary outputs, these 18 I/O pins are grouped into High/Low pairs.

### 16.2 Feature Description

The PWM module is designed for applications that require:

- High-resolution at high PWM frequencies
- The ability to drive Standard, Edge-Aligned, Center-Aligned Complementary mode, and Push-Pull mode outputs
- The ability to create multiphase PWM outputs

For Center-Aligned mode, the duty cycle, period phase and dead-time resolutions will be 8.32 ns.

Two common, medium power converter topologies are push-pull and half-bridge. These designs require the PWM output signal to be switched between alternate pins, as provided by the Push-Pull PWM mode.

Phase-shifted PWM describes the situation where each PWM generator provides outputs, but the phase relationship between the generator outputs is specifiable and changeable.

Multiphase PWM is often used to improve DC/DC converter load transient response, and reduce the size of output filter capacitors and inductors. Multiple DC/DC converters are often operated in parallel, but phase-shifted in time. A single PWM output operating at 250 kHz has a period of 4  $\mu$ s, but an array of four PWM channels, staggered by 1  $\mu$ s each, yields an effective switching frequency of 1 MHz. Multiphase PWM applications typically use a fixed-phase relationship.

Variable phase PWM is useful in Zero Voltage Transition (ZVT) power converters. Here, the PWM duty cycle is always 50%, and the power flow is controlled by varying the relative phase shift between the two PWM generators.

FIGURE 16-1: HIGH-SPEED PWM MODULE ARCHITECTURAL DIAGRAM

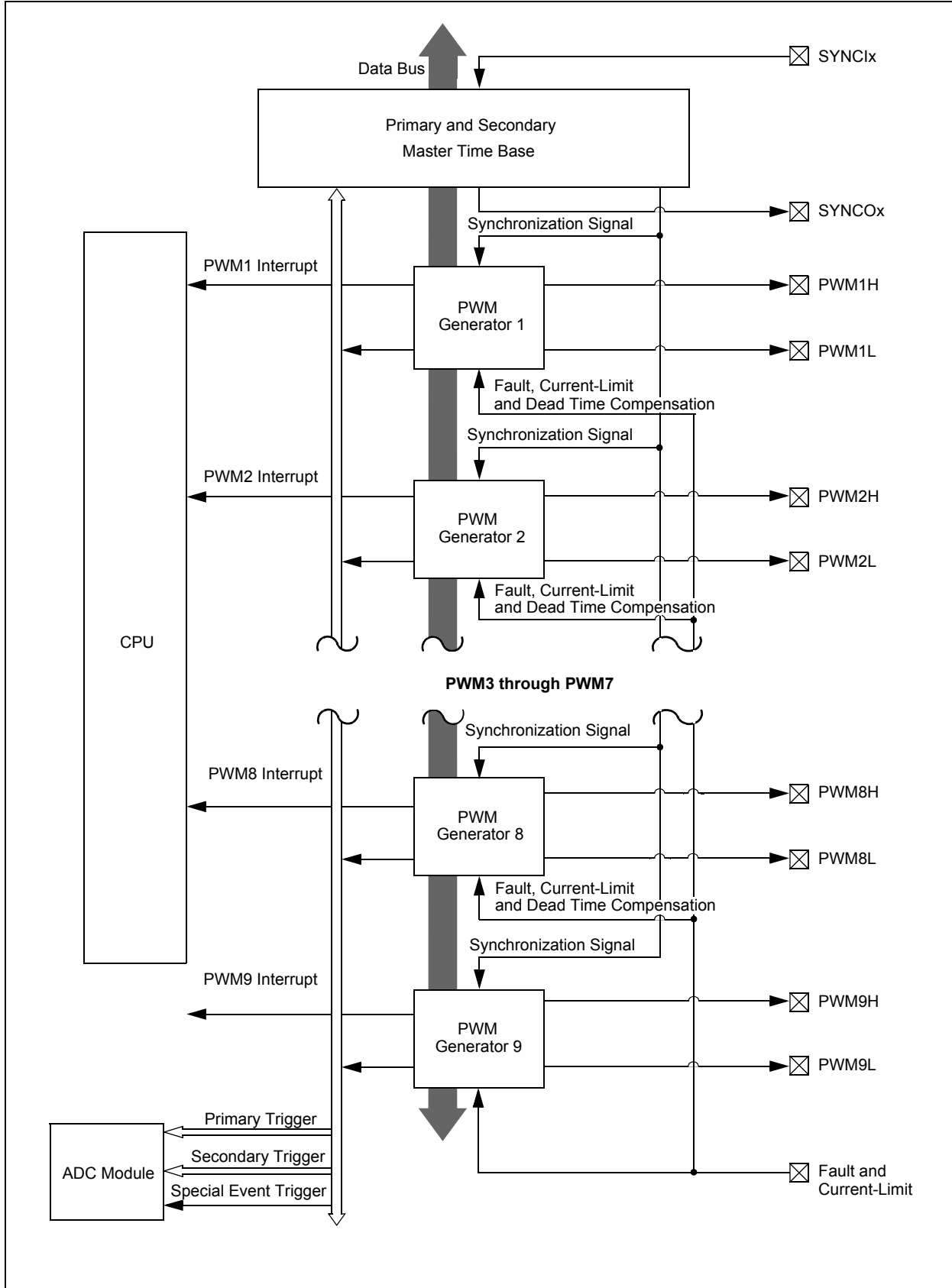
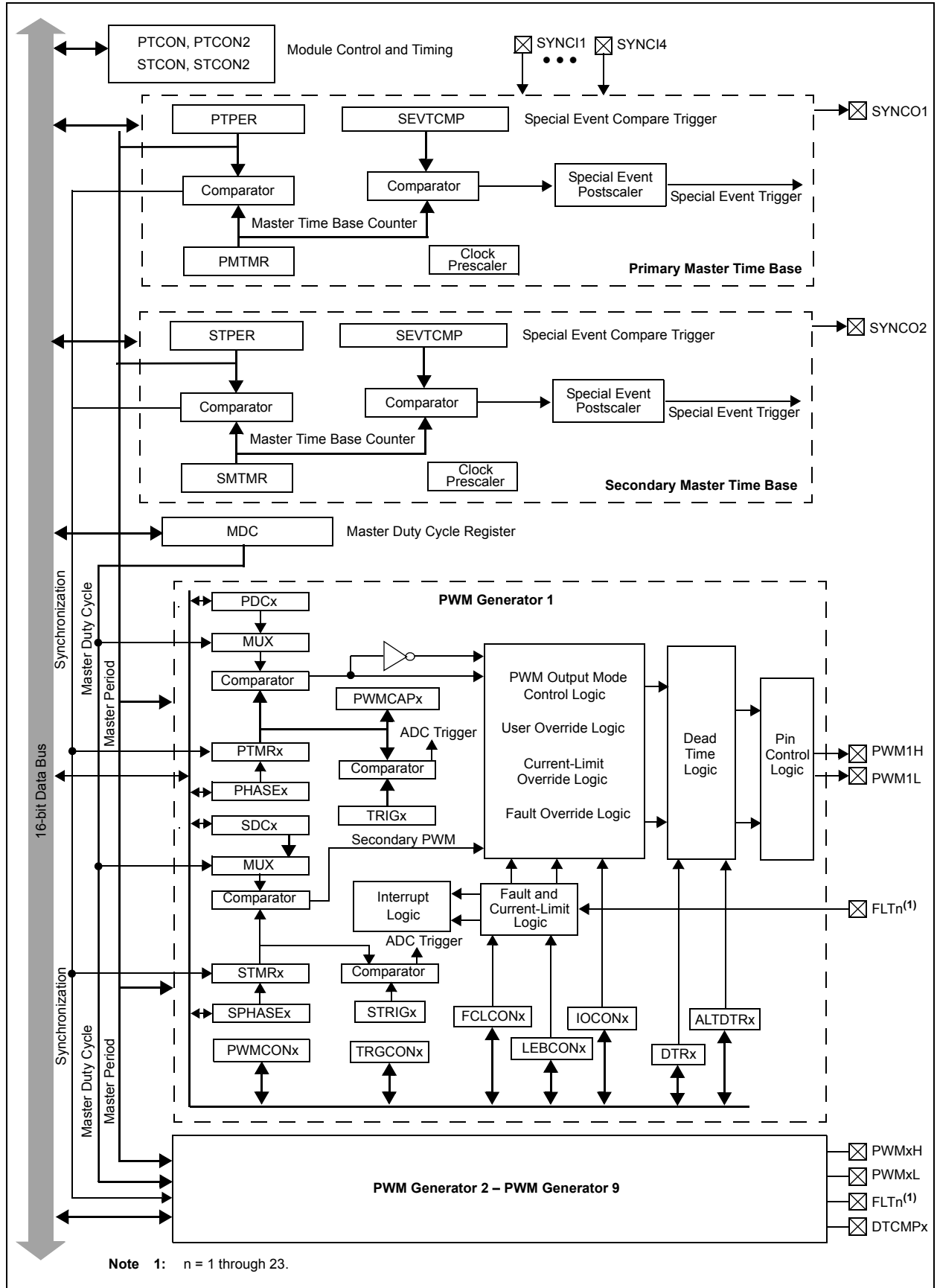


FIGURE 16-2: SIMPLIFIED CONCEPTUAL BLOCK DIAGRAM OF THE HIGH-SPEED PWM



## 16.3 Control Registers

The following registers control the operation of the High-Speed PWM module.

- PTCN: PWM Time Base Control Register
- PTCN2: PWM Clock Divider Select Register
- PTPER: Primary Master Time Base Period Register<sup>(1,2)</sup>
- SEVTCMP: PWM Special Event Compare Register<sup>(1)</sup>
- STCN: PWM Secondary Master Time Base Control Register
- STCN2: PWM Secondary Clock Divider Select Register 2
- STPER: Secondary Master Time Base Period Register
- SSEVTCMP: PWM Secondary Special Event Compare Register
- CHOP: PWM Chop Clock Generator Register
- MDC: PWM Master Duty Cycle Register
- PWMCONx: PWM Control Register
- PDCx: PWM Generator Duty Cycle Register
- PHASEx: PWM Primary Phase Shift Register
- DTRx: PWM Dead Time Register
- ALTDTRx: PWM Alternate Dead Time Register
- SDCx: PWM Secondary Duty Cycle Register
- SPHASEx: PWM Secondary Phase Shift Register
- TRGCONx: PWM Trigger Control Register
- IOCONx: PWM I/O Control Register
- FCLCONx: PWM Fault Current-Limit Control Register
- TRIGx: PWM Primary Trigger Compare Value Register
- STRIGx: PWM Secondary Trigger Compare Value Register<sup>(1)</sup>
- LEBCONx: Leading-Edge Blanking Control Register
- LEBDLYx: Leading-Edge Blanking Delay Register
- AUXCONx: PWM Auxiliary Control Register
- PWMCAPx: Primary PWM Time Base Capture Register

## REGISTER 16-1: PTCON: PWM TIME BASE CONTROL REGISTER

R/W-0	U-0	R/W-0	HS/HC-0	R/W-0	R/W-0	R/W-0	R/W-0
PTEN	—	PTSIDL	SESTAT	SEIEN	EIPU <sup>(1)</sup>	SYNCPOL <sup>(1)</sup>	SYNCOEN <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
SYNCEN <sup>(1)</sup>	SYNCSRC<2:0> <sup>(1)</sup>			SEVTPS<3:0> <sup>(1)</sup>			
bit 7						bit 0	

<b>Legend:</b>	HC = Cleared in Hardware	HS = Set in Hardware
R = Readable bit	W = Writable bit	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 Module Enable bit  
1 = PWM module is enabled  
0 = PWM module is disabled
- 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      **SESTAT:** Special Event Interrupt Status bit  
1 = Special Event Interrupt is pending  
0 = Special Event Interrupt is not pending
- bit 11      **SEIEN:** Special Event Interrupt Enable bit  
1 = Special Event Interrupt is enabled  
0 = Special Event Interrupt is disabled
- bit 10      **EIPU:** Enable Immediate Period Updates bit<sup>(1)</sup>  
1 = Active Period register is updated immediately  
0 = Active Period register updates occur on PWM cycle boundaries
- bit 9        **SYNCPOL:** Synchronize Input and Output Polarity bit<sup>(1)</sup>  
1 = SYNCIx/SYNCO1 polarity is inverted (active-low)  
0 = SYNCIx/SYNCO1 is active-high
- bit 8        **SYNCOEN:** Primary Time Base Sync Enable bit<sup>(1)</sup>  
1 = SYNCO1 output is enabled  
0 = SYNCO1 output is disabled
- bit 7        **SYNCEN:** External Time Base Synchronization Enable bit<sup>(1)</sup>  
1 = External synchronization of primary time base is enabled  
0 = External synchronization of primary time base is disabled
- bit 6-4      **SYNCSRC<2:0>:** Synchronous Source Selection bits<sup>(1)</sup>  
111 = Reserved  
101 = Reserved  
100 = Reserved  
011 = SYNCI4  
010 = SYNCI3  
001 = SYNCI2  
000 = SYNCI1

**Note 1:** These bits should be changed only when PTEN = 0. In addition, when using the SYNCIx feature, the user application must program the period register with a value that is slightly larger than the expected period of the external synchronization input signal.

## REGISTER 16-1: PTCON: PWM TIME BASE CONTROL REGISTER (CONTINUED)

bit 3-0      **SEVTPS<3:0>**: PWM Special Event Trigger Output Postscaler Select bits<sup>(1)</sup>

- 1111 = 1:16 Postscaler generates Special Event Trigger on every sixteenth compare match event
- 
- 
- 
- 0001 = 1:2 Postscaler generates Special Event Trigger on every second compare match event
- 0000 = 1:1 Postscaler generates Special Event Trigger on every compare match event

**Note 1:** These bits should be changed only when PTEN = 0. In addition, when using the SYNCIx feature, the user application must program the period register with a value that is slightly larger than the expected period of the external synchronization input signal.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 16-2: PTCON2: PWM CLOCK DIVIDER SELECT REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15						bit 8	

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	PCLKDIV<2:0> <sup>(1)</sup>		
bit 7					bit 0		

<b>Legend:</b>			
R = Readable bit	W = Writable bit	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      **PCLKDIV<2:0>:** PWM Input Clock Prescaler (Divider) Select bits<sup>(1)</sup>  
 111 = Reserved  
 110 = Divide by 64, maximum PWM timing resolution  
 101 = Divide by 32, maximum PWM timing resolution  
 100 = Divide by 16, maximum PWM timing resolution  
 011 = Divide by 8, maximum PWM timing resolution  
 010 = Divide by 4, maximum PWM timing resolution  
 001 = Divide by 2, maximum PWM timing resolution  
 000 = Divide by 1, maximum PWM timing resolution (power-on default)

**Note 1:** These bits should be changed only when PTEN = 0. Changing the clock selection during operation will yield unpredictable results.

## REGISTER 16-3: PTPER: PRIMARY MASTER TIME BASE PERIOD REGISTER<sup>(1,2)</sup>

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
PTPER<15:8>							
bit 15						bit 8	

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0
PTPER<7:0>							
bit 7					bit 0		

<b>Legend:</b>			
R = Readable bit	W = Writable bit	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      **PTPER<15:0>:** Primary Master Time Base (PMTMR) Period Value bits

**Note 1:** The PWM time base has a minimum value of 0x0010, and a maximum value of 0xFFFF8.  
**2:** Any Period value that is less than 0x0028 must have the least significant 3 bits set to '0', thus yielding a Period resolution at 8.32 ns (at fastest auxiliary clock rate).

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 16-4: SEVTCMP: PWM SPECIAL EVENT COMPARE 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
SEVTCMP<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
SEVTCMP<7:3>					—	—	—
bit 7							bit 0

**Legend:**

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

bit 15-3                      **SEVTCMP<15:3>**: Special Event Compare Count Value bits

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

**Note 1:** One LSB = 1.04 ns (at fastest auxiliary clock rate); therefore, the minimum SEVTCMP resolution is 8.32 ns.



## REGISTER 16-5: STCON: PWM SECONDARY MASTER TIME BASE CONTROL REGISTER

U-0	U-0	U-0	HS/HC-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	SESTAT	SEIEN	EIPU <sup>(1)</sup>	SYNCPOL	SYNCOEN
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
SYNCEN	SYNCSRC<2:0>			SEVTPS<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 15-13      **Unimplemented:** Read as '0'
- bit 12      **SESTAT:** Special Event Interrupt Status bit
  - 1 = Secondary Special Event Interrupt is pending
  - 0 = Secondary Special Event Interrupt is not pending
- bit 11      **SEIEN:** Special Event Interrupt Enable bit
  - 1 = Secondary Special Event Interrupt is enabled
  - 0 = Secondary Special Event Interrupt is disabled
- bit 10      **EIPU:** Enable Immediate Period Updates bit<sup>(1)</sup>
  - 1 = Active Secondary Period register is updated immediately
  - 0 = Active Secondary Period register updates occur on PWM cycle boundaries
- bit 9      **SYNCPOL:** Synchronize Input and Output Polarity bit
  - 1 = SYNCIx/SYNCO2 polarity is inverted (active-low)
  - 0 = SYNCIx/SYNCO2 polarity is active-high
- bit 8      **SYNCOEN:** Secondary Master Time Base Sync Enable bit
  - 1 = SYNCO2 output is enabled.
  - 0 = SYNCO2 output is disabled
- bit 7      **SYNCEN:** External Secondary Master Time Base Synchronization Enable bit
  - 1 = External synchronization of secondary time base is enabled
  - 0 = External synchronization of secondary time base is disabled
- bit 6-4      **SYNCSRC<2:0>:** Secondary Time Base Sync Source Selection bits
  - 111 = Reserved
  - 101 = Reserved
  - 100 = Reserved
  - 011 = SYNCI4
  - 010 = SYNCI3
  - 001 = SYNCI2
  - 000 = SYNCI1
- bit 3-0      **SEVTPS<3:0>:** PWM Secondary Special Event Trigger Output Postscaler Select bits
  - 1111 = 1:16 Postscale
  - 0001 = 1:2 Postscale
  - 
  - 
  - 
  - 0000 = 1:1 Postscale

**Note 1:** This bit only applies to the secondary master time base period.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 16-6: STCON2: PWM SECONDARY CLOCK DIVIDER SELECT 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	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	PCLKDIV<2:0> <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 15-3 **Unimplemented:** Read as '0'

bit 2-0 **PCLKDIV<2:0>:** PWM Input Clock Prescaler (Divider) Select bits<sup>(1)</sup>

111 = Reserved

110 = Divide by 64, maximum PWM timing resolution

101 = Divide by 32, maximum PWM timing resolution

100 = Divide by 16, maximum PWM timing resolution

011 = Divide by 8, maximum PWM timing resolution

010 = Divide by 4, maximum PWM timing resolution

001 = Divide by 2, maximum PWM timing resolution

000 = Divide by 1, maximum PWM timing resolution (power-on default)

**Note 1:** These bits should be changed only when PTEN = 0. Changing the clock selection during operation will yield unpredictable results.

## REGISTER 16-7: STPER: SECONDARY MASTER TIME BASE PERIOD 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
STPER<15:8>							
bit 15						bit 8	

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0
STPER<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 **STPER<15:0>:** Secondary Master Time Base (SMTMR) Period Value bits

## REGISTER 16-8: SSEVTCMP: PWM SECONDARY 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
SSEVTCMP<15:8>							
bit 15							bit 8
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
SSEVTCMP<7:3>					—	—	—
bit 7							bit 0

### Legend:

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

bit 15-3                      **SSEVTCMP<15:3>**: Special Event Compare Count Value bits

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

## REGISTER 16-9: CHOP: PWM CHOP CLOCK GENERATOR REGISTER

R/W-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
CHPCLKEN	—	—	—	—	—	CHOP<9:8>	
bit 15							bit 8
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
CHOP<7:3>					—	—	—
bit 7							bit 0

### Legend:

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

bit 15                      **CHPCLKEN**: Enable Chop Clock Generator bit

1 = Chop clock generator is enabled  
 0 = Chop clock generator is disabled

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

bit 9-3                      **CHOP<9:3>**: Chop Clock Divider bits

Value in 8.32 ns increments. The frequency of the chop clock signal is given by the following expression:

$$\text{Chop Frequency} = 1 / (16.64 * (\text{CHOP}<7:3> + 1) * \text{Primary Master PWM Input Clock Period})$$

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

**Note:** The chop clock generator operates with the primary PWM clock prescaler (PCLKDIV<2:0>) in the PTCON2 register ([Register 16-2](#)).

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 16-10: MDC: PWM MASTER DUTY CYCLE 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
MDC<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
MDC<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      **MDC<15:0>**: Master PWM Duty Cycle Value bits

**Note 1:** The smallest pulse width that can be generated on the PWM output corresponds to a value of 0x0008, while the maximum pulse width generated corresponds to a value of Period - 0x0008.

**2:** As the Duty Cycle gets closer to 0% or 100% of the PWM Period (0 to 40 ns, depending on the mode of operation), PWM Duty Cycle resolution will increase from 1 to 3 LSBs.

## REGISTER 16-11: PWMCONx: PWM CONTROL REGISTER

HS/HC-0	HS/HC-0	HS/HC-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
FLTSTAT <sup>(1)</sup>	CLSTAT <sup>(1)</sup>	TRGSTAT	FLTIEN	CLIEN	TRGIEN	ITB <sup>(3)</sup>	MDCS <sup>(3)</sup>
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
DTC<1:0>		DTCP <sup>(4)</sup>	—	MTBS	CAM <sup>(2,3,5)</sup>	XPRES <sup>(6)</sup>	IUE
bit 7						bit 0	

<b>Legend:</b>	HC = Cleared in Hardware	HS = Set in Hardware
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared
		x = Bit is unknown

- bit 15      **FLTSTAT:** Fault Interrupt Status bit<sup>(1)</sup>  
 1 = Fault interrupt is pending  
 0 = No Fault interrupt is pending  
 This bit is cleared by setting FLTIEN = 0.
- bit 14      **CLSTAT:** Current-Limit Interrupt Status bit<sup>(1)</sup>  
 1 = Current-limit interrupt is pending  
 0 = No current-limit interrupt is pending  
 This bit is cleared by setting CLIEN = 0.
- bit 13      **TRGSTAT:** Trigger Interrupt Status bit  
 1 = Trigger interrupt is pending  
 0 = No trigger interrupt is pending  
 This bit is cleared by setting TRGIEN = 0.
- bit 12      **FLTIEN:** Fault Interrupt Enable bit  
 1 = Fault interrupt is enabled  
 0 = Fault interrupt is disabled and FLTSTAT bit is cleared
- bit 11      **CLIEN:** Current-Limit Interrupt Enable bit  
 1 = Current-limit interrupt enabled  
 0 = Current-limit interrupt disabled and CLSTAT bit is cleared
- bit 10      **TRGIEN:** Trigger Interrupt Enable bit  
 1 = A trigger event generates an interrupt request  
 0 = Trigger event interrupts are disabled and TRGSTAT bit is cleared
- bit 9        **ITB:** Independent Time Base Mode bit<sup>(3)</sup>  
 1 = PHASEx/SPHASEx registers provide time base period for this PWM generator  
 0 = PTPER register provides timing for this PWM generator
- bit 8        **MDCS:** Master Duty Cycle Register Select bit<sup>(3)</sup>  
 1 = MDC register provides duty cycle information for this PWM generator  
 0 = PDCx and SDCx registers provide duty cycle information for this PWM generator

- Note 1:** Software must clear the interrupt status here, and in the corresponding IFS bit in the Interrupt Controller.
- 2:** The Independent Time Base mode (ITB = 1) must be enabled to use Center-Aligned mode. If ITB = 0, the CAM bit is ignored.
- 3:** These bits should not be changed after the PWM is enabled by setting PTEN = 1 (PTCON<15>).
- 4:** For DTCP to be effective, DTC<1:0> must be set to '11'; otherwise, DTCP is ignored.
- 5:** Center-Aligned mode ignores the least significant 3 bits of the duty cycle, phase, and dead time registers. The highest Center-Aligned mode resolution available is 8.32 ns with the clock prescaler set to the fastest clock.
- 6:** Configure CLMOD = 0 (FCLCONX<8>) and ITB = 1 (PWMCONx<9>) to operate in External Period Reset mode.

## REGISTER 16-11: PWMCONx: PWM CONTROL REGISTER (CONTINUED)

bit 7-6	<b>DTC&lt;1:0&gt;</b> : Dead Time Control bits 11 = Dead Time Compensation mode 10 = Dead time function is disabled 01 = Negative dead time actively applied for Complementary Output mode 00 = Positive dead time actively applied for all output modes
bit 5	<b>DTCP</b> : Dead Time Compensation Polarity bit <sup>(4)</sup> 1 = If DTCMPx = 0, PWMxL is shortened, and PWMxH is lengthened If DTCMPx = 1, PWMxH is shortened, and PWMxL is lengthened 0 = If DTCMPx = 0, PWMxH is shortened, and PWMLx is lengthened If DTCMPx = 1, PWMxL is shortened, and PWMxH is lengthened
bit 4	<b>Unimplemented</b> : Read as '0'
bit 3	<b>MTBS</b> : Master Time Base Select bit 1 = PWM generator uses the secondary master time base for synchronization and the clock source for the PWM generation logic (if secondary time base is available) 0 = PWM generator uses the primary master time base for synchronization and the clock source for the PWM generation logic
bit 2	<b>CAM</b> : Center-Aligned Mode Enable bit <sup>(2,3,5)</sup> 1 = Center-Aligned mode is enabled 0 = Edge-Aligned mode is enabled
bit 1	<b>XPRES</b> : External PWM Reset Control bit <sup>(6)</sup> 1 = Current-limit source resets the time base for this PWM generator if it is in Independent Time Base mode 0 = External pins do not affect PWM time base
bit 0	<b>IUE</b> : Immediate Update Enable bit 1 = Updates to the active MDC/PDCx/SDCx registers are immediate 0 = Updates to the active PDCx registers are synchronized to the PWM time base

- Note 1:** Software must clear the interrupt status here, and in the corresponding IFS bit in the Interrupt Controller.
- 2:** The Independent Time Base mode (ITB = 1) must be enabled to use Center-Aligned mode. If ITB = 0, the CAM bit is ignored.
- 3:** These bits should not be changed after the PWM is enabled by setting PTEN = 1 (PTCON<15>).
- 4:** For DTCP to be effective, DTC<1:0> must be set to '11'; otherwise, DTCP is ignored.
- 5:** Center-Aligned mode ignores the least significant 3 bits of the duty cycle, phase, and dead time registers. The highest Center-Aligned mode resolution available is 8.32 ns with the clock prescaler set to the fastest clock.
- 6:** Configure CLMOD = 0 (FCLCONX<8>) and ITB = 1 (PWMCONx<9>) to operate in External Period Reset mode.

## REGISTER 16-12: PDCx: PWM GENERATOR DUTY CYCLE 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
PDCx<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
PDCx<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      **PDCx<15:0>**: PWM Generator # Duty Cycle Value bits

- Note 1:** In Independent PWM mode, the PDCx register controls the PWMxH duty cycle only. In the Complementary, Redundant and Push-Pull PWM modes, the PDCx register controls the duty cycle of both the PWMxH and PWMxL.
- 2:** The smallest pulse width that can be generated on the PWM output corresponds to a value of 0x0008, while the maximum pulse width generated corresponds to a value of Period - 0x0008.
- 3:** As the Duty Cycle gets closer to 0% or 100% of the PWM Period (0 to 40 ns, depending on the mode of operation), PWM Duty Cycle resolution will increase from 1 to 3 LSBs.

## REGISTER 16-13: SDCx: PWM SECONDARY DUTY CYCLE 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
SDCx<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
SDCx<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      **SDCx<15:0>**: Secondary Duty Cycle bits for PWMxL Output Pin

- Note 1:** The SDCx register is used in Independent PWM mode only. When used in Independent PWM mode, the SDCx register controls the PWMxL duty cycle.
- 2:** The smallest pulse width that can be generated on the PWM output corresponds to a value of 0x0008, while the maximum pulse width generated corresponds to a value of Period - 0x0008.
- 3:** As the Duty Cycle gets closer to 0% or 100% of the PWM Period (0 to 40 ns, depending on the mode of operation), PWM Duty Cycle resolution will increase from 1 to 3 LSBs.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 16-14: PHASE<sub>x</sub>: PWM PRIMARY PHASE SHIFT 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
PHASE <sub>x</sub> <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
PHASE <sub>x</sub> <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      **PHASE<sub>x</sub><15:0>**: PWM Phase Shift Value or Independent Time Base Period bits for the PWM Generator

**Note 1:** If PWMCON<sub>x</sub><9> = 0, the following applies based on the mode of operation:

- Complementary, Redundant and Push-Pull Output mode (IOCON<sub>x</sub><10:8> = 00, 01, or 10) PHASE<sub>x</sub><15:0> = Phase shift value for PWM<sub>x</sub>H and PWM<sub>x</sub>L outputs
- True Independent Output mode (IOCON<sub>x</sub><10:8> = 11) PHASE<sub>x</sub><15:0> = Phase shift value for PWM<sub>x</sub>H only
- When the PHASE<sub>x</sub>/SPHASE<sub>x</sub> register provides the phase shift with respect to the master time base; therefore, the valid range is 0x0000 through Period.

**2:** If PWMCON<sub>x</sub><9> = 1, the following applies based on the mode of operation:

- Complementary, Redundant, and Push-Pull Output mode (IOCON<sub>x</sub><10:8> = 00, 01, or 10) PHASE<sub>x</sub><15:0> = Independent time base period value for PWM<sub>x</sub>H and PWM<sub>x</sub>L
- True Independent Output mode (IOCON<sub>x</sub><10:8> = 11) PHASE<sub>x</sub><15:0> = Independent time base period value for PWM<sub>x</sub>H only
- When the PHASE<sub>x</sub>/SPHASE<sub>x</sub> register provides the local period, the valid range is 0x0000 through 0xFFF8.



**REGISTER 16-15: SPHASE<sub>x</sub>: PWM SECONDARY PHASE SHIFT 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
SPHASE <sub>x</sub> <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
SPHASE <sub>x</sub> <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                      **SPHASE<sub>x</sub><15:0>**: Secondary Phase Offset bits for PWM<sub>x</sub>L Output Pin (used in Independent PWM mode only)

**Note 1:** If PWMCON<sub>x</sub><9> = 0, the following applies based on the mode of operation:

- Complementary, Redundant and Push-Pull Output mode (IOCON<sub>x</sub><10:8> = 00, 01, or 10) SPHASE<sub>x</sub><15:0> = Not used
- True Independent Output mode (IOCON<sub>x</sub><10:8> = 11) PHASE<sub>x</sub><15:0> = Phase shift value for PWM<sub>x</sub>L only

**2:** If PWMCON<sub>x</sub><9> = 1, the following applies based on the mode of operation:

- Complementary, Redundant and Push-Pull Output mode (IOCON<sub>x</sub><10:8> = 00, 01, or 10) SPHASE<sub>x</sub><15:0> = Not used
- True Independent Output mode (IOCON<sub>x</sub><10:8> = 11) PHASE<sub>x</sub><15:0> = Independent time base period value for PWM<sub>x</sub>L only
- When the PHASE<sub>x</sub>/SPHASE<sub>x</sub> register provides the local period, the valid range of values is 0x0010-0xFFFF8.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 16-16: DTRx: PWM DEAD TIME REGISTER

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	DTRx<13:8>					
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
DTRx<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 15-14      **Unimplemented:** Read as '0'

bit 13-0      **DTRx<13:0>:** Unsigned 14-bit Dead Time Value bits for PWMx Dead Time Unit

## REGISTER 16-17: ALTDTRx: PWM ALTERNATE DEAD TIME REGISTER

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	ALTDTRx<13:8>					
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
ALTDTRx<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 15-14      **Unimplemented:** Read as '0'

bit 13-0      **ALTDTRx<13:0>:** Unsigned 14-bit Dead Time Value bits for PWMx Dead Time Unit

**REGISTER 16-18: TRGCONx: PWM TRIGGER CONTROL REGISTER**

R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
TRGDIV<3:0>				—	—	—	—
bit 15				bit 8			

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DTM <sup>(1)</sup>	—	TRGSTRT<5:0>					
bit 7		bit 0					

<b>Legend:</b>			
R = Readable bit	W = Writable bit	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     **TRGDIV<3:0>**: Trigger # Output Divider bits  
 1111 = Trigger output for every 16th trigger event  
 1110 = Trigger output for every 15th trigger event  
 1101 = Trigger output for every 14th trigger event  
 1100 = Trigger output for every 13th trigger event  
 1011 = Trigger output for every 12th trigger event  
 1010 = Trigger output for every 11th trigger event  
 1001 = Trigger output for every 10th trigger event  
 1000 = Trigger output for every 9th trigger event  
 0111 = Trigger output for every 8th trigger event  
 0110 = Trigger output for every 7th trigger event  
 0101 = Trigger output for every 6th trigger event  
 0100 = Trigger output for every 5th trigger event  
 0011 = Trigger output for every 4th trigger event  
 0010 = Trigger output for every 3rd trigger event  
 0001 = Trigger output for every 2nd trigger event  
 0000 = Trigger output for every trigger event

bit 11-8     **Unimplemented**: Read as '0'

bit 7     **DTM**: Dual Trigger Mode bit<sup>(1)</sup>  
 1 = Secondary trigger event is combined with the primary trigger event to create PWM trigger  
 0 = Secondary trigger event is not combined with the primary trigger event to create PWM trigger. Two separate PWM triggers are generated.

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

bit 5-0     **TRGSTRT<5:0>**: Trigger Postscaler Start Enable Select bits  
 111111 = Wait 63 PWM cycles before generating the first trigger event after the module is enabled  
 •  
 •  
 •  
 000010 = Wait 2 PWM cycles before generating the first trigger event after the module is enabled  
 000001 = Wait 1 PWM cycles before generating the first trigger event after the module is enabled  
 000000 = Wait 0 PWM cycles before generating the first trigger event after the module is enabled

**Note 1:** The secondary PWM generator cannot generate PWM trigger interrupts.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 16-19: IOCONx: PWM I/O CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PENH	PENL	POLH	POLL	PMOD<1:0> <sup>(1)</sup>		OVRENH	OVRENL
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
OVRDAT<1:0>		FLTDAT<1:0>		CLDAT<1:0>		SWAP	OSYNC
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            **PENH:** PWMxH Output Pin Ownership bit  
                   1 = PWM module controls PWMxH pin  
                   0 = GPIO module controls PWMxH pin
- bit 14            **PENL:** PWMxL Output Pin Ownership bit  
                   1 = PWM module controls PWMxL pin  
                   0 = GPIO module controls PWMxL pin
- bit 13            **POLH:** PWMxH Output Pin Polarity bit  
                   1 = PWMxH pin is active-low  
                   0 = PWMxH pin is active-high
- bit 12            **POLL:** PWMxL Output Pin Polarity bit  
                   1 = PWMxL pin is active-low  
                   0 = PWMxL pin is active-high
- bit 11-10        **PMOD<1:0>:** PWM # I/O Pin Mode bits<sup>(1)</sup>  
                   11 = PWM I/O pin pair is in the True Independent Output mode  
                   10 = PWM I/O pin pair is in the Push-Pull Output mode  
                   01 = PWM I/O pin pair is in the Redundant Output mode  
                   00 = PWM I/O pin pair is in the Complementary Output mode
- bit 9             **OVRENH:** Override Enable for PWMxH Pin bit  
                   1 = OVRDAT<1> provides data for output on PWMxH pin  
                   0 = PWM generator provides data for PWMxH pin
- bit 8             **OVRENL:** Override Enable for PWMxL Pin bit  
                   1 = OVRDAT<0> provides data for output on PWMxL pin  
                   0 = PWM generator provides data for PWMxL pin
- bit 7-6          **OVRDAT<1:0>:** Data for PWMxH, PWMxL Pins if Override is Enabled bits  
                   If OVERENH = 1, OVRDAT<1> provides data for PWMxH  
                   If OVERENL = 1, OVRDAT<0> provides data for PWMxL
- bit 5-4          **FLTDAT<1:0>:** State<sup>(2)</sup> for PWMxH and PWMxL Pins if FLTMOD is Enabled bits  
                   FCLCONx<IFLTMOD> = 0: Normal Fault mode  
                   If Fault active, then FLTDAT<1> provides state for PWMxH  
                   If Fault active, then FLTDAT<0> provides state for PWMxL  
  
                   FCLCONx<IFLTMOD> = 1: Independent Fault mode  
                   If Current-Limit active, then FLTDAT<1> provides data for PWMxH  
                   If Fault active, then FLTDAT<0> provides state for PWMxL

- Note 1:** These bits should not be changed after the PWM module is enabled (PTEN = 1).  
**Note 2:** State represents the active/inactive state of the PWM depending on the POLH and POLL bit settings.

## REGISTER 16-19: IOCONx: PWM I/O CONTROL REGISTER (CONTINUED)

- bit 3-2      **CLDAT<1:0>**: State<sup>(2)</sup> for PWMxH and PWMxL Pins if CLMOD is Enabled bits  
FCLCONx<IFLTMOD> = 0: Normal Fault mode  
If current-limit active, then CLDAT<1> provides state for PWMxH  
If current-limit active, then CLDAT<0> provides state for PWMxL
- FCLCONx<IFLTMOD> = 1: Independent Fault mode  
CLDAT<1:0> is ignored
- bit 1      **SWAP**: SWAP PWMxH and PWMxL pins bit  
1 = PWMxH output signal is connected to PWMxL pins; PWMxL output signal is connected to PWMxH pins  
0 = PWMxH and PWMxL pins are mapped to their respective pins
- bit 0      **OSYNC**: Output Override Synchronization bit  
1 = Output overrides via the OVRDAT<1:0> bits are synchronized to the PWM time base  
0 = Output overrides via the OVDDAT<1:0> bits occur on next CPU clock boundary

- Note 1:** These bits should not be changed after the PWM module is enabled (PTEN = 1).  
**2:** State represents the active/inactive state of the PWM depending on the POLH and POLL bit settings.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 16-20: TRIGx: PWM PRIMARY TRIGGER COMPARE VALUE 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
TRGCMP<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
TRGCMP<7:3>					—	—	—
bit 7							bit 0

**Legend:**

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

- bit 15-3      **TRGCMP<15:3>**: Trigger Compare Value bits  
When the primary PWM functions in local time base, this register contains the compare values that can trigger the ADC module.
- bit 2-0      **Unimplemented**: Read as '0'

**REGISTER 16-21: FCLCONx: PWM FAULT CURRENT-LIMIT CONTROL REGISTER**

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IFLTMOD	CLSRC<4:0> <sup>(2,3)</sup>					CLPOL <sup>(1)</sup>	CLMOD
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
FLTSRC<4:0> <sup>(2,3)</sup>					FLTPOL <sup>(1)</sup>	FLTMOD<1:0>	
bit 7							bit 0

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

bit 15      **IFLTMOD:** Independent Fault Mode Enable bit  
 1 = Independent Fault mode: Current-limit input maps FLTDAT<1> to PWMxH output, and Fault input maps FLTDAT<0> to PWMxL output. The CLDAT<1:0> bits are not used for override functions.  
 0 = Normal Fault mode: Current-Limit mode maps CLDAT<1:0> bits to the PWMxH and PWMxL outputs. The PWM Fault mode maps FLTDAT<1:0> to the PWMxH and PWMxL outputs.

bit 14-10    **CLSRC<4:0>:** Current-Limit Control Signal Source Select bits for PWM Generator #<sup>(2,4)</sup>. These bits also specify the source for the dead time compensation input signal, DTCMPx.  
 11111 = Reserved  
 11110 = Fault 23  
 11101 = Fault 22  
 11100 = Fault 21  
 11011 = Fault 20  
 11010 = Fault 19  
 11001 = Fault 18  
 11000 = Fault 17  
 10111 = Fault 16  
 10110 = Fault 15  
 10101 = Fault 14  
 10100 = Fault 13  
 10011 = Fault 12  
 10010 = Fault 11  
 10001 = Fault 10  
 10000 = Fault 9  
 01111 = Fault 8  
 01110 = Fault 7  
 01101 = Fault 6  
 01100 = Fault 5  
 01011 = Fault 4  
 01010 = Fault 3  
 01001 = Fault 2  
 01000 = Fault 1  
 00111 = Reserved  
 00110 = Reserved  
 00101 = Reserved  
 00100 = Reserved  
 00011 = Analog Comparator 4  
 00010 = Analog Comparator 3  
 00001 = Analog Comparator 2  
 00000 = Analog Comparator 1

- Note 1:** These bits should be changed only when PTEN = 0 (PTCON<15>).  
**Note 2:** When Independent Fault mode is enabled (IFLTMOD = 1), and Fault 1 is used for Current-Limit mode (CLSRC<4:0> = b0000), the Fault Control Source Select bits (FLTSRC<4:0>) should be set to an unused Fault source to prevent Fault 1 from disabling both the PWMxL and PWMxH outputs.  
**Note 3:** When Independent Fault mode is enabled (IFLTMOD = 1) and Fault 1 is used for Fault mode (FLTSRC<4:0> = b0000), the Current-Limit Control Source Select bits (CLSRC<4:0>) should be set to an unused current-limit source to prevent the current-limit source from disabling both the PWMxH and PWMxL outputs.

## REGISTER 16-21: FCLCONx: PWM FAULT CURRENT-LIMIT CONTROL REGISTER (CONTINUED)

bit 9	<b>CLPOL:</b> Current-Limit Polarity bit for PWM Generator # <sup>(1)</sup> 1 = The selected current-limit source is active-low 0 = The selected current-limit source is active-high
bit 8	<b>CLMOD:</b> Current-Limit Mode Enable bit for PWM Generator # 1 = Current-Limit mode is enabled 0 = Current-Limit mode is disabled
bit 7-3	<b>FLTSRC&lt;4:0&gt;:</b> Fault Control Signal Source Select bits for PWM Generator # <sup>(2,4)</sup> 11111 = Reserved 11110 = Fault 23 11101 = Fault 22 11100 = Fault 21 11011 = Fault 20 11010 = Fault 19 11001 = Fault 18 11000 = Fault 17 10111 = Fault 16 10110 = Fault 15 10101 = Fault 14 10100 = Fault 13 10011 = Fault 12 10010 = Fault 11 10001 = Fault 10 10000 = Fault 9 01111 = Fault 8 01110 = Fault 7 01101 = Fault 6 01100 = Fault 5 01011 = Fault 4 01010 = Fault 3 01001 = Fault 2 01000 = Fault 1 00111 = Reserved 00110 = Reserved 00101 = Reserved 00100 = Reserved 00011 = Analog Comparator 4 00010 = Analog Comparator 3 00001 = Analog Comparator 2 00000 = Analog Comparator 1
bit 2	<b>FLTPOL:</b> Fault Polarity bit for PWM Generator # <sup>(1)</sup> 1 = The selected Fault source is active-low 0 = The selected Fault source is active-high
bit 1-0	<b>FLTMOD&lt;1:0&gt;:</b> Fault Mode bits for PWM Generator # 11 = Fault input is disabled 10 = Reserved 01 = The selected Fault source forces PWMxH, PWMxL pins to FLTDAT values (cycle) 00 = The selected Fault source forces PWMxH, PWMxL pins to FLTDAT values (latched condition)

- Note 1:** These bits should be changed only when PTEN = 0 (PTCON<15>).
- 2:** When Independent Fault mode is enabled (IFLTMOD = 1), and Fault 1 is used for Current-Limit mode (CLSRC<4:0> = b0000), the Fault Control Source Select bits (FLTSRC<4:0>) should be set to an unused Fault source to prevent Fault 1 from disabling both the PWMxL and PWMxH outputs.
- 3:** When Independent Fault mode is enabled (IFLTMOD = 1) and Fault 1 is used for Fault mode (FLTSRC<4:0> = b0000), the Current-Limit Control Source Select bits (CLSRC<4:0>) should be set to an unused current-limit source to prevent the current-limit source from disabling both the PWMxH and PWMxL outputs.



## REGISTER 16-22: STRIGx: PWM SECONDARY TRIGGER COMPARE VALUE 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
STRGCMP<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
STRGCMP<7:3>					—	—	—
bit 7							bit 0

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 15-3      **STRGCMP<15:3>**: Secondary Trigger Compare Value bits

When the secondary PWM functions in local time base, this register contains the compare values that can trigger the ADC module.

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

**Note 1:** STRIGx cannot generate the PWM trigger interrupts.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 16-23: LEBCONx: LEADING-EDGE BLANKING CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0
PHR	PHF	PLR	PLF	FLTLEBEN	CLLEBEN	—	—
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
—	—	BCH <sup>(1)</sup>	BCL <sup>(1)</sup>	BPHH	BPHL	BPLH	BPLL
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      **PHR:** PWMxH Rising Edge Trigger Enable bit  
 1 = Rising edge of PWMxH will trigger Leading-Edge Blanking counter  
 0 = Leading-Edge Blanking ignores rising edge of PWMxH
- bit 14      **PHF:** PWMxH Falling Edge Trigger Enable bit  
 1 = Falling edge of PWMxH will trigger Leading-Edge Blanking counter  
 0 = Leading-Edge Blanking ignores falling edge of PWMxH
- bit 13      **PLR:** PWMxL Rising Edge Trigger Enable bit  
 1 = Rising edge of PWMxL will trigger Leading-Edge Blanking counter  
 0 = Leading-Edge Blanking ignores rising edge of PWMxL
- bit 12      **PLF:** PWMxL Falling Edge Trigger Enable bit  
 1 = Falling edge of PWMxL will trigger Leading-Edge Blanking counter  
 0 = Leading-Edge Blanking ignores falling edge of PWMxL
- bit 11      **FLTLEBEN:** Fault Input Leading-Edge Blanking Enable bit  
 1 = Leading-Edge Blanking is applied to selected fault input  
 0 = Leading-Edge Blanking is not applied to selected fault input
- bit 10      **CLLEBEN:** Current-Limit Leading-Edge Blanking Enable bit  
 1 = Leading-Edge Blanking is applied to selected current-limit input  
 0 = Leading-Edge Blanking is not applied to selected current-limit input
- bit 9-6      **Unimplemented:** Read as '0'
- bit 5      **BCH:** Blanking in Selected-Blanking Signal High Enable bit<sup>(1)</sup>  
 1 = State blanking (of current-limit and/or fault input signals) when selected blanking signal is high  
 0 = No blanking when selected blanking signal is high
- bit 4      **BCL:** Blanking in Selected-Blanking Signal Low Enable bit<sup>(1)</sup>  
 1 = State blanking (of current-limit and/or fault input signals) when selected blanking signal is low  
 0 = No blanking when selected blanking signal is low
- bit 3      **BPHH:** Blanking in PWMxH High Enable bit  
 1 = State blanking (of current-limit and/or fault input signals) when PWMxH output is high  
 0 = No blanking when PWMxH output is high
- bit 2      **BPHL:** Blanking in PWMxH Low Enable bit  
 1 = State blanking (of current-limit and/or fault input signals) when PWMxH output is low  
 0 = No blanking when PWMxH output is low
- bit 1      **BPLH:** Blanking in PWMxL High Enable bit  
 1 = State blanking (of current-limit and/or fault input signals) when PWMxL output is high  
 0 = No blanking when PWMxL output is high
- bit 0      **BPLL:** Blanking in PWMxL Low Enable bit  
 1 = State blanking (of current-limit and/or fault input signals) when PWMxL output is low  
 0 = No blanking when PWMxL output is low

**Note 1:** The blanking signal is selected via the BLANKSEL bits in the AUXCONx register.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 16-24: LEBDLYx: LEADING-EDGE BLANKING DELAY REGISTER

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	LEB<8:5>			
bit 15				bit 8			

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
LEB<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 15-12     **Unimplemented:** Read as '0'

bit 11-3     **LEB<8:0>:** Leading-Edge Blanking Delay bits for Current-Limit and Fault Inputs  
Value in 8.32 ns increments

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

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 16-25: AUXCONx: PWM AUXILIARY CONTROL REGISTER

R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
HRPDIS	HRDDIS	—	—	BLANKSEL<3:0>			
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
—	—	CHOPSEL<3:0>				CHOPHEN	CHOPLEN
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      **HRPDIS:** High Resolution PWM Period Disable bit  
 1 = High resolution PWM period is disabled to reduce power consumption  
 0 = High resolution PWM period is enabled
- bit 14      **HRDDIS:** High Resolution PWM Duty Cycle Disable bit  
 1 = High resolution PWM duty cycle is disabled to reduce power consumption  
 0 = High resolution PWM duty cycle is enabled
- bit 13-12   **Unimplemented:** Read as '0'
- bit 11-8    **BLANKSEL<3:0>:** PWM State Blank Source Select bits  
 The selected state blank signal will block the current limit and/or fault input signals (if enabled via the BCH and BCL bits in the LEBCONx register)  
 1001 = PWM9H selected as state blank source  
 1000 = PWM8H selected as state blank source  
 0111 = PWM7H selected as state blank source  
 0110 = PWM6H selected as state blank source  
 0101 = PWM5H selected as state blank source  
 0100 = PWM4H selected as state blank source  
 0011 = PWM3H selected as state blank source  
 0010 = PWM2H selected as state blank source  
 0001 = PWM1H selected as state blank source  
 0000 = 1'b0 (no state blanking)
- bit 7-6     **Unimplemented:** Read as '0'
- bit 5-2     **CHOPSEL<3:0>:** PWM Chop Clock Source Select bits  
 The selected signal will enable and disable (CHOP) the selected PWM outputs  
 1001 = PWM9H selected as CHOP clock source  
 1000 = PWM8H selected as CHOP clock source  
 0111 = PWM7H selected as CHOP clock source  
 0110 = PWM6H selected as CHOP clock source  
 0101 = PWM5H selected as CHOP clock source  
 0100 = PWM4H selected as CHOP clock source  
 0011 = PWM3H selected as CHOP clock source  
 0010 = PWM2H selected as CHOP clock source  
 0001 = PWM1H selected as CHOP clock source  
 0000 = Chop Clock generator selected as CHOP clock source
- bit 1        **CHOPHEN:** PWMxH Output Chopping Enable bit  
 1 = PWMxH chopping function is enabled  
 0 = PWMxH chopping function is disabled
- bit 0        **CHOPLEN:** PWMxL Output Chopping Enable bit  
 1 = PWMxL chopping function is enabled  
 0 = PWMxL chopping function is disabled

**REGISTER 16-26: PWMCAPx: PRIMARY PWM TIME BASE CAPTURE REGISTER**

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
PWMCAP<15:8> <sup>(1,2,3,4)</sup>							
bit 15							bit 8

R-0	R-0	R-0	R-0	R-0	U-0	U-0	U-0
PWMCAP<7:3> <sup>(1,2,3,4)</sup>					—	—	—
bit 7							bit 0

<b>Legend:</b>			
R = Readable bit	W = Writable bit	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      **PWMCAP<15:3>**: Captured PWM Time Base Value bits<sup>(1,2,3,4)</sup>  
 The value in this register represents the captured PWM time base value when a leading edge is detected on the current-limit input.

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

- Note 1:** The capture feature is only available on primary output (PWMxH).  
**2:** This feature is active only after LEB processing on the current-limit input signal is complete.  
**3:** The minimum capture resolution is 8.32 ns.  
**4:** This feature can be used when the XPRES bit (PWMCONx<1>) is set to '0'.

NOTES:

## 17.0 QUADRATURE ENCODER INTERFACE (QEI) MODULE

**Note 1:** This data sheet summarizes the features of the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 families of devices. 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](http://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 chapter 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.

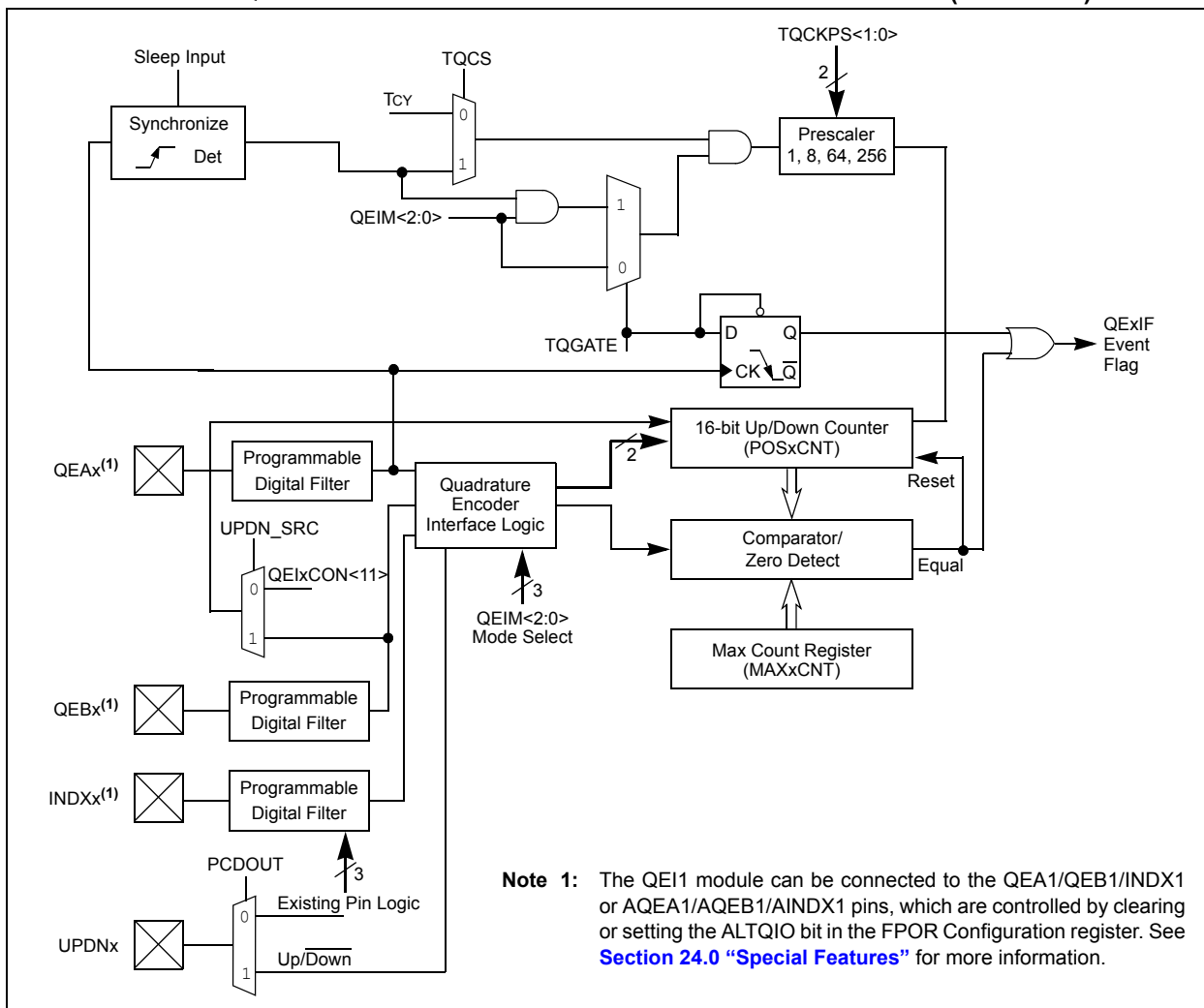
The operational features of the QEI include:

- Three input channels for two phase signals and 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

These operating modes are determined by setting the appropriate bits, QEIM<2:0> in (QEIxCON<10:8>). **Figure 17-1** depicts the Quadrature Encoder Interface block diagram.

**Note:** An ‘x’ used in the names of pins, control/status bits and registers denotes a particular QEI module number (x = 1 or 2).

**FIGURE 17-1: QUADRATURE ENCODER INTERFACE BLOCK DIAGRAM (x = 1 OR 2)**



**Note 1:** The QEI1 module can be connected to the QEA1/QEB1/INDX1 or AQEA1/AQEB1/AINDX1 pins, which are controlled by clearing or setting the ALTQIO bit in the FPOR Configuration register. See **Section 24.0 “Special Features”** for more information.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 17-1: QEIxCON: QEIx CONTROL REGISTER (x = 1 or 2)

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	TQCKPS<1:0>		POSRES	TQCS	UPDN_SRC
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<sup>(1)</sup>  
 1 = Position count error has occurred  
 0 = No position count error has occurred
- 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<sup>(2)</sup>  
 1 = Position Counter Direction is positive (+)  
 0 = Position Counter Direction is negative (-)
- 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 (MAXxCNT)  
 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 (MAXxCNT)  
 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)

- Note 1:** CNTERR flag only applies when QEIM<2:0> = '110' or '100'.  
**Note 2:** Read-only bit when QEIM<2:0> = '1xx'. Read/write bit when QEIM<2:0> = '001'.  
**Note 3:** Prescaler utilized for 16-bit Timer mode only.  
**Note 4:** This bit applies only when QEIM<2:0> = 100 or 110.  
**Note 5:** When configured for QEI mode, this control bit is a 'don't care'.



## REGISTER 17-1: QEIxCON: QEIx CONTROL REGISTER (x = 1 or 2) (CONTINUED)

bit 5	<b>TQGATE:</b> Timer Gated Time Accumulation Enable bit 1 = Timer gated time accumulation enabled 0 = Timer gated time accumulation disabled
bit 4-3	<b>TQCKPS&lt;1:0&gt;:</b> Timer Input Clock Prescale Select bits <sup>(3)</sup> 11 = 1:256 prescale value 10 = 1:64 prescale value 01 = 1:8 prescale value 00 = 1:1 prescale value
bit 2	<b>POSRES:</b> Position Counter Reset Enable bit <sup>(4)</sup> 1 = Index Pulse resets Position Counter 0 = Index Pulse does not reset Position Counter
bit 1	<b>TQCS:</b> Timer Clock Source Select bit 1 = External clock from pin QEAx (on the rising edge) 0 = Internal clock (TCY)
bit 0	<b>UPDN_SRC:</b> Position Counter Direction Selection Control bit <sup>(5)</sup> 1 = QEBx pin state defines position counter direction 0 = Control/Status bit, UPDN (QEIxCON<11>), defines timer counter (POSxCNT) direction

- Note 1:** CNTERR flag only applies when QEIM<2:0> = '110' or '100'.
- 2:** Read-only bit when QEIM<2:0> = '1xx'. Read/write bit when QEIM<2:0> = '001'.
- 3:** Prescaler utilized for 16-bit Timer mode only.
- 4:** This bit applies only when QEIM<2:0> = 100 or 110.
- 5:** When configured for QEI mode, this control bit is a 'don't care'.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 17-2: DFLT<sub>x</sub>CON: DIGITAL FILTER 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 application to specify the state of the QEAx and QEBx input pins during an Index pulse when the POSxCNT register is to be reset.  
In x4 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 x4 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'

**18.0 SERIAL PERIPHERAL INTERFACE (SPI)**

**Note 1:** This data sheet summarizes the features of the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 families of devices. 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](http://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 can be serial EEPROMs, shift registers, display drivers, analog-to-digital converters and so on. The SPI module is compatible with SPI and SIOP from Motorola®.

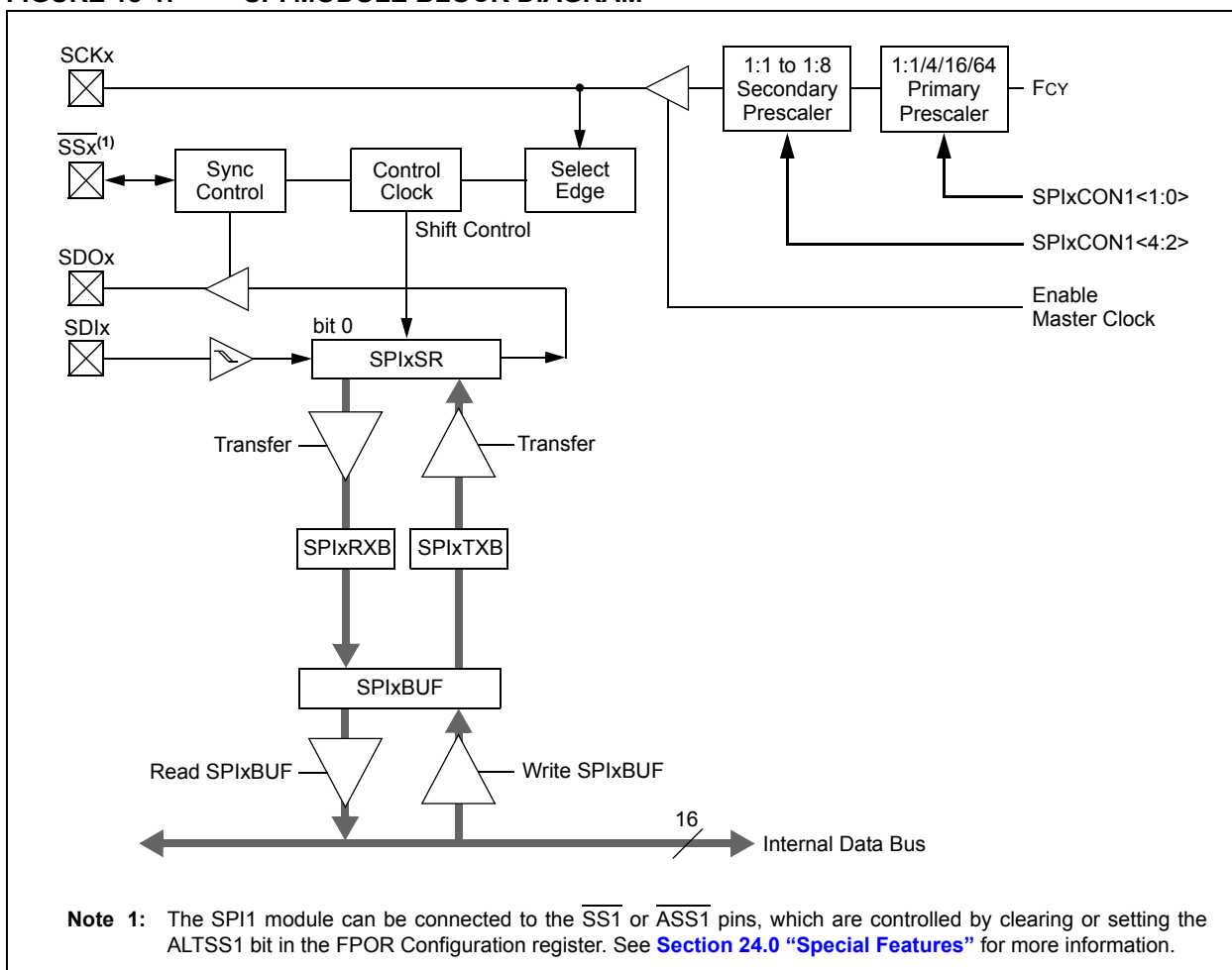
The SPI module consists of a 16-bit shift register, SPIxSR (where x = 1), used for shifting data in and out, and a buffer register, SPIxBUF. A control register, SPIxCON, configures the module. Additionally, a STATUS register, SPIxSTAT, indicates status conditions.

The serial interface consists of these four pins:

- SDIx (Serial Data Input)
- SDOx (Serial Data Output)
- SCKx (Shift Clock Input Or Output)
- SSx (Active-Low Slave Select).

In Master mode operation, SCK is a clock output; in Slave mode, it is a clock input.

**FIGURE 18-1: SPI MODULE BLOCK DIAGRAM**



# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## 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

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

- bit 15      **SPIEN:** SPIx Enable bit  
 1 = Enables module and configures SCKx, SDOx, SDIx and  $\overline{SSx}$  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> <sup>(2)</sup>			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)<sup>(3)</sup>  
 1 = 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. Program this bit to '0' for the Framed SPI modes (FRMEN = 1).
- 2:** Do not set both 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. Program this bit to '0' for the Framed SPI modes (FRMEN = 1).

**2:** Do not set both 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 clock  
 0 = Frame sync pulse precedes first bit clock
- bit 0        **Unimplemented:** This bit must not be set to '1' by the user application

NOTES:



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

**Note 1:** This data sheet summarizes the features of the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 families of devices. 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 (I<sup>2</sup>C™)”** (DS70195) in the “dsPIC33F/PIC24H Family Reference Manual”, which is available from the Microchip web site ([www.microchip.com](http://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 provides complete hardware support for both Slave and Multi-Master modes of the I<sup>2</sup>C serial communication standard with a 16-bit interface.

The I<sup>2</sup>C module has a 2-pin interface:

- The SCLx pin is clock.
- The SDAx pin is data.

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

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

## 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-bit 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, refer to the “dsPIC33F/PIC24H Family Reference Manual”. Please see the Microchip web site ([www.microchip.com](http://www.microchip.com)) for the latest “dsPIC33F/PIC24H Family Reference Manual” sections.

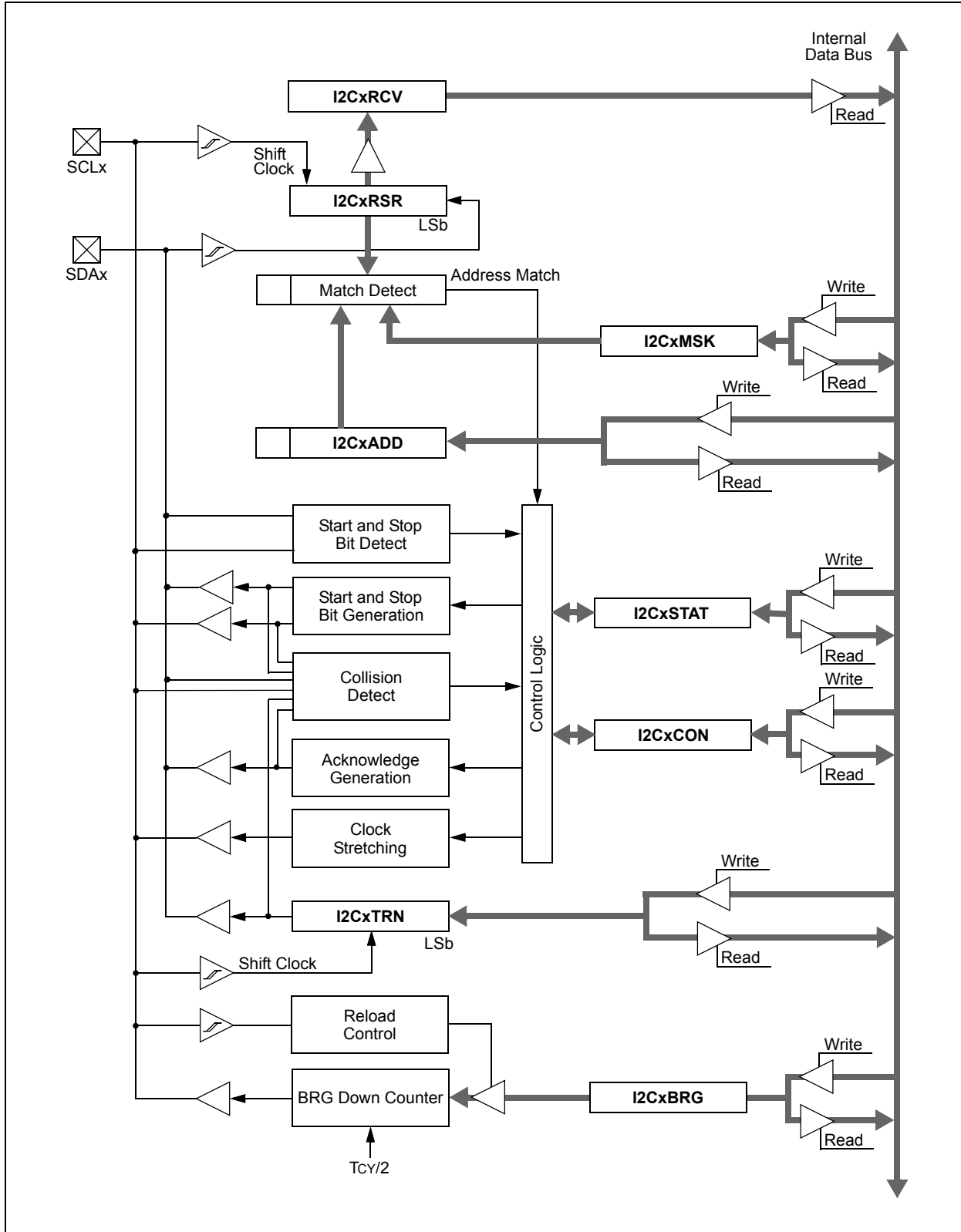
## 19.2 I<sup>2</sup>C Registers

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 I2CxSTAT are read/write:

- I2CxRSR is the shift register used for shifting data internal to the module and the user application has no access to it.
- I2CxRCV is the receive buffer and the register to which data bytes are written, or from which data bytes are read.
- 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.

FIGURE 19-1: I<sup>2</sup>C™ BLOCK DIAGRAM (x = 1)



## 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	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC
GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN
bit 7							bit 0

<b>Legend:</b>	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 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 can 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 can 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 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 is 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, HSC	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	P	S	R_W	RBF	TBF
bit 7						bit 0	

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

- bit 15     **ACKSTAT:** Acknowledge Status bit  
(when operating as I<sup>2</sup>C 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 device address  
Hardware clear at device address match. Hardware set by reception of slave byte.
- 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.

## REGISTER 19-2: I2CxSTAT: I2Cx STATUS REGISTER (CONTINUED)

- 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<sup>2</sup>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
—	—	—	—	—	—	AMSK<9:8>	
bit 15						bit 8	

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
AMSK<7:0>							
bit 7						bit 0	

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 15-10

**Unimplemented:** Read as '0'

bit 9-0

**AMSK<9:0>:** 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

NOTES:



## 20.0 UNIVERSAL ASYNCHRONOUS RECEIVER TRANSMITTER (UART)

**Note 1:** This data sheet summarizes the features of the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 families of devices. 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](http://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 dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 device families. The UART is a full-duplex, asynchronous system that can communicate with peripheral devices, such as personal computers, LIN, 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.

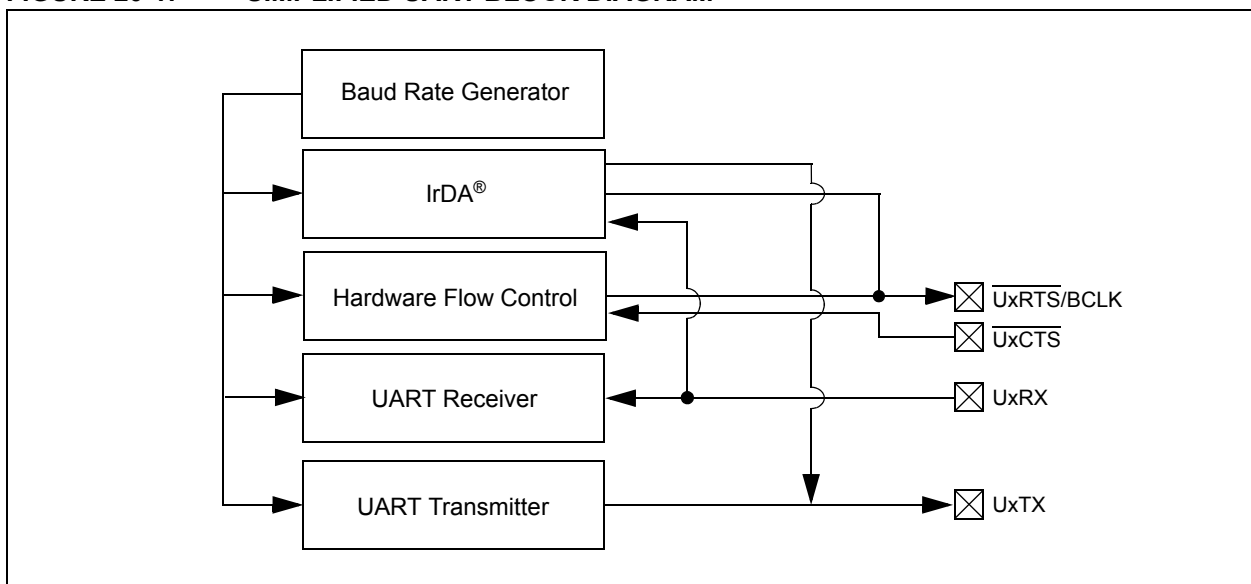
The primary features of the UART module are:

- 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
- Baud Rates Ranging from 12.5 Mbps to 47 bps at 50 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
- Support for Automatic Baud Rate Detection
- IrDA Encoder and Decoder Logic
- 16x Baud Clock Output for IrDA Support
- Support for DMA

A simplified block diagram of the UART module is shown in **Figure 20-1**. The UART module consists of these key hardware elements:

- Baud Rate Generator
- Asynchronous Transmitter
- Asynchronous Receiver

**FIGURE 20-1: SIMPLIFIED UART BLOCK DIAGRAM**



# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## 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	

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

- bit 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 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<sup>®</sup> Encoder and Decoder Enable bit<sup>(2)</sup>  
 1 = IrDA encoder and decoder enabled  
 0 = IrDA encoder and decoder disabled
- bit 11      **RTSMD:** Mode Selection for  $\overline{\text{UxRTS}}$  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 Pin Enable bits  
 11 = UxTX, UxRX and BCLK pins are enabled and used;  $\overline{\text{UxCTS}}$  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; UxCTS 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 falling edge; bit cleared in hardware on 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 (55h) 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. That section of the manual is available on the Microchip web site, [www.microchip.com](http://www.microchip.com).

**2:** This feature is only available for the 16x BRG mode (BRGH = 0).

## REGISTER 20-1: UxMODE: UARTx MODE REGISTER (CONTINUED)

bit 4	<b>URXINV:</b> Receive Polarity Inversion bit 1 = UxRX Idle state is '0' 0 = UxRX Idle state is '1'
bit 3	<b>BRGH:</b> 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	<b>PDSEL&lt;1:0&gt;:</b> 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	<b>STSEL:</b> Stop Bit Selection bit 1 = Two Stop bits 0 = 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. That section of the manual is available on the Microchip web site, [www.microchip.com](http://www.microchip.com).

**2:** This feature is only available for the 16x BRG mode (BRGH = 0).

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## 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
URXISEL<1:0>	ADDEN	RIDLE	PERR	FERR	OERR	URXDA	
bit 7						bit 0	

<b>Legend:</b>	HC = Hardware Clearable 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 15,13 **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, 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<sup>®</sup> 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 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 transmit buffer is empty (the last transmission has completed)
  - 0 = Transmit Shift register is not empty, a transmission is in progress or queued
- bit 7-6 **URXISEL<1:0>**: Receive Interrupt Mode Selection bits
- 11 = Interrupt is set on UxRSR transfer making the receive buffer full (i.e., has 4 data characters)
  - 10 = Interrupt is set on 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

**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. That section of the manual is available on the Microchip web site, [www.microchip.com](http://www.microchip.com).

## REGISTER 20-2: UxSTA: UARTx STATUS AND CONTROL REGISTER (CONTINUED)

- 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 (clear/read-only)  
1 = Receive buffer has overflowed  
0 = Receive buffer has not overflowed. Clearing a previously set OERR bit (1 →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. That section of the manual is available on the Microchip web site, [www.microchip.com](http://www.microchip.com).

NOTES:

## 21.0 ENHANCED CAN (ECAN™) MODULE

**Note 1:** This data sheet summarizes the features of the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 families of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 21. “Enhanced Controller Area Network (ECAN™)”** (DS70185) in the *dsPIC33F/PIC24H Family Reference Manual*, which is available from the Microchip web site ([www.microchip.com](http://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) 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 dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 devices contain up to two ECAN modules.

The ECAN module is a communication controller implementing the CAN 2.0 A/B protocol, as defined in the BOSCH CAN specification. The module supports 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 can 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 can contain up to 8 bytes of data)
- Up to 32 receive buffers (each buffer can contain up to 8 bytes of data)
- Up to 16 full (standard/extended identifier) acceptance filters
- Three full acceptance filter masks
- DeviceNet™ 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 CAN1) for time-stamping and network synchronization
- Low-power Sleep and Idle mode

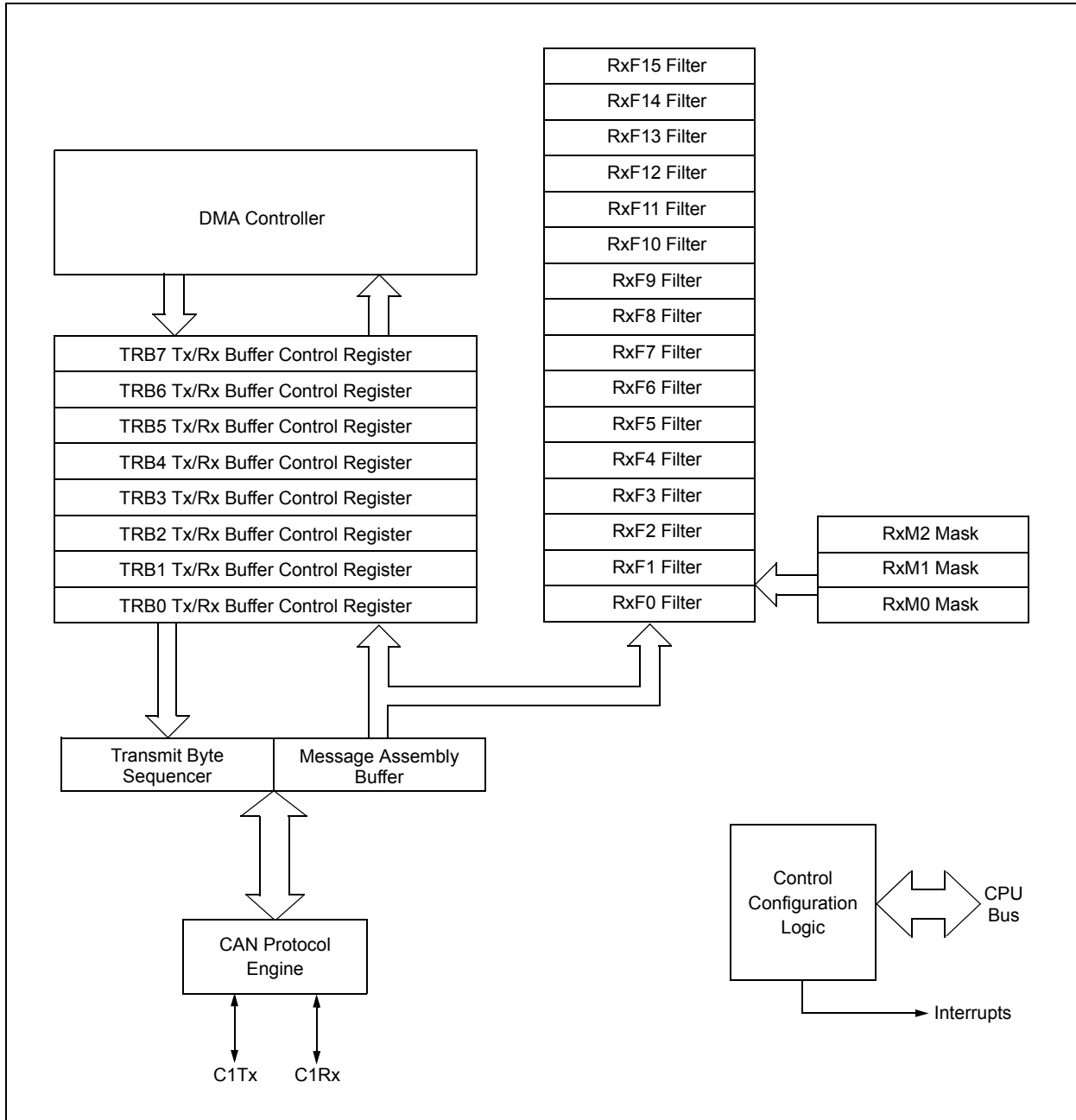
The CAN bus module consists of a protocol engine and message buffering/control. The CAN protocol engine handles all functions for receiving and transmitting messages on the CAN bus. Messages are transmitted by first loading the appropriate data registers. Status and errors can be checked by reading the 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 ECAN 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 sends 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 inter-frame 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 can 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.

FIGURE 21-1: ECAN™ MODULE BLOCK DIAGRAM





## 21.3 Modes of Operation

The ECAN 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 does 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 does not transmit or receive. The error counters are cleared and the interrupt flags remain unchanged. The user application has access to Configuration registers that are access restricted in other modes. The module protects 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 ECAN module is not 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 does not transmit or receive. The module has the ability to set the WAKIF bit due to bus activity, however, any pending interrupts remains and the error counters retains their value.

If the REQOP<2:0> bits (CiCTRL1<10:8>) = 001, the module enters the Module Disable mode. If the module is active, the module waits 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 reverts 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 ECAN module is allowed to transmit in a particular mode of operation and a transmission is requested immediately after the ECAN 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 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 assumes the CAN bus functions. The module transmits 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 connects 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.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## 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			

<b>Legend:</b>	C = Writable bit, but only '0' can be written to clear the bit	r = Bit is Reserved
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared
		x = Bit is unknown

- bit 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 not use
- bit 10-8     **REQOP<2:0>:** Request Operation Mode bits
  - 111 = Set Listen All Messages mode
  - 110 = Reserved
  - 101 = Reserved
  - 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	

<b>Legend:</b>	C = Writeable bit, but only '0' can be written to clear the 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 15-5      **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

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## 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

<b>Legend:</b>	C = Writeable bit, but only '0' can be written to clear the bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-13 **Unimplemented:** Read as '0'

bit 12-8 **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

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

<b>Legend:</b>	C = Writeable bit, but only '0' can be written to clear the bit
R = Readable bit	W = Writable bit      U = Unimplemented bit, read as '0'
-n = Value at POR	'1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown

bit 15-13      **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 = Read buffer RB31
- 11110 = Read buffer RB30
- 
- 
- 
- 00001 = Tx/Rx buffer TRB1
- 00000 = Tx/Rx buffer TRB0

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## 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	
—	—	FNRB<5:0>						
bit 7							bit 0	

<b>Legend:</b>	C = Writable bit, but only '0' can be written to clear the bit
R = Readable bit	W = Writable bit      U = Unimplemented bit, read as '0'
-n = Value at POR	'1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown

bit 15-14      **Unimplemented:** Read as '0'

bit 13-8      **FBP<5:0>:** FIFO 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

## REGISTER 21-6: CIINTF: ECAN™ INTERRUPT FLAG REGISTER

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

<b>Legend:</b>	C = Writeable bit, but only '0' can be written to clear the 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      **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
- bit 11      **RXBP:** Receiver in Error State Bus Passive bit  
                  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 State Warning state  
                  0 = Transmitter or Receiver is not in Error State Warning state
- bit 7      **IVRIF:** Invalid Message Received Interrupt Flag bit  
                  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

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## 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

<b>Legend:</b>	C = Writeable bit, but only '0' can be written to clear the bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

- bit 15-8      **Unimplemented:** Read as '0'
- bit 7        **IVRIE:** Invalid Message Received Interrupt Enable bit  
1 = Interrupt Request Enabled  
0 = Interrupt Request not enabled
- bit 6        **WAKIE:** Bus Wake-up Activity Interrupt Flag bit  
1 = Interrupt Request Enabled  
0 = Interrupt Request not enabled
- bit 5        **ERRIE:** Error Interrupt Enable bit  
1 = Interrupt Request Enabled  
0 = Interrupt Request not enabled
- bit 4        **Unimplemented:** Read as '0'
- bit 3        **FIFOIE:** FIFO Almost Full Interrupt Enable bit  
1 = Interrupt Request Enabled  
0 = Interrupt Request not enabled
- bit 2        **RBOVIE:** RX Buffer Overflow Interrupt Enable bit  
1 = Interrupt Request Enabled  
0 = Interrupt Request not enabled
- bit 1        **RBIE:** RX Buffer Interrupt Enable bit  
1 = Interrupt Request Enabled  
0 = Interrupt Request not enabled
- bit 0        **TBIE:** TX Buffer Interrupt Enable bit  
1 = Interrupt Request Enabled  
0 = Interrupt Request not enabled



# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## 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
RERRCNT<7:0>							
bit 7				bit 0			

<b>Legend:</b>	C = Writeable bit, but only '0' can be written to clear the bit
R = Readable bit	W = Writable bit      U = Unimplemented bit, read as '0'
-n = Value at POR	'1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown

bit 15-8      **TERRCNT<7:0>**: Transmit Error Count bits

bit 7-0      **RERRCNT<7:0>**: Receive Error Count bits

## REGISTER 21-9: C1CFG1: 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<5:0>					
bit 7				bit 0			

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

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 21-10: C1CFG2: 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 Segment 2 bits  
                                     111 = Length is 8 x T<sub>Q</sub>  
                                     •  
                                     •  
                                     •  
                                     000 = Length is 1 x T<sub>Q</sub>
- 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 point  
                                     0 = Bus line is sampled once at the sample point
- bit 5-3                      **SEG1PH<2:0>:** Phase Segment 1 bits  
                                     111 = Length is 8 x T<sub>Q</sub>  
                                     •  
                                     •  
                                     •  
                                     000 = Length is 1 x T<sub>Q</sub>
- bit 2-0                      **PRSEG<2:0>:** Propagation Time Segment bits  
                                     111 = Length is 8 x T<sub>Q</sub>  
                                     •  
                                     •  
                                     •  
                                     000 = Length is 1 x T<sub>Q</sub>

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## 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	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
FLTEN7	FLTEN6	FLTEN5	FLTEN4	FLTEN3	FLTEN2	FLTEN1	FLTEN0
bit 7							bit 0

<b>Legend:</b>	C = Writeable bit, but only '0' can be written to clear the 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      **FLTENn**: Enable Filter n to Accept Messages bits  
 1 = Enable Filter n  
 0 = Disable Filter n

## REGISTER 21-12: CiBUFNT1: 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

<b>Legend:</b>	C = Writeable bit, but only '0' can be written to clear the 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 15-12      **F3BP<3:0>**: RX Buffer Mask for Filter 3 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 Mask for Filter 2 bits (same values as bit 15-12)

bit 7-4      **F1BP<3:0>**: RX Buffer Mask for Filter 1 bits (same values as bit 15-12)

bit 3-0      **F0BP<3:0>**: RX Buffer Mask for Filter 0 bits (same values as bit 15-12)

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## 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			

<b>Legend:</b>	C = Writeable bit, but only '0' can be written to clear the 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 15-12    **F7BP<3:0>**: RX Buffer Mask for Filter 7 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 Mask for Filter 6 bits (same values as bit 15-12)  
 bit 7-4    **F5BP<3:0>**: RX Buffer Mask for Filter 5 bits (same values as bit 15-12)  
 bit 3-0    **F4BP<3:0>**: RX Buffer Mask for Filter 4 bits (same values as bit 15-12)

## 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			

<b>Legend:</b>	C = Writeable bit, but only '0' can be written to clear the 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 15-12    **F11BP<3:0>**: RX Buffer Mask for Filter 11 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 Mask for Filter 10 bits (same values as bit 15-12)  
 bit 7-4    **F9BP<3:0>**: RX Buffer Mask for Filter 9 bits (same values as bit 15-12)  
 bit 3-0    **F8BP<3:0>**: RX Buffer Mask for Filter 8 bits (same values as bit 15-12)

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## 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			

<b>Legend:</b>	C = Writeable bit, but only '0' can be written to clear the 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 15-12      **F15BP<3:0>**: RX Buffer Mask for Filter 15 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 Mask for Filter 14 bits (same values as bit 15-12)

bit 7-4      **F13BP<3:0>**: RX Buffer Mask for Filter 13 bits (same values as bit 15-12)

bit 3-0      **F12BP<3:0>**: RX Buffer Mask for Filter 12 bits (same values as bit 15-12)

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 21-16: C<sub>IRXF<sub>n</sub></sub>SID: ECAN™ ACCEPTANCE FILTER STANDARD IDENTIFIER REGISTER n (n = 0-15)

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3
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
SID2	SID1	SID0	—	EXIDE	—	EID17	EID16
bit 7							bit 0

<b>Legend:</b>	C = Writeable bit, but only '0' can be written to clear the 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 15-5      **SID<10:0>**: Standard Identifier bits
  - 1 = Message address bit SID<sub>x</sub> must be '1' to match filter
  - 0 = Message address bit SID<sub>x</sub> 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 addresses
  - 0 = 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 EID<sub>x</sub> must be '1' to match filter
  - 0 = Message address bit EID<sub>x</sub> must be '0' to match filter

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 21-17: CiRXFnEID: ECAN™ ACCEPTANCE FILTER EXTENDED IDENTIFIER REGISTER n (n = 0-15)

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8
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
EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0
bit 7						bit 0	

<b>Legend:</b>	C = Writeable bit, but only '0' can be written to clear the 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      **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>		F6MSK<1:0>		F5MSK<1:0>		F4MSK<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>		F2MSK<1:0>		F1MSK<1:0>		F0MSK<1:0>	
bit 7						bit 0	

<b>Legend:</b>	C = Writeable bit, but only '0' can be written to clear the 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      **F7MSK<1:0>**: Mask Source for Filter 7 bits  
 11 = Reserved  
 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 bits (same values as bit 15-14)

bit 11-10      **F5MSK<1:0>**: Mask Source for Filter 5 bits (same values as bit 15-14)

bit 9-8        **F4MSK<1:0>**: Mask Source for Filter 4 bits (same values as bit 15-14)

bit 7-6        **F3MSK<1:0>**: Mask Source for Filter 3 bits (same values as bit 15-14)

bit 5-4        **F2MSK<1:0>**: Mask Source for Filter 2 bits (same values as bit 15-14)

bit 3-2        **F1MSK<1:0>**: Mask Source for Filter 1 bits (same values as bit 15-14)

bit 1-0        **F0MSK<1:0>**: Mask Source for Filter 0 bits (same values as bit 15-14)

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 21-19: C1FMSKSEL2: 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
F15MSK<1:0>		F14MSK<1:0>		F13MSK<1:0>		F12MSK<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
F11MSK<1:0>		F10MSK<1:0>		F9MSK<1:0>		F8MSK<1:0>	
bit 7							bit 0

<b>Legend:</b>	C = Writeable bit, but only '0' can be written to clear the 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    **F15MSK<1:0>**: Mask Source for Filter 15 bits  
                   11 = Reserved  
                   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 bits (same values as bit 15-14)
- bit 11-10    **F13MSK<1:0>**: Mask Source for Filter 13 bits (same values as bit 15-14)
- bit 9-8        **F12MSK<1:0>**: Mask Source for Filter 12 bits (same values as bit 15-14)
- bit 7-6        **F11MSK<1:0>**: Mask Source for Filter 11 bits (same values as bit 15-14)
- bit 5-4        **F10MSK<1:0>**: Mask Source for Filter 10 bits (same values as bit 15-14)
- bit 3-2        **F9MSK<1:0>**: Mask Source for Filter 9 bits (same values as bit 15-14)
- bit 1-0        **F8MSK<1:0>**: Mask Source for Filter 8 bits (same values as bit 15-14)



# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 21-20: CiRXMnSID: ECAN™ ACCEPTANCE FILTER MASK STANDARD IDENTIFIER REGISTER n (n = 0-2)

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3
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
SID2	SID1	SID0	—	MIDE	—	EID17	EID16
bit 7							bit 0

<b>Legend:</b>	C = Writeable bit, but only '0' can be written to clear the 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 15-5      **SID<10:0>**: Standard Identifier bits
  - 1 = Include bit SIDx in filter comparison
  - 0 = Bit SIDx is 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 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 don't care in filter comparison

## REGISTER 21-21: CiRXMnEID: ECAN™ ACCEPTANCE FILTER MASK EXTENDED IDENTIFIER REGISTER n (n = 0-2)

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8
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
EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0
bit 7							bit 0

<b>Legend:</b>	C = Writeable bit, but only '0' can be written to clear the 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 15-0      **EID<15:0>**: Extended Identifier bits
  - 1 = Include bit EIDx in filter comparison
  - 0 = Bit EIDx is don't care in filter comparison

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 21-22: C<sub>i</sub>RXFUL1: 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	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXFUL7	RXFUL6	RXFUL5	RXFUL4	RXFUL3	RXFUL2	RXFUL1	RXFUL0
bit 7							bit 0

<b>Legend:</b>	C = Writeable bit, but only '0' can be written to clear the 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      **RXFUL<15:0>**: Receive Buffer n Full bits  
                   1 = Buffer is full (set by module)  
                   0 = Buffer is empty

## REGISTER 21-23: C<sub>i</sub>RXFUL2: ECAN™ RECEIVE BUFFER FULL REGISTER 2

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXFUL31	RXFUL30	RXFUL29	RXFUL28	RXFUL27	RXFUL26	RXFUL25	RXFUL24
bit 15							bit 8

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXFUL23	RXFUL22	RXFUL21	RXFUL20	RXFUL19	RXFUL18	RXFUL17	RXFUL16
bit 7							bit 0

<b>Legend:</b>	C = Writeable bit, but only '0' can be written to clear the 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      **RXFUL<31:16>**: Receive Buffer n Full bits  
                   1 = Buffer is full (set by module)  
                   0 = Buffer is empty

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 21-24: C<sub>I</sub>RXOVF1: 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	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXOVF7	RXOVF6	RXOVF5	RXOVF4	RXOVF3	RXOVF2	RXOVF1	RXOVF0
bit 7							bit 0

<b>Legend:</b>	C = Writeable bit, but only '0' can be written to clear the 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      **RXOVF<15:0>**: Receive Buffer n Overflow bits  
 1 = Module attempted to write to a full buffer (set by module)  
 0 = No overflow condition

## REGISTER 21-25: C<sub>I</sub>RXOVF2: ECAN™ RECEIVE BUFFER OVERFLOW REGISTER 2

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXOVF31	RXOVF30	RXOVF29	RXOVF28	RXOVF27	RXOVF26	RXOVF25	RXOVF24
bit 15							bit 8

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXOVF23	RXOVF22	RXOVF21	RXOVF20	RXOVF19	RXOVF18	RXOVF17	RXOVF16
bit 7							bit 0

<b>Legend:</b>	C = Writeable bit, but only '0' can be written to clear the 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      **RXOVF<31:16>**: Receive Buffer n Overflow bits  
 1 = Module attempted to write to a full buffer (set by module)  
 0 = No overflow condition

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 21-26: CiTRmnCON: ECAN™ Tx/Rx BUFFER m 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	TXmPRI<1:0>	
bit 7						bit 0	

<b>Legend:</b>	C = Writeable bit, but only '0' can be written to clear the bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

- bit 15-8 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  
1 = Requests that a message be sent. The bit automatically clears when the message is successfully sent.  
0 = Clearing the bit to '0' while set requests 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.

**21.4 ECAN Message Buffers**

ECAN Message Buffers are part of DMA RAM Memory. They are not ECAN Special Function Registers. The user application must directly write into the DMA RAM area that is configured for ECAN Message Buffers. The location and size of the buffer area is defined by the user application.

**BUFFER 21-1: ECAN™ MESSAGE BUFFER WORD 0**

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	SID10	SID9	SID8	SID7	SID6
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
SID5	SID4	SID3	SID2	SID1	SID0	SRR	IDE
bit 7							bit 0

<b>Legend:</b>			
R = Readable bit	W = Writable bit	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 identifier
  - 0 = Message will transmit standard identifier

**BUFFER 21-2: ECAN™ MESSAGE BUFFER WORD 1**

U-0	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	—	EID17	EID16	EID15	EID14
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
EID13	EID12	EID11	EID10	EID9	EID8	EID7	EID6
bit 7							bit 0

<b>Legend:</b>			
R = Readable bit	W = Writable bit	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

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## BUFFER 21-3: ECAN™ MESSAGE BUFFER WORD 2

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-x	U-x	U-x	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

## BUFFER 21-4: ECAN™ MESSAGE BUFFER WORD 3

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
Byte 1							
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
Byte 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      **Byte 1<15:8>**: ECAN™ Message Byte 0
- bit 7-0       **Byte 0<7:0>**: ECAN Message Byte 1

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## BUFFER 21-5: ECAN™ MESSAGE BUFFER WORD 4

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
Byte 3							
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
Byte 2							
bit 7				bit 0			

### Legend:

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

bit 15-8                      **Byte 3<15:8>**: ECAN™ Message Byte 3

bit 7-0                      **Byte 2<7:0>**: ECAN Message Byte 2

## BUFFER 21-6: ECAN™ MESSAGE BUFFER WORD 5

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
Byte 5							
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
Byte 4							
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                      **Byte 5<15:8>**: ECAN™ Message Byte 5

bit 7-0                      **Byte 4<7:0>**: ECAN Message Byte 4

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## BUFFER 21-7: ECAN™ MESSAGE BUFFER WORD 6

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
Byte 7							
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
Byte 6							
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                      **Byte 7<15:8>**: ECAN™ Message Byte 7

bit 7-0                      **Byte 6<7:0>**: ECAN Message Byte 6

## BUFFER 21-8: ECAN™ MESSAGE BUFFER WORD 7

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	FILHIT<4:0> <sup>(1)</sup>				
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<sup>(1)</sup>  
 Encodes number of filter that resulted in writing this buffer.

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

**Note 1:** Only written by module for receive buffers, unused for transmit buffers.



## 22.0 HIGH-SPEED 10-BIT ANALOG-TO-DIGITAL CONVERTER (ADC)

**Note 1:** This data sheet summarizes the features of the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 families of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 44. “High-Speed 10-Bit Analog-to-Digital Converter (ADC)”** (DS70321) in the “dsPIC33F/PIC24H Family Reference Manual”, which is available from the Microchip web site ([www.microchip.com](http://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 dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 devices provide high-speed successive approximation Analog-to-Digital conversions to support applications such as AC/DC and DC/DC power converters.

### 22.1 Features Overview

The ADC module incorporates the following features:

- 10-bit resolution
- Unipolar inputs
- Up to two Successive Approximation Registers (SARs)
- Up to 24 external input channels
- Two internal analog inputs
- Dedicated result register for each analog input
- $\pm 1$  LSB accuracy at 3.3V
- Single supply operation
- 4 Msps conversion rate at 3.3V (devices with two SARs)
- 2 Msps conversion rate at 3.3V (devices with one SAR)
- Low-power CMOS technology

### 22.2 Module Description

This ADC module is designed for applications that require low latency between the request for conversion and the resultant output data. Typical applications include:

- AC/DC power supplies
- DC/DC converters
- Power Factor Correction (PFC)

This ADC works with the high-speed PWM module in power control applications that require high-frequency control loops. This module can sample and convert two analog inputs in a 0.5 microsecond when two SARs are used. This small conversion delay reduces the “phase lag” between measurement and control system response.

Up to five inputs may be sampled at a time (four inputs from the dedicated sample and hold circuits and one from the shared sample and hold circuit). If multiple inputs request conversion, the ADC will convert them in a sequential manner, starting with the lowest order input.

This ADC design provides each pair of analog inputs (AN1,AN0), (AN3,AN2),..., the ability to specify its own trigger source out of a maximum of sixteen different trigger sources. This capability allows this ADC to sample and convert analog inputs that are associated with PWM generators operating on independent time bases.

The user application typically requires synchronization between analog data sampling and PWM output to the application circuit. The very high-speed operation of this ADC module allows “data on demand”.

In addition, several hardware features have been added to the peripheral interface to improve real-time performance in a typical DSP-based application.

- Result alignment options
- Automated sampling
- External conversion start control
- Two internal inputs to monitor the INTREF and EXTREF input signals

A block diagram of the ADC module is shown in [Figure 22-2](#).

## 22.3 Module Functionality

The high-speed 10-bit ADC is designed to support power conversion applications when used with the High-Speed PWM module. The ADC may have one or two SAR modules, depending on the device variant. If two SARs are present on a device, two conversions can be processed at a time, yielding 4 Msps conversion rate. If only one SAR is present on a device, only one conversion can be processed at a time, yielding 2 Msps conversion rate. The high-speed 10-bit ADC produces two 10-bit conversion results in a 0.5 microsecond.

The ADC module supports up to 24 external analog inputs and two internal analog inputs. To monitor reference voltage, two internal inputs, AN24 and AN25, are connected to EXTREF and INTREF, respectively.

The analog reference voltage is defined as the device supply voltage (AVDD/AVSS).

The ADC module uses the following control and STATUS registers:

- [ADC CON: ADC Control Register](#)
- [ADSTAT: ADC Status Register](#)
- [ADBASE: ADC Base Register<sup>\(1,2\)</sup>](#)
- [ADPCFG: ADC Port Configuration Register](#)
- [ADPCFG2: ADC Port Configuration Register 2](#)
- [ADCPC0: Analog-to-Digital Convert Pair Control Register 0](#)
- [ADCPC1: Analog-to-Digital Convert Pair Control Register 1](#)
- [ADCPC2: Analog-to-Digital Convert Pair Control Register 2](#)
- [ADCPC3: Analog-to-Digital Convert Pair Control Register 3](#)
- [ADCPC4: Analog-to-Digital Convert Pair Control Register 4](#)
- [ADCPC5: Analog-to-Digital Convert Pair Control Register 5](#)
- [ADCPC6: Analog-to-Digital Convert Pair Control Register 6\(2\)](#)

The ADCON register controls the operation of the ADC module. The ADSTAT register displays the status of the conversion processes. The ADPCFG registers configure the port pins as analog inputs or as digital I/O. The ADCPCx registers control the triggering of the ADC conversions. See [Register 22-1](#) through [Register 22-12](#) for detailed bit configurations.

**Note:** A unique feature of the ADC module is its ability to sample inputs in an asynchronous manner. Individual sample and hold circuits can be triggered independently of each other.

FIGURE 22-1: ADC BLOCK DIAGRAM FOR dsPIC33FJ32GS406 AND dsPIC33FJ64GS406 DEVICES WITH ONE SAR

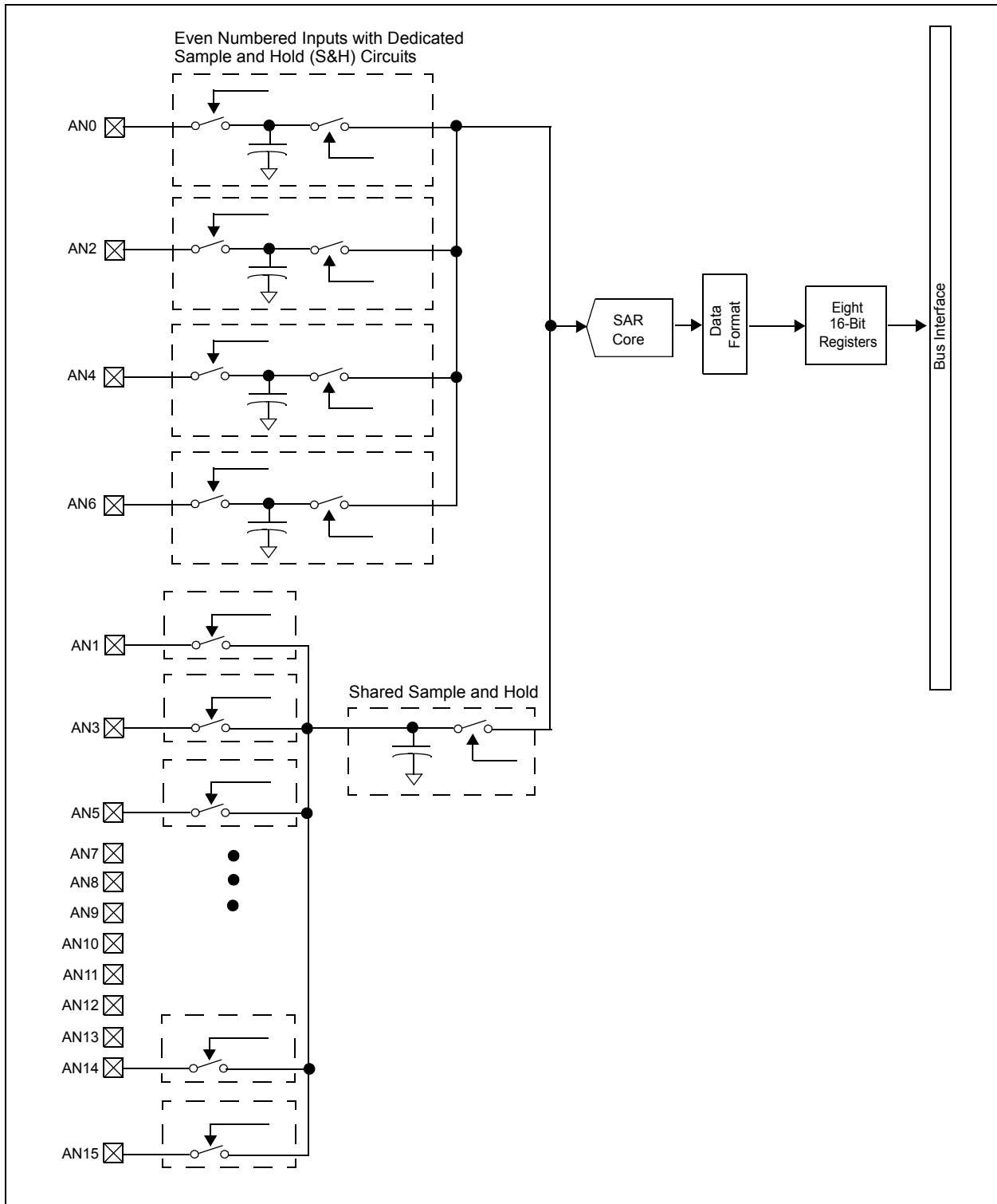
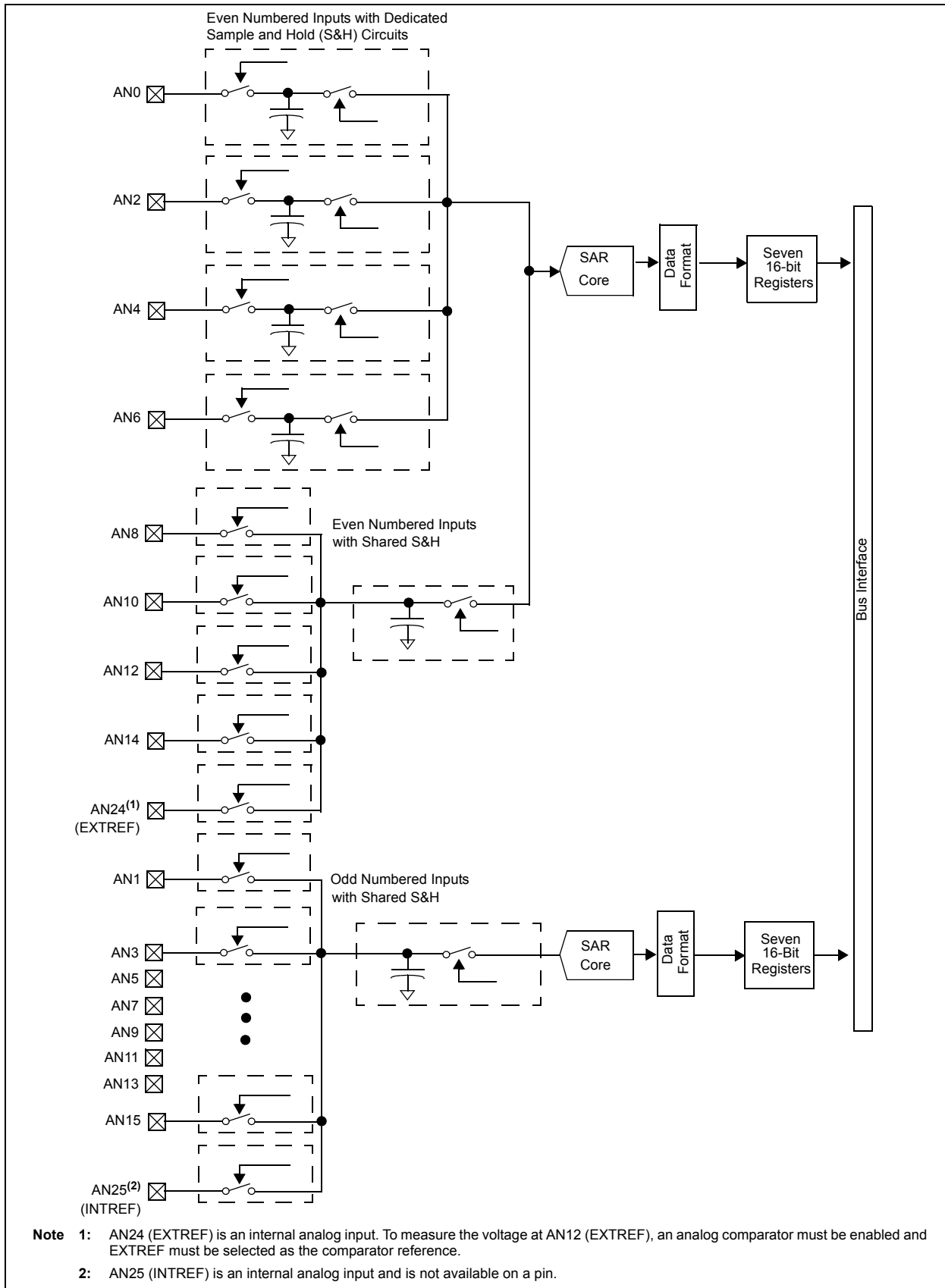


FIGURE 22-2: ADC BLOCK DIAGRAM FOR dsPIC33FJ32GS606 AND dsPIC33FJ64GS606 DEVICES WITH TWO SARS



**FIGURE 22-3: ADC BLOCK DIAGRAM FOR dsPIC33FJ32GS608 AND dsPIC33FJ64GS608 DEVICES WITH TWO SARS**

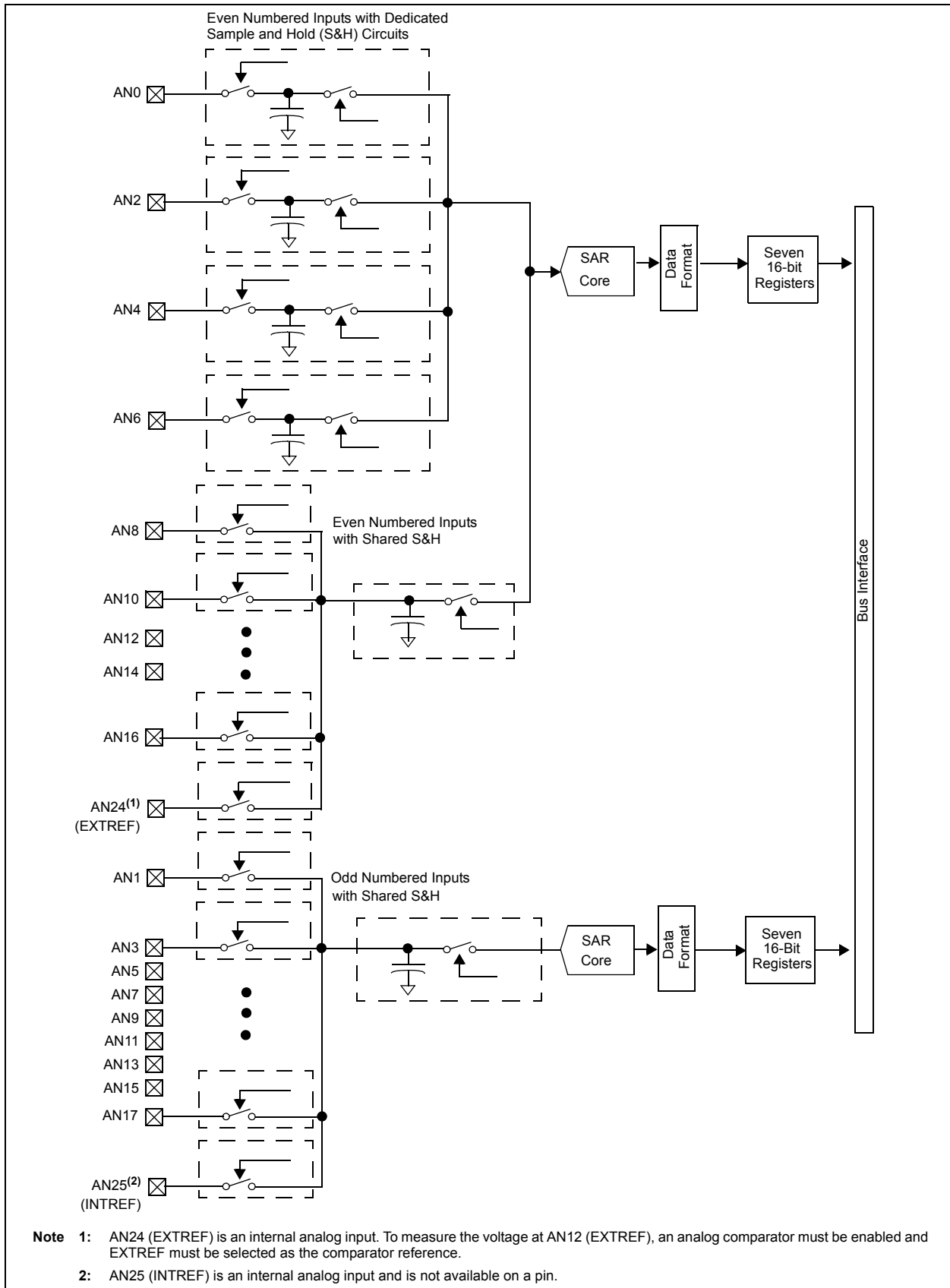
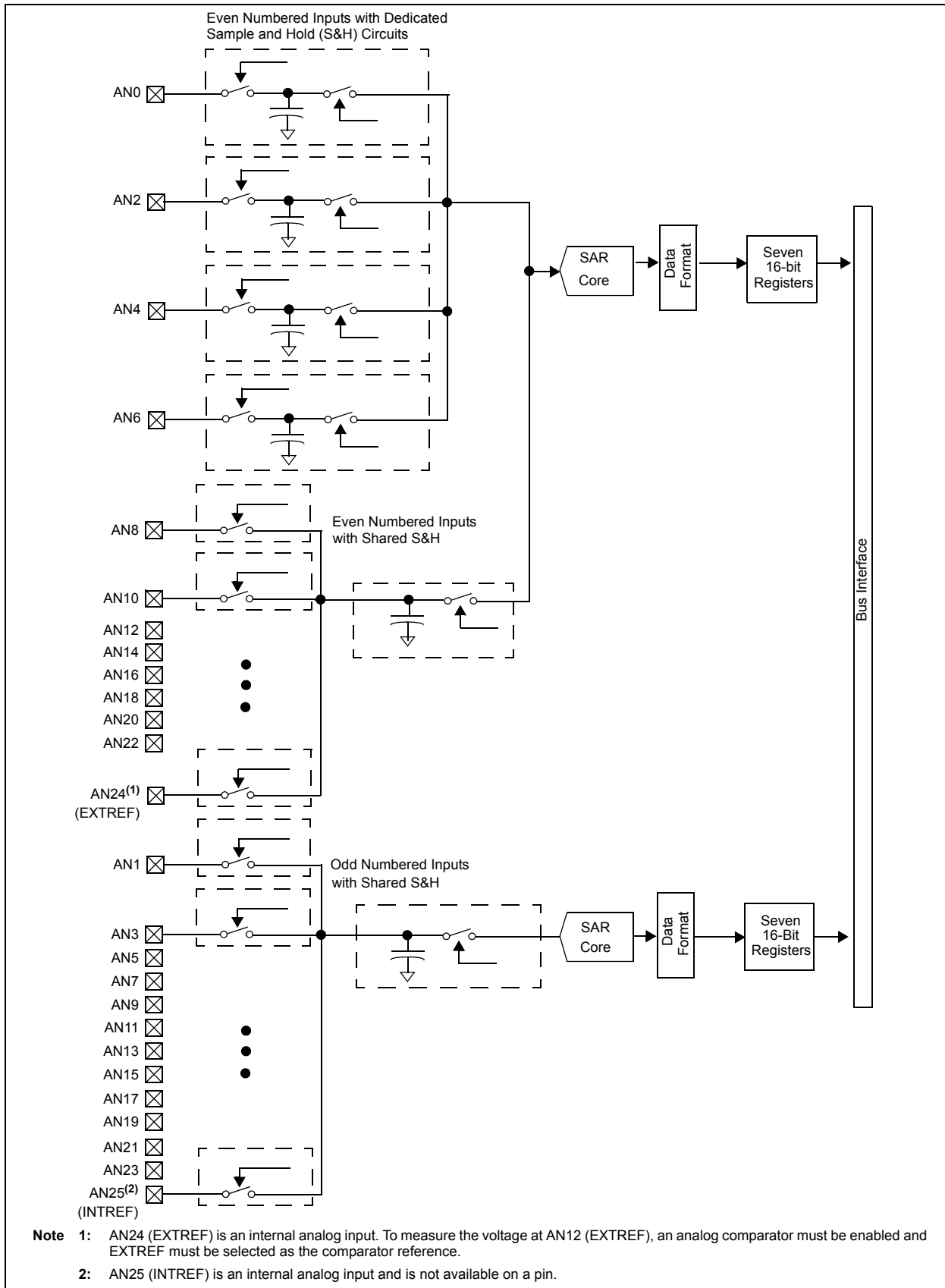


FIGURE 22-4: ADC BLOCK DIAGRAM FOR dsPIC33FJ32GS610 AND dsPIC33FJ64GS610 DEVICES WITH TWO SARS



## REGISTER 22-1: ADON: ADC CONTROL REGISTER

R/W-0	U-0	R/W-0	R/W-0	U-0	R/W-0	U-0	R/W-0
ADON	—	ADSIDL	SLOWCLK <sup>(1)</sup>	—	GSWTRG	—	FORM <sup>(1)</sup>
bit 15							bit 8
R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-1	R/W-1
EIE <sup>(1)</sup>	ORDER <sup>(1,2)</sup>	SEQSAMP <sup>(1,2)</sup>	ASYNCAMP <sup>(1)</sup>	—	ADCS<2:0> <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      **ADON:** ADC Module Operating Mode bit  
             1 = ADC module is operating  
             0 = ADC module 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      **SLOWCLK:** Enable The Slow Clock Divider bit<sup>(1)</sup>  
             1 = ADC is clocked by the auxiliary PLL (ACLK)  
             0 = ADC is clock by the primary PLL (FVCO)
- bit 11      **Unimplemented:** Read as '0'
- bit 10      **GSWTRG:** Global Software Trigger bit  
             When this bit is set by the user, it will trigger conversions if selected by the TRGSRC<4:0> bits in the ADCPCx registers. This bit must be cleared by the user prior to initiating another global trigger (i.e., this bit is not auto-clearing).
- bit 9        **Unimplemented:** Read as '0'
- bit 8        **FORM:** Data Output Format bit<sup>(1)</sup>  
             1 = Fractional (DOUT = dddd dddd dd00 0000)  
             0 = Integer (DOUT = 0000 00dd dddd dddd)
- bit 7        **EIE:** Early Interrupt Enable bit<sup>(1)</sup>  
             1 = Interrupt is generated after first conversion is completed  
             0 = Interrupt is generated after second conversion is completed
- bit 6        **ORDER:** Conversion Order bit<sup>(1,2)</sup>  
             1 = Odd numbered analog input is converted first, followed by conversion of even numbered input  
             0 = Even numbered analog input is converted first, followed by conversion of odd numbered input
- bit 5        **SEQSAMP:** Sequential Sample Enable bit<sup>(1,2)</sup>  
             1 = Shared Sample and Hold (S&H) circuit is sampled at the start of the second conversion if ORDER = 0. If ORDER = 1, then the shared S&H is sampled at the start of the first conversion.  
             0 = Shared S&H is sampled at the same time the dedicated S&H is sampled if the shared S&H is not currently busy with an existing conversion process. If the shared S&H is busy at the time the dedicated S&H is sampled, then the shared S&H will sample at the start of the new conversion cycle.

**Note 1:** This control bit can only be changed while the ADC is disabled (ADON = 0).

**2:** This control bit is only active on devices that have one SAR.

## REGISTER 22-1: ADCON: ADC CONTROL REGISTER (CONTINUED)

- bit 4      **ASYNCSAMP:** Asynchronous Dedicated S&H Sampling Enable bit<sup>(1)</sup>  
1 = The dedicated S&H is constantly sampling and then terminates sampling as soon as the trigger pulse is detected.  
0 = The dedicated S&H starts sampling when the trigger event is detected and completes the sampling process in two ADC clock cycles.
- bit 3      **Unimplemented:** Read as '0'
- bit 2-0    **ADCS<2:0>:** Analog-to-Digital Conversion Clock Divider Select bits<sup>(1)</sup>  
111 = FADC/8  
110 = FADC/7  
101 = FADC/6  
100 = FADC/5  
011 = FADC/4 (default)  
010 = FADC/3  
001 = FADC/2  
000 = FADC/1

- Note 1:** This control bit can only be changed while the ADC is disabled (ADON = 0).  
**2:** This control bit is only active on devices that have one SAR.



## REGISTER 22-2: ADSTAT: ADC STATUS REGISTER

U-0	U-0	U-0	R/C-0, HS	R/C-0, HS	R/C-0, HS	R/C-0, HS	R/C-0, HS
—	—	—	P12RDY	P11RDY	P10RDY	P9RDY	P8RDY
bit 15						bit 8	

R/C-0, HS	R/C-0, HS	R/C-0, HS	R/C-0, HS	R/C-0, HS	R/C-0, HS	R/C-0, HS	R/C-0, HS
P7RDY	P6RDY	P5RDY	P4RDY	P3RDY	P2RDY	P1RDY	P0RDY
bit 7						bit 0	

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

- bit 15-13     **Unimplemented:** Read as '0'
- bit 6        **P12RDY:** Conversion Data for Pair 12 Ready bit  
Bit is set when data is ready in buffer, cleared when a '0' is written to this bit.
- bit 5        **P11RDY:** Conversion Data for Pair 11 Ready bit  
Bit is set when data is ready in buffer, cleared when a '0' is written to this bit.
- bit 4        **P10RDY:** Conversion Data for Pair 10 Ready bit  
Bit is set when data is ready in buffer, cleared when a '0' is written to this bit.
- bit 3        **P9RDY:** Conversion Data for Pair 9 Ready bit  
Bit is set when data is ready in buffer, cleared when a '0' is written to this bit.
- bit 2        **P8RDY:** Conversion Data for Pair 8 Ready bit  
Bit is set when data is ready in buffer, cleared when a '0' is written to this bit.
- bit 1        **P7RDY:** Conversion Data for Pair 7 Ready bit  
Bit is set when data is ready in buffer, cleared when a '0' is written to this bit.
- bit 6        **P6RDY:** Conversion Data for Pair 6 Ready bit  
Bit is set when data is ready in buffer, cleared when a '0' is written to this bit.
- bit 5        **P5RDY:** Conversion Data for Pair 5 Ready bit  
Bit is set when data is ready in buffer, cleared when a '0' is written to this bit.
- bit 4        **P4RDY:** Conversion Data for Pair 4 Ready bit  
Bit is set when data is ready in buffer, cleared when a '0' is written to this bit.
- bit 3        **P3RDY:** Conversion Data for Pair 3 Ready bit  
Bit is set when data is ready in buffer, cleared when a '0' is written to this bit.
- bit 2        **P2RDY:** Conversion Data for Pair 2 Ready bit  
Bit is set when data is ready in buffer, cleared when a '0' is written to this bit.
- bit 1        **P1RDY:** Conversion Data for Pair 1 Ready bit  
Bit is set when data is ready in buffer, cleared when a '0' is written to this bit.
- bit 0        **P0RDY:** Conversion Data for Pair 0 Ready bit  
Bit is set when data is ready in buffer, cleared when a '0' is written to this bit.

**Note:** Not all PxRDY bits are available on all devices. See [Figure 22-1](#), [Figure 22-2](#), [Figure 22-3](#), and [Figure 22-4](#) for the available analog inputs.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 22-3: ADBASE: ADC BASE REGISTER<sup>(1,2)</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
ADBASE<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0
ADBASE<7:1>							—
bit 7							bit 0

### Legend:

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

bit 15-1      **ADBASE<15:1>**: This register contains the base address of the user's ADC Interrupt Service Routine jump table. This register, when read, contains the sum of the ADBASE register contents and the encoded value of the P<sub>x</sub>RDY Status bits.

The encoder logic provides the bit number of the highest priority P<sub>x</sub>RDY bits where P<sub>0</sub>RDY is the highest priority, and P<sub>6</sub>RDY is the lowest priority.

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

- Note 1:** The encoding results are shifted left two bits so bits 1-0 of the result are always zero.
- Note 2:** As an alternative to using the ADBASE Register, the ADCP<sub>0</sub>-ADCP<sub>12</sub> ADC Pair Conversion Complete Interrupts can be used to invoke A to D conversion completion routines for individual ADC input pairs.

## REGISTER 22-4: ADPCFG: ADC PORT CONFIGURATION REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PCFG15	PCFG14	PCFG13	PCFG12	PCFG11	PCFG10	PCFG9	PCFG8
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
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, analog-to-digital input multiplexer connected to AVss  
                                     0 = Port pin in Analog mode, port read input disabled, analog-to-digital samples pin voltage

**Note:** Not all PCFGx bits are available on all devices. See [Figure 22-1](#), [Figure 22-2](#), [Figure 22-3](#), and [Figure 22-4](#) for the available analog inputs (PCFGx = ANx, where x = 0-15).

## REGISTER 22-5: ADPCFG2: ADC PORT CONFIGURATION 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	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	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 15-8                      **Unimplemented:** Read as '0'  
 bit 7-0                      **PCFG<23:16>**: ADC Port Configuration Control bits  
                                     1 = Port pin in Digital mode, port read input enabled, analog-to-digital input multiplexer connected to AVss  
                                     0 = Port pin in Analog mode, port read input disabled, analog-to-digital samples pin voltage

**Note:** Not all PCFGx bits are available on all devices. See [Figure 22-1](#), [Figure 22-2](#), [Figure 22-3](#), and [Figure 22-4](#) for the available analog inputs (PCFGx = ANx, where x can be 0 through 15).

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 22-6: ADCPC0: ANALOG-TO-DIGITAL CONVERT PAIR CONTROL REGISTER 0

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IRQEN1	PEND1	SWTRG1	TRGSRC1<4:0>				
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
IRQEN0	PEND0	SWTRG0	TRGSRC0<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            **IRQEN1**: Interrupt Request Enable 1 bit  
                  1 = Enable IRQ generation when requested conversion of channels AN3 and AN2 is completed  
                  0 = IRQ is not generated
- bit 14            **PEND1**: Pending Conversion Status 1 bit  
                  1 = Conversion of channels AN3 and AN2 is pending. Set when selected trigger is asserted  
                  0 = Conversion is complete
- bit 13            **SWTRG1**: Software Trigger 1 bit  
                  1 = Start conversion of AN3 and AN2 (if selected in TRGSRC bits)<sup>(1)</sup>  
                  This bit is automatically cleared by hardware when the PEND1 bit is set.  
                  0 = Conversion is not started

**Note 1:** The trigger source must be set as a global software trigger prior to setting this bit to '1'. If other conversions are in progress, the conversion is performed when the conversion resources are available.

**REGISTER 22-6: ADCPC0: ANALOG-TO-DIGITAL CONVERT PAIR CONTROL REGISTER 0  
(CONTINUED)**

- bit 12-8      **TRGSRC1<4:0>**: Trigger 1 Source Selection bits  
Selects trigger source for conversion of analog channels AN3 and AN2.  
11111 = Timer2 period match  
11110 = PWM Generator 8 current-limit ADC trigger  
11101 = PWM Generator 7 current-limit ADC trigger  
11100 = PWM Generator 6 current-limit ADC trigger  
11011 = PWM Generator 5 current-limit ADC trigger  
11010 = PWM Generator 4 current-limit ADC trigger  
11001 = PWM Generator 3 current-limit ADC trigger  
11000 = PWM Generator 2 current-limit ADC trigger  
10111 = PWM Generator 1 current-limit ADC trigger  
10110 = PWM Generator 9 secondary trigger selected  
10101 = PWM Generator 8 secondary trigger selected  
10100 = PWM Generator 7 secondary trigger selected  
10011 = PWM Generator 6 secondary trigger selected  
10010 = PWM Generator 5 secondary trigger selected  
10001 = PWM Generator 4 secondary trigger selected  
10000 = PWM Generator 3 secondary trigger selected  
01111 = PWM Generator 2 secondary trigger selected  
01110 = PWM Generator 1 secondary trigger selected  
01101 = PWM secondary special event trigger selected  
01100 = Timer1 period match  
01011 = PWM Generator 8 primary trigger selected  
01010 = PWM Generator 7 primary trigger selected  
01001 = PWM Generator 6 primary trigger selected  
01000 = PWM Generator 5 primary trigger selected  
00111 = PWM Generator 4 primary trigger selected  
00110 = PWM Generator 3 primary trigger selected  
00101 = PWM Generator 2 primary trigger selected  
00100 = PWM Generator 1 primary trigger selected  
00011 = PWM Special Event Trigger selected  
00010 = Global software trigger selected  
00001 = Individual software trigger selected  
00000 = No conversion enabled
- bit 7      **IRQEN0**: Interrupt Request Enable 0 bit  
1 = Enable IRQ generation when requested conversion of channels AN1 and AN0 is completed  
0 = IRQ is not generated
- bit 6      **PEND0**: Pending Conversion Status 0 bit  
1 = Conversion of channels AN1 and AN0 is pending; set when selected trigger is asserted  
0 = Conversion is complete
- bit 5      **SWTRG0**: Software Trigger 0 bit  
1 = Start conversion of AN1 and AN0 (if selected by TRGSRC bits)<sup>(1)</sup>  
This bit is automatically cleared by hardware when the PEND0 bit is set.  
0 = Conversion is not started.

**Note 1:** The trigger source must be set as a global software trigger prior to setting this bit to '1'. If other conversions are in progress, the conversion is performed when the conversion resources are available.

## REGISTER 22-6: ADCPC0: ANALOG-TO-DIGITAL CONVERT PAIR CONTROL REGISTER 0 (CONTINUED)

bit 4-0      **TRGSRC0<4:0>**: Trigger 0 Source Selection bits  
Selects trigger source for conversion of analog channels AN1 and AN0.

- 11111 = Timer2 period match
- 11110 = PWM Generator 8 current-limit ADC trigger
- 11101 = PWM Generator 7 current-limit ADC trigger
- 11100 = PWM Generator 6 current-limit ADC trigger
- 11011 = PWM Generator 5 current-limit ADC trigger
- 11010 = PWM Generator 4 current-limit ADC trigger
- 11001 = PWM Generator 3 current-limit ADC trigger
- 11000 = PWM Generator 2 current-limit ADC trigger
- 10111 = PWM Generator 1 current-limit ADC trigger
- 10110 = PWM Generator 9 secondary trigger selected
- 10101 = PWM Generator 8 secondary trigger selected
- 10100 = PWM Generator 7 secondary trigger selected
- 10011 = PWM Generator 6 secondary trigger selected
- 10010 = PWM Generator 5 secondary trigger selected
- 10001 = PWM Generator 4 secondary trigger selected
- 10000 = PWM Generator 3 secondary trigger selected
- 01111 = PWM Generator 2 secondary trigger selected
- 01110 = PWM Generator 1 secondary trigger selected
- 01101 = PWM secondary special event trigger selected
- 01100 = Timer1 period match
- 01011 = PWM Generator 8 primary trigger selected
- 01010 = PWM Generator 7 primary trigger selected
- 01001 = PWM Generator 6 primary trigger selected
- 01000 = PWM Generator 5 primary trigger selected
- 00111 = PWM Generator 4 primary trigger selected
- 00110 = PWM Generator 3 primary trigger selected
- 00101 = PWM Generator 2 primary trigger selected
- 00100 = PWM Generator 1 primary trigger selected
- 00011 = PWM Special Event Trigger selected
- 00010 = Global software trigger selected
- 00001 = Individual software trigger selected
- 00000 = No conversion enabled

**Note 1:** The trigger source must be set as a global software trigger prior to setting this bit to '1'. If other conversions are in progress, the conversion is performed when the conversion resources are available.

## REGISTER 22-7: ADCPC1: ANALOG-TO-DIGITAL CONVERT PAIR 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
IRQEN3	PEND3	SWTRG3	TRGSRC3<4:0>				
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
IRQEN2	PEND2	SWTRG2	TRGSRC2<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      **IRQEN3:** Interrupt Request Enable 3 bit  
             1 = Enable IRQ generation when requested conversion of channels AN7 and AN6 is completed  
             0 = IRQ is not generated
  
- bit 14      **PEND3:** Pending Conversion Status 3 bit  
             1 = Conversion of channels AN7 and AN6 is pending. Set when selected trigger is asserted  
             0 = Conversion is complete
  
- bit 13      **SWTRG3:** Software Trigger 3 bit  
             1 = Start conversion of AN7 and AN6 (if selected in TRGSRC bits)<sup>(1)</sup>  
             This bit is automatically cleared by hardware when the PEND3 bit is set.  
             0 = Conversion is not started

**Note 1:** The trigger source must be set as a global software trigger prior to setting this bit to '1'. If other conversions are in progress, the conversion is performed when the conversion resources are available.

## REGISTER 22-7: ADCPC1: ANALOG-TO-DIGITAL CONVERT PAIR CONTROL REGISTER 1 (CONTINUED)

- bit 12-8      **TRGSRC3<4:0>**: Trigger 3 Source Selection bits<sup>(1)</sup>  
Selects trigger source for conversion of analog channels AN7 and AN6.  
11111 = Timer2 period match  
11110 = PWM Generator 8 current-limit ADC trigger  
11101 = PWM Generator 7 current-limit ADC trigger  
11100 = PWM Generator 6 current-limit ADC trigger  
11011 = PWM Generator 5 current-limit ADC trigger  
11010 = PWM Generator 4 current-limit ADC trigger  
11001 = PWM Generator 3 current-limit ADC trigger  
11000 = PWM Generator 2 current-limit ADC trigger  
10111 = PWM Generator 1 current-limit ADC trigger  
10110 = PWM Generator 9 secondary trigger selected  
10101 = PWM Generator 8 secondary trigger selected  
10100 = PWM Generator 7 secondary trigger selected  
10011 = PWM Generator 6 secondary trigger selected  
10010 = PWM Generator 5 secondary trigger selected  
10001 = PWM Generator 4 secondary trigger selected  
10000 = PWM Generator 3 secondary trigger selected  
01111 = PWM Generator 2 secondary trigger selected  
01110 = PWM Generator 1 secondary trigger selected  
01101 = PWM secondary special event trigger selected  
01100 = Timer1 period match  
01011 = PWM Generator 8 primary trigger selected  
01010 = PWM Generator 7 primary trigger selected  
01001 = PWM Generator 6 primary trigger selected  
01000 = PWM Generator 5 primary trigger selected  
00111 = PWM Generator 4 primary trigger selected  
00110 = PWM Generator 3 primary trigger selected  
00101 = PWM Generator 2 primary trigger selected  
00100 = PWM Generator 1 primary trigger selected  
00011 = PWM Special Event Trigger selected  
00010 = Global software trigger selected  
00001 = Individual software trigger selected  
00000 = No conversion enabled
- bit 7      **IRQEN2**: Interrupt Request Enable 2 bit  
1 = Enable IRQ generation when requested conversion of channels AN5 and AN4 is completed  
0 = IRQ is not generated
- bit 6      **PEND2**: Pending Conversion Status 2 bit  
1 = Conversion of channels AN5 and AN4 is pending; set when selected trigger is asserted.  
0 = Conversion is complete
- bit 5      **SWTRG2**: Software Trigger 2 bit  
1 = Start conversion of AN5 and AN4 (if selected by TRGSRC bits)<sup>(1)</sup>  
This bit is automatically cleared by hardware when the PEND2 bit is set.  
0 = Conversion is not started

**Note 1:** The trigger source must be set as a global software trigger prior to setting this bit to '1'. If other conversions are in progress, the conversion is performed when the conversion resources are available.



## REGISTER 22-7: ADCPC1: ANALOG-TO-DIGITAL CONVERT PAIR CONTROL REGISTER 1 (CONTINUED)

bit 4-0      **TRGSRC2<4:0>**: Trigger 2 Source Selection bits  
Selects trigger source for conversion of analog channels AN5 and AN4.

- 11111 = Timer2 period match
- 11110 = PWM Generator 8 current-limit ADC trigger
- 11101 = PWM Generator 7 current-limit ADC trigger
- 11100 = PWM Generator 6 current-limit ADC trigger
- 11011 = PWM Generator 5 current-limit ADC trigger
- 11010 = PWM Generator 4 current-limit ADC trigger
- 11001 = PWM Generator 3 current-limit ADC trigger
- 11000 = PWM Generator 2 current-limit ADC trigger
- 10111 = PWM Generator 1 current-limit ADC trigger
- 10110 = PWM Generator 9 secondary trigger selected
- 10101 = PWM Generator 8 secondary trigger selected
- 10100 = PWM Generator 7 secondary trigger selected
- 10011 = PWM Generator 6 secondary trigger selected
- 10010 = PWM Generator 5 secondary trigger selected
- 10001 = PWM Generator 4 secondary trigger selected
- 10000 = PWM Generator 3 secondary trigger selected
- 01111 = PWM Generator 2 secondary trigger selected
- 01110 = PWM Generator 1 secondary trigger selected
- 01101 = PWM secondary special event trigger selected
- 01100 = Timer1 period match
- 01011 = PWM Generator 8 primary trigger selected
- 01010 = PWM Generator 7 primary trigger selected
- 01001 = PWM Generator 6 primary trigger selected
- 01000 = PWM Generator 5 primary trigger selected
- 00111 = PWM Generator 4 primary trigger selected
- 00110 = PWM Generator 3 primary trigger selected
- 00101 = PWM Generator 2 primary trigger selected
- 00100 = PWM Generator 1 primary trigger selected
- 00011 = PWM Special Event Trigger selected
- 00010 = Global software trigger selected
- 00001 = Individual software trigger selected
- 00000 = No conversion enabled

**Note 1:** The trigger source must be set as a global software trigger prior to setting this bit to '1'. If other conversions are in progress, the conversion is performed when the conversion resources are available.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 22-8: ADCPC2: ANALOG-TO-DIGITAL CONVERT PAIR 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
IRQEN5	PEND5	SWTRG5	TRGSRC5<4:0>				
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
IRQEN4	PEND4	SWTRG4	TRGSRC4<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      **IRQEN5:** Interrupt Request Enable 5 bit  
1 = Enable IRQ generation when requested conversion of channels AN11 and AN10 is completed  
0 = IRQ is not generated
- bit 14      **PEND5:** Pending Conversion Status 5 bit  
1 = Conversion of channels AN11 and AN10 is pending; set when selected trigger is asserted  
0 = Conversion is complete
- bit 13      **SWTRG5:** Software Trigger 5 bit  
1 = Start conversion of AN11 and AN10 (if selected in TRGSRC bits)<sup>(1)</sup>  
This bit is automatically cleared by hardware when the PEND5 bit is set.  
0 = Conversion is not started

**Note 1:** The trigger source must be set as a global software trigger prior to setting this bit to '1'. If other conversions are in progress, the conversion is performed when the conversion resources are available.

## REGISTER 22-8: ADCPC2: ANALOG-TO-DIGITAL CONVERT PAIR CONTROL REGISTER 2 (CONTINUED)

- bit 12-8      **TRGSRC5<4:0>**: Trigger 5 Source Selection bits  
Selects trigger source for conversion of analog channels AN11 and AN10.  
11111 = Timer2 period match  
11110 = PWM Generator 8 current-limit ADC trigger  
11101 = PWM Generator 7 current-limit ADC trigger  
11100 = PWM Generator 6 current-limit ADC trigger  
11011 = PWM Generator 5 current-limit ADC trigger  
11010 = PWM Generator 4 current-limit ADC trigger  
11001 = PWM Generator 3 current-limit ADC trigger  
11000 = PWM Generator 2 current-limit ADC trigger  
10111 = PWM Generator 1 current-limit ADC trigger  
10110 = PWM Generator 9 secondary trigger selected  
10101 = PWM Generator 8 secondary trigger selected  
10100 = PWM Generator 7 secondary trigger selected  
10011 = PWM Generator 6 secondary trigger selected  
10010 = PWM Generator 5 secondary trigger selected  
10001 = PWM Generator 4 secondary trigger selected  
10000 = PWM Generator 3 secondary trigger selected  
01111 = PWM Generator 2 secondary trigger selected  
01110 = PWM Generator 1 secondary trigger selected  
01101 = PWM secondary special event trigger selected  
01100 = Timer1 period match  
01011 = PWM Generator 8 primary trigger selected  
01010 = PWM Generator 7 primary trigger selected  
01001 = PWM Generator 6 primary trigger selected  
01000 = PWM Generator 5 primary trigger selected  
00111 = PWM Generator 4 primary trigger selected  
00110 = PWM Generator 3 primary trigger selected  
00101 = PWM Generator 2 primary trigger selected  
00100 = PWM Generator 1 primary trigger selected  
00011 = PWM Special Event Trigger selected  
00010 = Global software trigger selected  
00001 = Individual software trigger selected  
00000 = No conversion enabled
- bit 7      **IRQEN4**: Interrupt Request Enable 4 bit  
1 = Enable IRQ generation when requested conversion of channels AN9 and AN8 is completed  
0 = IRQ is not generated
- bit 6      **PEND4**: Pending Conversion Status 4 bit  
1 = Conversion of channels AN9 and AN8 is pending; set when selected trigger is asserted  
0 = Conversion is complete
- bit 5      **SWTRG4**: Software Trigger4 bit  
1 = Start conversion of AN9 and AN8 (if selected by TRGSRC bits)<sup>(1)</sup>  
This bit is automatically cleared by hardware when the PEND4 bit is set.  
0 = Conversion is not started

**Note 1:** The trigger source must be set as a global software trigger prior to setting this bit to '1'. If other conversions are in progress, the conversion is performed when the conversion resources are available.

## REGISTER 22-8: ADCPC2: ANALOG-TO-DIGITAL CONVERT PAIR CONTROL REGISTER 2 (CONTINUED)

bit 4-0      **TRGSRC4<4:0>**: Trigger 4 Source Selection bits  
Selects trigger source for conversion of analog channels AN9 and AN8.

- 11111 = Timer2 period match
- 11110 = PWM Generator 8 current-limit ADC trigger
- 11101 = PWM Generator 7 current-limit ADC trigger
- 11100 = PWM Generator 6 current-limit ADC trigger
- 11011 = PWM Generator 5 current-limit ADC trigger
- 11010 = PWM Generator 4 current-limit ADC trigger
- 11001 = PWM Generator 3 current-limit ADC trigger
- 11000 = PWM Generator 2 current-limit ADC trigger
- 10111 = PWM Generator 1 current-limit ADC trigger
- 10110 = PWM Generator 9 secondary trigger selected
- 10101 = PWM Generator 8 secondary trigger selected
- 10100 = PWM Generator 7 secondary trigger selected
- 10011 = PWM Generator 6 secondary trigger selected
- 10010 = PWM Generator 5 secondary trigger selected
- 10001 = PWM Generator 4 secondary trigger selected
- 10000 = PWM Generator 3 secondary trigger selected
- 01111 = PWM Generator 2 secondary trigger selected
- 01110 = PWM Generator 1 secondary trigger selected
- 01101 = PWM secondary special event trigger selected
- 01100 = Timer1 period match
- 01011 = PWM Generator 8 primary trigger selected
- 01010 = PWM Generator 7 primary trigger selected
- 01001 = PWM Generator 6 primary trigger selected
- 01000 = PWM Generator 5 primary trigger selected
- 00111 = PWM Generator 4 primary trigger selected
- 00110 = PWM Generator 3 primary trigger selected
- 00101 = PWM Generator 2 primary trigger selected
- 00100 = PWM Generator 1 primary trigger selected
- 00011 = PWM Special Event Trigger selected
- 00010 = Global software trigger selected
- 00001 = Individual software trigger selected
- 00000 = No conversion enabled

**Note 1:** The trigger source must be set as a global software trigger prior to setting this bit to '1'. If other conversions are in progress, the conversion is performed when the conversion resources are available.

## REGISTER 22-9: ADCPC3: ANALOG-TO-DIGITAL CONVERT PAIR CONTROL 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
IRQEN7	PEND7	SWTRG7	TRGSRC7<4:0>				
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
IRQEN6	PEND6	SWTRG6	TRGSRC6<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      **IRQEN7:** Interrupt Request Enable 7 bit  
 1 = Enable IRQ generation when requested conversion of channels AN15 and AN14 is completed  
 0 = IRQ is not generated
- bit 14      **PEND7:** Pending Conversion Status 7 bit  
 1 = Conversion of channels AN15 and AN14 is pending; set when selected trigger is asserted  
 0 = Conversion is complete
- bit 13      **SWTRG7:** Software Trigger 7 bit  
 1 = Start conversion of AN15 and AN14 (if selected in TRGSRC bits)<sup>(1)</sup>  
 This bit is automatically cleared by hardware when the PEND7 bit is set.  
 0 = Conversion is not started

**Note 1:** The trigger source must be set as a global software trigger prior to setting this bit to '1'. If other conversions are in progress, the conversion is performed when the conversion resources are available.

## REGISTER 22-9: ADCPC3: ANALOG-TO-DIGITAL CONVERT PAIR CONTROL REGISTER 3 (CONTINUED)

- bit 12-8 **TRGSRC7<4:0>**: Trigger 7 Source Selection bits  
Selects trigger source for conversion of analog channels AN15 and 14.  
11111 = Timer2 period match  
11110 = PWM Generator 8 current-limit ADC trigger  
11101 = PWM Generator 7 current-limit ADC trigger  
11100 = PWM Generator 6 current-limit ADC trigger  
11011 = PWM Generator 5 current-limit ADC trigger  
11010 = PWM Generator 4 current-limit ADC trigger  
11001 = PWM Generator 3 current-limit ADC trigger  
11000 = PWM Generator 2 current-limit ADC trigger  
10111 = PWM Generator 1 current-limit ADC trigger  
10110 = PWM Generator 9 secondary trigger selected  
10101 = PWM Generator 8 secondary trigger selected  
10100 = PWM Generator 7 secondary trigger selected  
10011 = PWM Generator 6 secondary trigger selected  
10010 = PWM Generator 5 secondary trigger selected  
10001 = PWM Generator 4 secondary trigger selected  
10000 = PWM Generator 3 secondary trigger selected  
01111 = PWM Generator 2 secondary trigger selected  
01110 = PWM Generator 1 secondary trigger selected  
01101 = PWM secondary special event trigger selected  
01100 = Timer1 period match  
01011 = PWM Generator 8 primary trigger selected  
01010 = PWM Generator 7 primary trigger selected  
01001 = PWM Generator 6 primary trigger selected  
01000 = PWM Generator 5 primary trigger selected  
00111 = PWM Generator 4 primary trigger selected  
00110 = PWM Generator 3 primary trigger selected  
00101 = PWM Generator 2 primary trigger selected  
00100 = PWM Generator 1 primary trigger selected  
00011 = PWM Special Event Trigger selected  
00010 = Global software trigger selected  
00001 = Individual software trigger selected  
00000 = No conversion enabled
- bit 7 **IRQEN6**: Interrupt Request Enable 6 bit  
1 = Enable IRQ generation when requested conversion of channels AN13 and AN12 is completed  
0 = IRQ is not generated
- bit 6 **PEND6**: Pending Conversion Status 6 bit  
1 = Conversion of channels AN13 and AN12 is pending; set when selected trigger is asserted  
0 = Conversion is complete
- bit 5 **SWTRG6**: Software Trigger 6 bit  
1 = Start conversion of AN13 and AN12 (if selected by TRGSRC bits)<sup>(1)</sup>  
This bit is automatically cleared by hardware when the PEND6 bit is set.  
0 = Conversion is not started

**Note 1:** The trigger source must be set as a global software trigger prior to setting this bit to '1'. If other conversions are in progress, the conversion is performed when the conversion resources are available.

## REGISTER 22-9: ADCPC3: ANALOG-TO-DIGITAL CONVERT PAIR CONTROL REGISTER 3 (CONTINUED)

bit 4-0      **TRGSRC6<4:0>**: Trigger 6 Source Selection bits  
Selects trigger source for conversion of analog channels AN13 and AN12.

- 11111 = Timer2 period match
- 11110 = PWM Generator 8 current-limit ADC trigger
- 11101 = PWM Generator 7 current-limit ADC trigger
- 11100 = PWM Generator 6 current-limit ADC trigger
- 11011 = PWM Generator 5 current-limit ADC trigger
- 11010 = PWM Generator 4 current-limit ADC trigger
- 11001 = PWM Generator 3 current-limit ADC trigger
- 11000 = PWM Generator 2 current-limit ADC trigger
- 10111 = PWM Generator 1 current-limit ADC trigger
- 10110 = PWM Generator 9 secondary trigger selected
- 10101 = PWM Generator 8 secondary trigger selected
- 10100 = PWM Generator 7 secondary trigger selected
- 10011 = PWM Generator 6 secondary trigger selected
- 10010 = PWM Generator 5 secondary trigger selected
- 10001 = PWM Generator 4 secondary trigger selected
- 10000 = PWM Generator 3 secondary trigger selected
- 01111 = PWM Generator 2 secondary trigger selected
- 01110 = PWM Generator 1 secondary trigger selected
- 01101 = PWM secondary special event trigger selected
- 01100 = Timer1 period match
- 01011 = PWM Generator 8 primary trigger selected
- 01010 = PWM Generator 7 primary trigger selected
- 01001 = PWM Generator 6 primary trigger selected
- 01000 = PWM Generator 5 primary trigger selected
- 00111 = PWM Generator 4 primary trigger selected
- 00110 = PWM Generator 3 primary trigger selected
- 00101 = PWM Generator 2 primary trigger selected
- 00100 = PWM Generator 1 primary trigger selected
- 00011 = PWM Special Event Trigger selected
- 00010 = Global software trigger selected
- 00001 = Individual software trigger selected
- 00000 = No conversion enabled

**Note 1:** The trigger source must be set as a global software trigger prior to setting this bit to '1'. If other conversions are in progress, the conversion is performed when the conversion resources are available.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 22-10: ADCPC4: ANALOG-TO-DIGITAL CONVERT PAIR CONTROL 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
IRQEN9	PEND9	SWTRG9	TRGSRC9<4:0>				
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
IRQEN8	PEND8	SWTRG8	TRGSRC8<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      **IRQEN9:** Interrupt Request Enable 9 bit  
 1 = Enable IRQ generation when requested conversion of channels AN19 and AN18 is completed  
 0 = IRQ is not generated
- bit 14      **PEND9:** Pending Conversion Status 9 bit  
 1 = Conversion of channels AN19 and AN18 is pending; set when selected trigger is asserted  
 0 = Conversion is complete
- bit 13      **SWTRG9:** Software Trigger 9 bit  
 1 = Start conversion of AN19 and AN18 (if selected in TRGSRC bits)<sup>(1)</sup>  
 This bit is automatically cleared by hardware when the PEND9 bit is set.  
 0 = Conversion is not started

**Note 1:** The trigger source must be set as a global software trigger prior to setting this bit to '1'. If other conversions are in progress, the conversion is performed when the conversion resources are available.



## REGISTER 22-10: ADCPC4: ANALOG-TO-DIGITAL CONVERT PAIR CONTROL REGISTER 4 (CONTINUED)

- bit 12-8      **TRGSRC9<4:0>**: Trigger 9 Source Selection bits  
Selects trigger source for conversion of analog channels AN19 and AN18.  
11111 = Timer2 period match  
11110 = PWM Generator 8 current-limit ADC trigger  
11101 = PWM Generator 7 current-limit ADC trigger  
11100 = PWM Generator 6 current-limit ADC trigger  
11011 = PWM Generator 5 current-limit ADC trigger  
11010 = PWM Generator 4 current-limit ADC trigger  
11001 = PWM Generator 3 current-limit ADC trigger  
11000 = PWM Generator 2 current-limit ADC trigger  
10111 = PWM Generator 1 current-limit ADC trigger  
10110 = PWM Generator 9 secondary trigger selected  
10101 = PWM Generator 8 secondary trigger selected  
10100 = PWM Generator 7 secondary trigger selected  
10011 = PWM Generator 6 secondary trigger selected  
10010 = PWM Generator 5 secondary trigger selected  
10001 = PWM Generator 4 secondary trigger selected  
10000 = PWM Generator 3 secondary trigger selected  
01111 = PWM Generator 2 secondary trigger selected  
01110 = PWM Generator 1 secondary trigger selected  
01101 = PWM secondary special event trigger selected  
01100 = Timer1 period match  
01011 = PWM Generator 8 primary trigger selected  
01010 = PWM Generator 7 primary trigger selected  
01001 = PWM Generator 6 primary trigger selected  
01000 = PWM Generator 5 primary trigger selected  
00111 = PWM Generator 4 primary trigger selected  
00110 = PWM Generator 3 primary trigger selected  
00101 = PWM Generator 2 primary trigger selected  
00100 = PWM Generator 1 primary trigger selected  
00011 = PWM Special Event Trigger selected  
00010 = Global software trigger selected  
00001 = Individual software trigger selected  
00000 = No conversion enabled
- bit 7      **IRQEN8**: Interrupt Request Enable 8 bit  
1 = Enable IRQ generation when requested conversion of channels AN17 and AN16 is completed  
0 = IRQ is not generated
- bit 6      **PEND8**: Pending Conversion Status 8 bit  
1 = Conversion of channels AN17 and AN16 is pending; set when selected trigger is asserted  
0 = Conversion is complete
- bit 5      **SWTRG8**: Software Trigger 8 bit  
1 = Start conversion of AN17 and AN16 (if selected by TRGSRC bits)<sup>(1)</sup>  
This bit is automatically cleared by hardware when the PEND8 bit is set.  
0 = Conversion is not started

**Note 1:** The trigger source must be set as a global software trigger prior to setting this bit to '1'. If other conversions are in progress, the conversion is performed when the conversion resources are available.

## REGISTER 22-10: ADCPC4: ANALOG-TO-DIGITAL CONVERT PAIR CONTROL REGISTER 4 (CONTINUED)

bit 4-0      **TRGSRC8<4:0>**: Trigger 8 Source Selection bits  
Selects trigger source for conversion of analog channels AN17 and AN16.

- 11111 = Timer2 period match
- 11110 = PWM Generator 8 current-limit ADC trigger
- 11101 = PWM Generator 7 current-limit ADC trigger
- 11100 = PWM Generator 6 current-limit ADC trigger
- 11011 = PWM Generator 5 current-limit ADC trigger
- 11010 = PWM Generator 4 current-limit ADC trigger
- 11001 = PWM Generator 3 current-limit ADC trigger
- 11000 = PWM Generator 2 current-limit ADC trigger
- 10111 = PWM Generator 1 current-limit ADC trigger
- 10110 = PWM Generator 9 secondary trigger selected
- 10101 = PWM Generator 8 secondary trigger selected
- 10100 = PWM Generator 7 secondary trigger selected
- 10011 = PWM Generator 6 secondary trigger selected
- 10010 = PWM Generator 5 secondary trigger selected
- 10001 = PWM Generator 4 secondary trigger selected
- 10000 = PWM Generator 3 secondary trigger selected
- 01111 = PWM Generator 2 secondary trigger selected
- 01110 = PWM Generator 1 secondary trigger selected
- 01101 = PWM secondary special event trigger selected
- 01100 = Timer1 period match
- 01011 = PWM Generator 8 primary trigger selected
- 01010 = PWM Generator 7 primary trigger selected
- 01001 = PWM Generator 6 primary trigger selected
- 01000 = PWM Generator 5 primary trigger selected
- 00111 = PWM Generator 4 primary trigger selected
- 00110 = PWM Generator 3 primary trigger selected
- 00101 = PWM Generator 2 primary trigger selected
- 00100 = PWM Generator 1 primary trigger selected
- 00011 = PWM Special Event Trigger selected
- 00010 = Global software trigger selected
- 00001 = Individual software trigger selected
- 00000 = No conversion enabled

**Note 1:** The trigger source must be set as a global software trigger prior to setting this bit to '1'. If other conversions are in progress, the conversion is performed when the conversion resources are available.

## REGISTER 22-11: ADCPC5: ANALOG-TO-DIGITAL CONVERT PAIR CONTROL REGISTER 5

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IRQEN11	PEND11	SWTRG11	TRGSRC11<4:0>				
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
IRQEN10	PEND10	SWTRG10	TRGSRC10<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      **IRQEN11:** Interrupt Request Enable 11 bit  
 1 = Enable IRQ generation when requested conversion of channels AN23 and AN22 is completed  
 0 = IRQ is not generated
- bit 14      **PEND11:** Pending Conversion Status 11 bit  
 1 = Conversion of channels AN23 and AN22 is pending; set when selected trigger is asserted  
 0 = Conversion is complete
- bit 13      **SWTRG11:** Software Trigger 11 bit  
 1 = Start conversion of AN23 and AN22 (if selected in TRGSRC bits)<sup>(1)</sup>. This bit is automatically cleared by hardware when the PEND11 bit is set.  
 0 = Conversion is not started

**Note 1:** The trigger source must be set as a global software trigger prior to setting this bit to '1'. If other conversions are in progress, the conversion is performed when the conversion resources are available.

## REGISTER 22-11: ADCPC5: ANALOG-TO-DIGITAL CONVERT PAIR CONTROL REGISTER 5 (CONTINUED)

- bit 12-8      **TRGSRC11<4:0>**: Trigger 11 Source Selection bits  
Selects trigger source for conversion of analog channels AN23 and AN22.  
11111 = Timer2 period match  
11110 = PWM Generator 8 current-limit ADC trigger  
11101 = PWM Generator 7 current-limit ADC trigger  
11100 = PWM Generator 6 current-limit ADC trigger  
11011 = PWM Generator 5 current-limit ADC trigger  
11010 = PWM Generator 4 current-limit ADC trigger  
11001 = PWM Generator 3 current-limit ADC trigger  
11000 = PWM Generator 2 current-limit ADC trigger  
10111 = PWM Generator 1 current-limit ADC trigger  
10110 = PWM Generator 9 secondary trigger selected  
10101 = PWM Generator 8 secondary trigger selected  
10100 = PWM Generator 7 secondary trigger selected  
10011 = PWM Generator 6 secondary trigger selected  
10010 = PWM Generator 5 secondary trigger selected  
10001 = PWM Generator 4 secondary trigger selected  
10000 = PWM Generator 3 secondary trigger selected  
01111 = PWM Generator 2 secondary trigger selected  
01110 = PWM Generator 1 secondary trigger selected  
01101 = PWM secondary special event trigger selected  
01100 = Timer1 period match  
01011 = PWM Generator 8 primary trigger selected  
01010 = PWM Generator 7 primary trigger selected  
01001 = PWM Generator 6 primary trigger selected  
01000 = PWM Generator 5 primary trigger selected  
00111 = PWM Generator 4 primary trigger selected  
00110 = PWM Generator 3 primary trigger selected  
00101 = PWM Generator 2 primary trigger selected  
00100 = PWM Generator 1 primary trigger selected  
00011 = PWM Special Event Trigger selected  
00010 = Global software trigger selected  
00001 = Individual software trigger selected  
00000 = No conversion enabled
- bit 7      **IRQEN10**: Interrupt Request Enable 10 bit  
1 = Enable IRQ generation when requested conversion of channels AN21 and AN20 is completed  
0 = IRQ is not generated
- bit 6      **PEND10**: Pending Conversion Status 10 bit  
1 = Conversion of channels AN21 and AN20 is pending; set when selected trigger is asserted  
0 = Conversion is complete
- bit 5      **SWTRG10**: Software Trigger 10 bit  
1 = Start conversion of AN21 and AN20 (if selected by TRGSRC bits)<sup>(1)</sup>. This bit is automatically cleared by hardware when the PEND10 bit is set.  
0 = Conversion is not started

**Note 1:** The trigger source must be set as a global software trigger prior to setting this bit to '1'. If other conversions are in progress, the conversion is performed when the conversion resources are available.

## REGISTER 22-11: ADCPC5: ANALOG-TO-DIGITAL CONVERT PAIR CONTROL REGISTER 5 (CONTINUED)

bit 4-0      **TRGSRC10<4:0>**: Trigger 10 Source Selection bits  
Selects trigger source for conversion of analog channels AN21 and AN20.

- 11111 = Timer2 period match
- 11110 = PWM Generator 8 current-limit ADC trigger
- 11101 = PWM Generator 7 current-limit ADC trigger
- 11100 = PWM Generator 6 current-limit ADC trigger
- 11011 = PWM Generator 5 current-limit ADC trigger
- 11010 = PWM Generator 4 current-limit ADC trigger
- 11001 = PWM Generator 3 current-limit ADC trigger
- 11000 = PWM Generator 2 current-limit ADC trigger
- 10111 = PWM Generator 1 current-limit ADC trigger
- 10110 = PWM Generator 9 secondary trigger selected
- 10101 = PWM Generator 8 secondary trigger selected
- 10100 = PWM Generator 7 secondary trigger selected
- 10011 = PWM Generator 6 secondary trigger selected
- 10010 = PWM Generator 5 secondary trigger selected
- 10001 = PWM Generator 4 secondary trigger selected
- 10000 = PWM Generator 3 secondary trigger selected
- 01111 = PWM Generator 2 secondary trigger selected
- 01110 = PWM Generator 1 secondary trigger selected
- 01101 = PWM secondary special event trigger selected
- 01100 = Timer1 period match
- 01011 = PWM Generator 8 primary trigger selected
- 01010 = PWM Generator 7 primary trigger selected
- 01001 = PWM Generator 6 primary trigger selected
- 01000 = PWM Generator 5 primary trigger selected
- 00111 = PWM Generator 4 primary trigger selected
- 00110 = PWM Generator 3 primary trigger selected
- 00101 = PWM Generator 2 primary trigger selected
- 00100 = PWM Generator 1 primary trigger selected
- 00011 = PWM Special Event Trigger selected
- 00010 = Global software trigger selected
- 00001 = Individual software trigger selected
- 00000 = No conversion enabled

**Note 1:** The trigger source must be set as a global software trigger prior to setting this bit to '1'. If other conversions are in progress, the conversion is performed when the conversion resources are available.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 22-12: ADCPC6: ANALOG-TO-DIGITAL CONVERT PAIR CONTROL REGISTER 6<sup>(2)</sup>

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
IRQEN12	PEND12	SWTRG12	TRGSRC12<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 15-8      **Unimplemented:** Read as '0'
- bit 7      **IRQEN12:** Interrupt Request Enable 12 bit  
1 = Enable IRQ generation when requested conversion of channels AN25 and AN24 is completed  
0 = IRQ is not generated
- bit 6      **PEND12:** Pending Conversion Status 12 bit  
1 = Conversion of channels AN25 and AN24 is pending; set when selected trigger is asserted  
0 = Conversion is complete
- bit 5      **SWTRG12:** Software Trigger 12 bit  
1 = Start conversion of AN25 (INTREF) and AN24 (EXTREF) if selected by TRGSRC bits<sup>(1)</sup>  
This bit is automatically cleared by hardware when the PEND12 bit is set.  
0 = Conversion is not started

- Note 1:** The trigger source must be set as a global software trigger prior to setting this bit to '1'. If other conversions are in progress, the conversion is performed when the conversion resources are available.
- 2:** This register is not available on dsPIC33FJ32GS406 and dsPIC33FJ64GS406 devices.

## REGISTER 22-12: ADCPC6: ANALOG-TO-DIGITAL CONVERT PAIR CONTROL REGISTER 6<sup>(2)</sup>

bit 4-0      **TRGSRC12<4:0>**: Trigger 12 Source Selection bits  
Selects trigger source for conversion of analog channels AN25 and AN24.

- 11111 = Timer2 period match
- 11110 = PWM Generator 8 current-limit ADC trigger
- 11101 = PWM Generator 7 current-limit ADC trigger
- 11100 = PWM Generator 6 current-limit ADC trigger
- 11011 = PWM Generator 5 current-limit ADC trigger
- 11010 = PWM Generator 4 current-limit ADC trigger
- 11001 = PWM Generator 3 current-limit ADC trigger
- 11000 = PWM Generator 2 current-limit ADC trigger
- 10111 = PWM Generator 1 current-limit ADC trigger
- 10110 = PWM Generator 9 secondary trigger selected
- 10101 = PWM Generator 8 secondary trigger selected
- 10100 = PWM Generator 7 secondary trigger selected
- 10011 = PWM Generator 6 secondary trigger selected
- 10010 = PWM Generator 5 secondary trigger selected
- 10001 = PWM Generator 4 secondary trigger selected
- 10000 = PWM Generator 3 secondary trigger selected
- 01111 = PWM Generator 2 secondary trigger selected
- 01110 = PWM Generator 1 secondary trigger selected
- 01101 = PWM secondary special event trigger selected
- 01100 = Timer1 period match
- 01011 = PWM Generator 8 primary trigger selected
- 01010 = PWM Generator 7 primary trigger selected
- 01001 = PWM Generator 6 primary trigger selected
- 01000 = PWM Generator 5 primary trigger selected
- 00111 = PWM Generator 4 primary trigger selected
- 00110 = PWM Generator 3 primary trigger selected
- 00101 = PWM Generator 2 primary trigger selected
- 00100 = PWM Generator 1 primary trigger selected
- 00011 = PWM Special Event Trigger selected
- 00010 = Global software trigger selected
- 00001 = Individual software trigger selected
- 00000 = No conversion enabled

- Note 1:** The trigger source must be set as a global software trigger prior to setting this bit to '1'. If other conversions are in progress, the conversion is performed when the conversion resources are available.
- 2:** This register is not available on dsPIC33FJ32GS406 and dsPIC33FJ64GS406 devices.

NOTES:



## 23.0 HIGH-SPEED ANALOG COMPARATOR

**Note 1:** This data sheet summarizes the features of the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 families of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 45. “High-Speed Analog Comparator”** (DS70296) in the “dsPIC33F/PIC24H Family Reference Manual”, which is available from the Microchip web site ([www.microchip.com](http://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 dsPIC33F SMPS Comparator module monitors current and/or voltage transients that may be too fast for the CPU and ADC to capture.

### 23.1 Features Overview

The SMPS comparator module offers the following major features:

- 16 selectable comparator inputs
- Up to four analog comparators
- 10-bit DAC for each analog comparator

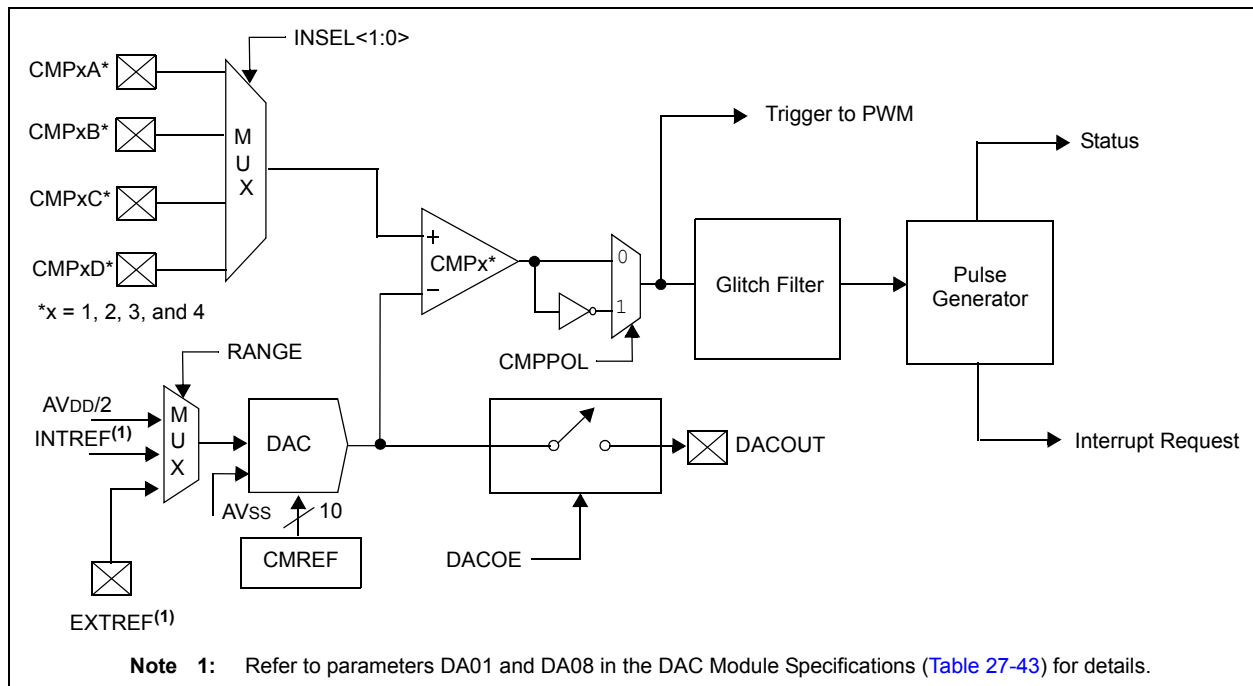
- Programmable output polarity
- Interrupt generation capability
- DACOUT pin to provide DAC output
- DAC has three ranges of operation:
  - $AV_{DD}/2$
  - Internal Reference (INTREF)
  - External Reference (EXTREF)
- ADC sample and convert trigger capability
- Disable capability reduces power consumption
- Functional support for PWM module:
  - PWM duty cycle control
  - PWM period control
  - PWM Fault detect

### 23.2 Module Description

Figure 23-1 shows a functional block diagram of one analog comparator from the SMPS comparator module. The analog comparator provides high-speed operation with a typical delay of 20 ns. The comparator has a typical offset voltage of  $\pm 5$  mV. The negative input of the comparator is always connected to the DAC circuit. The positive input of the comparator is connected to an analog multiplexer that selects the desired source pin.

The analog comparator input pins are typically shared with pins used by the Analog-to-Digital Converter (ADC) module. Both the comparator and the ADC can use the same pins at the same time. This capability enables a user to measure an input voltage with the ADC and detect voltage transients with the comparator.

**FIGURE 23-1: HIGH-SPEED ANALOG COMPARATOR MODULE BLOCK DIAGRAM**



## 23.3 Module Applications

This module provides a means for the SMPS dsPIC DSC devices to monitor voltage and currents in a power conversion application. The ability to detect transient conditions and stimulate the dsPIC DSC processor and/or peripherals, without requiring the processor and ADC to constantly monitor voltages or currents, frees the dsPIC DSC to perform other tasks.

The comparator module has a high-speed comparator and an associated 10-bit DAC that provides a programmable reference voltage to the inverting input of the comparator. The polarity of the comparator output is user-programmable. The output of the module can be used in the following modes:

- Generate an Interrupt
- Trigger an ADC Sample and Convert Process
- Truncate the PWM Signal (current limit)
- Truncate the PWM Period (current minimum)
- Disable the PWM Outputs (Fault latch)

The output of the comparator module may be used in multiple modes at the same time, such as: 1) generate an interrupt, 2) have the ADC take a sample and convert it, and 3) truncate the PWM output in response to a voltage being detected beyond its expected value.

The comparator module can also be used to wake-up the system from Sleep or Idle mode when the analog input voltage exceeds the programmed threshold voltage.

## 23.4 DAC

The range of the DAC is controlled via an analog multiplexer that selects either  $AV_{DD}/2$ , an internal reference source, INTREF, or an external reference source, EXTREF. The full range of the DAC ( $AV_{DD}/2$ ) will typically be used when the chosen input source pin is shared with the ADC. The reduced range option (INTREF) will likely be used when monitoring current levels using a current sense resistor. Usually, the measured voltages in such applications are small ( $<1.25V$ ); therefore the option of using a reduced reference range for the comparator extends the available DAC resolution in these applications. The use of an external reference enables the user to connect to a reference that better suits their application.

DACOUT, shown in [Figure 23-1](#), can only be associated with a single comparator at a given time.

**Note:** It should be ensured in software that multiple DACOE bits are not set. The output on the DACOUT pin will be indeterminate if multiple comparators enable the DAC output.

## 23.5 Interaction with I/O Buffers

If the comparator module is enabled and a pin has been selected as the source for the comparator, then the chosen I/O pad must disable the digital input buffer associated with the pad to prevent excessive currents in the digital buffer due to analog input voltages.

## 23.6 Digital Logic

The CMPCONx register (see [Register 23-1](#)) provides the control logic that configures the comparator module. The digital logic provides a glitch filter for the comparator output to mask transient signals in less than two instruction cycles. In Sleep or Idle mode, the glitch filter is bypassed to enable an asynchronous path from the comparator to the interrupt controller. This asynchronous path can be used to wake-up the processor from Sleep or Idle mode.

The comparator can be disabled while in Idle mode if the CMPSIDL bit is set. If a device has multiple comparators, if any CMPSIDL bit is set, then the entire group of comparators will be disabled while in Idle mode. This behavior reduces complexity in the design of the clock control logic for this module.

The digital logic also provides a one  $T_{CY}$  width pulse generator for triggering the ADC and generating interrupt requests.

The CMPDACx (see [Register 23-2](#)) register provides the digital input value to the reference DAC.

If the module is disabled, the DAC and comparator are disabled to reduce power consumption.

## 23.7 Comparator Input Range

The comparator has a limitation for the input Common Mode Range (CMR) of  $(AV_{DD} - 1.5V)$ , typical. This means that both inputs should not exceed this range. As long as one of the inputs is within the Common Mode Range, the comparator output will be correct. However, any input exceeding the CMR limitation will cause the comparator input to be saturated.

If both inputs exceed the CMR, the comparator output will be indeterminate.

## 23.8 DAC Output Range

The DAC has a limitation for the maximum reference voltage input of  $(AV_{DD} - 1.6)$  volts. An external reference voltage input should not exceed this value or the reference DAC output will become indeterminate.

## 23.9 Comparator Registers

The comparator module is controlled by the following registers:

- [CMPCONx: Comparator Control Register](#)
- [CMPDACx: Comparator DAC Control Register](#)

## REGISTER 23-1: CMPCONx: COMPARATOR CONTROL REGISTER

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0
CMPON	—	CMPSIDL	—	—	—	—	DACOE
bit 15							bit 8

R/W-0	R/W-0	R/W-0	U-0	R/W-0	U-0	R/W-0	R/W-0
INSEL<1:0>		EXTREF	—	CMPSTAT	—	CMPPOL	RANGE
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      **CMPON:** Comparator Operating Mode bit
  - 1 = Comparator module is enabled
  - 0 = Comparator module is disabled (reduces power consumption)
- bit 14      **Unimplemented:** Read as '0'
- bit 13      **CMPSIDL:** Stop in Idle Mode bit
  - 1 = Discontinue module operation when device enters Idle mode.
  - 0 = Continue module operation in Idle mode

If a device has multiple comparators, any CMPSIDL bit set to '1' disables **ALL** comparators while in Idle mode.
- bit 12-9    **Reserved:** Read as '0'
- bit 8        **DACOE:** DAC Output Enable
  - 1 = DAC analog voltage is output to DACOUT pin<sup>(1)</sup>
  - 0 = DAC analog voltage is not connected to DACOUT pin
- bit 7-6     **INSEL<1:0>:** Input Source Select for Comparator bits
  - 11 = Select CMPxD input pin
  - 10 = Select CMPxC input pin
  - 01 = Select CMPxB input pin
  - 00 = Select CMPxA input pin
- bit 5        **EXTREF:** Enable External Reference bit
  - 1 = External source provides reference to DAC (maximum DAC voltage determined by external voltage source)
  - 0 = Internal reference sources provide reference to DAC (maximum DAC voltage determined by RANGE bit setting)
- bit 4        **Reserved:** Read as '0'
- bit 3        **CMPSTAT:** Current State of Comparator Output Including CMPPOL Selection bit
- bit 2        **Reserved:** Read as '0'
- bit 1        **CMPPOL:** Comparator Output Polarity Control bit
  - 1 = Output is inverted
  - 0 = Output is non-inverted
- bit 0        **RANGE:** Selects DAC Output Voltage Range bit
  - 1 = High Range: Max DAC Value = AVDD/2, 1.65V at 3.3V AVDD
  - 0 = Low Range: Max DAC Value = INTREF

**Note 1:** DACOUT can be associated only with a single comparator at any given time. The software must ensure that multiple comparators do not enable the DAC output by setting their respective DACOE bit.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## REGISTER 23-2: CMPDACx: COMPARATOR DAC CONTROL REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	CMREF<9:8>	
bit 15						bit 8	

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CMREF<7:0>							
bit 7						bit 0	

### Legend:

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

bit 15-10      **Reserved:** Read as '0'

bit 9-0      **CMREF<9:0>:** Comparator Reference Voltage Select bits

1111111111 = (CMREF \* INTREF/1024) or (CMREF \* (AVDD/2)/1024) volts depending on RANGE bit or (CMREF \* EXTREF/1024) if EXTREF is set

- 
- 
- 

0000000000 = 0.0 volts

## 24.0 SPECIAL FEATURES

**Note 1:** This data sheet summarizes the features of the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 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”. Please see the Microchip web site ([www.microchip.com](http://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.

The dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 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
- Brown-out Reset (BOR)

## 24.1 Configuration Bits

The dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 devices provide non-volatile memory implementations for device Configuration bits. Refer to **Section 25. “Device Configuration”** (DS70194) in 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 individual Configuration bit descriptions for the Configuration registers are shown in [Table 24-2](#).

Note that address, 0xF80000, is beyond the user program memory space. It belongs to the configuration memory space (0x800000-0xFFFFF), which can only be accessed using table reads and table writes.

To prevent inadvertent configuration changes during code execution, all programmable Configuration bits are write-once. After a bit is initially programmed during a power cycle, it cannot be written again. Changing a device configuration requires that power to the device be cycled.

The device Configuration register map is shown in [Table 24-1](#).

**TABLE 24-1: DEVICE CONFIGURATION REGISTER MAP**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0xF80000	FBS	—	—	—	—	BSS<2:0>			BWRP
0xF80002	RESERVED	—	—	—	—	—	—	—	—
0xF80004	FGS	—	—	—	—	—	GSS<1:0>		GWRP
0xF80006	FOSCSEL	IESO	—	—	—		FNOSC<2:0>		
0xF80008	FOSC	FCKSM<1:0>		—	—	—	OSCIOFNC	POSCMD<1:0>	
0xF8000A	FWDT	FWDTEN	WINDIS	—	WDTPRE	WDTPOST<3:0>			
0xF8000C	FPOR	—	ALTQIO	ALTSS1	—	—	FPWRT<2:0>		
0xF8000E	FICD	Reserved <sup>(1)</sup>	Reserved <sup>(1)</sup>	JTAGEN	—	—	—	ICS<1:0>	
0xF80010	FCMP	—	—	CMPPOL1 <sup>(2)</sup>	HYST1<1:0> <sup>(2)</sup>		CMPPOL0 <sup>(2)</sup>	HYST0<1:0> <sup>(2)</sup>	

**Legend:** — = unimplemented bit, read as ‘0’.

**Note 1:** These bits are reserved for use by development tools and must be programmed as ‘1’.

**2:** These bits are reserved on dsPIC33FJXXXGS406 devices and always read as ‘1’.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

**TABLE 24-2: dsPIC33F CONFIGURATION BITS DESCRIPTION**

Bit Field	Register	RTSP Effect	Description
BWRP	FBS	Immediate	Boot Segment Program Flash Write Protection bit 1 = Boot segment can 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 256 instruction words (except interrupt vectors) 110 = Standard security; boot program Flash segment ends at 0x0003FE 010 = High security; boot program Flash segment ends at 0x0003FE Boot space is 768 instruction words (except interrupt vectors) 101 = Standard security; boot program Flash segment ends at 0x0007FE 001 = High security; boot program Flash segment ends at 0x0007FE Boot space is 1792 instruction words (except interrupt vectors) 100 = Standard security; boot program Flash segment ends at 0x000FFE 000 = High security; boot program Flash segment ends at 0x000FFE
GSS<1:0>	FGS	Immediate	General Segment Code-Protect bits 11 = User program memory is not code-protected 10 = Standard security 0x = High security
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

**TABLE 24-2: dsPIC33F CONFIGURATION BITS DESCRIPTION (CONTINUED)**

Bit Field	Register	RTSP Effect	Description
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
WDTPRE	FWDT	Immediate	Watchdog Timer Prescaler bit 1 = 1:128 0 = 1:32
WDTPOST<3:0>	FWDT	Immediate	Watchdog Timer Postscaler bits 1111 = 1:32,768 1110 = 1:16,384 • • • 0001 = 1:2 0000 = 1:1
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 is enabled 0 = JTAG is disabled
ICS<1:0>	FICD	Immediate	ICD Communication Channel Select Enable bits 11 = Communicate on PGEC1 and PGED1 10 = Communicate on PGEC2 and PGED2 01 = Communicate on PGEC3 and PGED3 00 = Reserved, do not use.
ALTQIO	FPOR	Immediate	Enable Alternate QE1 pin bit 1 = QEA1, QEB1 and INDX1 are selected as inputs to QE1 0 = AQEA1, AQEB1 and AINDX1 are selected as inputs to QE1
ALTSS1	FPOR	Immediate	Enable Alternate SS1 pin bit 1 = $\overline{ASS1}$ is selected as the I/O pin for SPI1 0 = SS1 is selected as the I/O pin for SPI1
CMPPOL0	FCMP	Immediate	Comparator Hysteresis Polarity (for even numbered comparators) 1 = Hysteresis is applied to falling edge 0 = Hysteresis is applied to rising edge
HYST0<1:0>	FCMP	Immediate	Comparator Hysteresis Select 11 = 45 mV Hysteresis 10 = 30 mV Hysteresis 01 = 15 mV Hysteresis 00 = No Hysteresis

## dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

---

**TABLE 24-2: dsPIC33F CONFIGURATION BITS DESCRIPTION (CONTINUED)**

Bit Field	Register	RTSP Effect	Description
CMPPOL1	FCMP	Immediate	Comparator Hysteresis Polarity (for odd numbered comparators) 1 = Hysteresis is applied to falling edge 0 = Hysteresis is applied to rising edge
HYST1<1:0>	FCMP	Immediate	Comparator Hysteresis Select 11 = 45 mV Hysteresis 10 = 30 mV Hysteresis 01 = 15 mV Hysteresis 00 = No Hysteresis



## 24.2 On-Chip Voltage Regulator

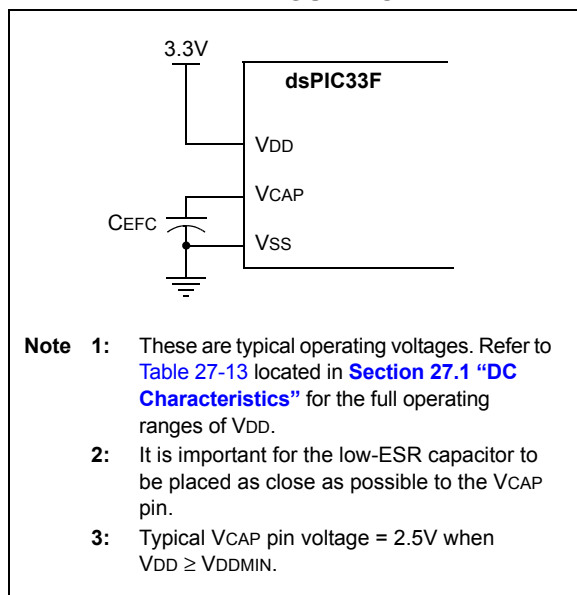
The dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 devices power their core digital logic at a nominal 2.5V. This can create a conflict for designs that are required to operate at a higher typical voltage, such as 3.3V. To simplify system design, all devices in the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 families 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. When the regulator is enabled, a low-ESR (less than 5 ohms) capacitor (such as tantalum or ceramic) must be connected to the VCAP pin (Figure 24-1). This helps to maintain the stability of the regulator. The recommended value for the filter capacitor is provided in Table 27-13 located in Section 27.1 “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 24-1: CONNECTIONS FOR THE ON-CHIP VOLTAGE REGULATOR<sup>(1,2,3)</sup>**



## 24.3 BOR: Brown-out Reset

The Brown-out Reset (BOR) module is based on an internal voltage reference circuit. 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 (for example, 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 generates a Reset pulse, which resets the device. The BOR selects the clock source, based on the device Configuration bit values (FNOSC<2:0> and POSCMD<1:0>).

If an oscillator mode is selected, the BOR activates the Oscillator Start-up Timer (OST). The system clock is held until OST expires. If the PLL is used, the clock is held until the LOCK bit (OSCCON<5>) is ‘1’.

Concurrently, the PWRT time-out (TPWRT) is 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>) is set to indicate that a BOR has occurred. The BOR circuit continues to operate while in Sleep or Idle modes and resets the device should VDD fall below the BOR threshold voltage.

## 24.4 Watchdog Timer (WDT)

For dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 devices, the WDT is driven by the LPRC oscillator. When the WDT is enabled, the clock source is also enabled.

### 24.4.1 PRESCALER/POSTSCALER

The nominal WDT clock source from LPRC is 32.767 kHz. This feeds a prescaler that can be configured for either 5-bit (divide-by-32) or 7-bit (divide-by-128) operation. The prescaler is set by the WDTPRE Configuration bit. With a 32.767 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 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 `PWRSVAV` 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

**Note:** The `CLRWDT` and `PWRSVAV` instructions clear the prescaler and postscaler counts when executed.

## 24.4.2 SLEEP AND IDLE MODES

If the WDT is enabled, it will continue to run during Sleep or Idle modes. When the WDT time-out occurs, the WDT will wake the device and code execution will continue from where the `PWRSVAV` instruction was executed. The corresponding `SLEEP` or `IDLE` bits (`RCON<3:2>`) will need to be cleared in software after the device wakes up.

## 24.4.3 ENABLING WDT

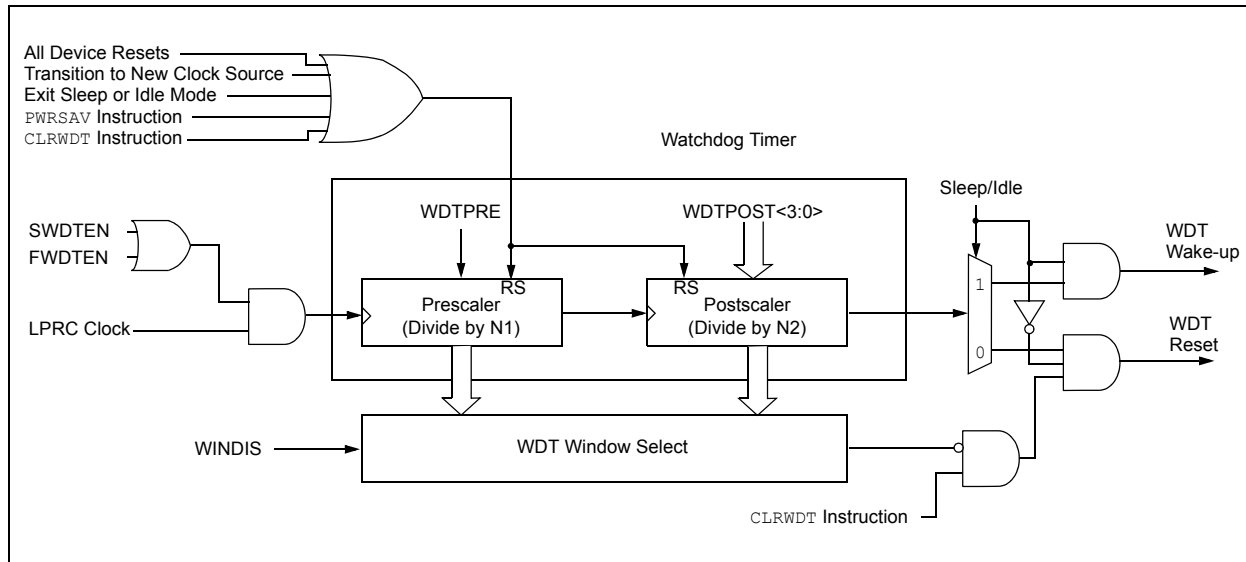
The WDT is enabled or disabled by the `FWDTEN` Configuration bit in the `FWDT` Configuration register. When the `FWDTEN` Configuration bit is set, the WDT is always enabled.

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 application 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.

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.

**FIGURE 24-2: WDT BLOCK DIAGRAM**



## 24.5 JTAG Interface

dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 devices implement a JTAG interface, which supports boundary scan device testing, as well as in-circuit programming. Detailed information on this interface will be provided in future revisions of the document.

## 24.6 In-Circuit Serial Programming

dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 family digital signal controllers can be serially programmed while in the end application circuit. This is done with two lines for clock and data and three other lines for power, ground and the programming sequence. Serial programming allows customers to manufacture boards with unprogrammed devices and then program the digital signal controller just before shipping the product. Serial programming also allows the most recent firmware or a custom firmware to be programmed. Refer to the “*dsPIC33F/PIC24H Flash Programming Specification*” (DS70152) for details about In-Circuit Serial Programming (ICSP).

Any of the three pairs of programming clock/data pins can be used:

- PGEC1 and PGED1
- PGEC2 and PGED2
- PGEC3 and PGED3

## 24.7 In-Circuit Debugger

When MPLAB ICD 3 is selected as a debugger, the in-circuit debugging functionality is enabled. This function allows simple debugging functions when used with MPLAB IDE. Debugging functionality is controlled through the EMUCx (Emulation/Debug Clock) and EMUDx (Emulation/Debug Data) pin functions.

Any of the three pairs of debugging clock/data pins can 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, PGECx, PGEDx and the EMUDx/EMUCx 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.

## 24.8 Code Protection and CodeGuard™ Security

The dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 devices offer the intermediate 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 in collaborative system designs.

When coupled with software encryption libraries, CodeGuard™ Security can be used to securely update Flash even when multiple IPs reside on a single chip.

The code protection features are controlled by the Configuration registers: FBS and FGS.

Secure segment and RAM protection is not implemented in dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 devices.

**Note:** Refer to **Section 23. “CodeGuard™ Security”** (DS70199) for further information on usage, configuration and operation of CodeGuard Security.

**TABLE 24-3: CODE FLASH SECURITY SEGMENT SIZES FOR 64K BYTE DEVICES**

BSS<2:0> = x11 0K		BSS<2:0> = x10 1K		BSS<2:0> = x01 4K		BSS<2:0> = x00 8K	
VS = 256 IW	000000h 0001FEh 000200h	VS = 256 IW	000000h 0001FEh 000200h	VS = 256 IW	000000h 0001FEh 000200h	VS = 256 IW	000000h 0001FEh 000200h
		BS = 768 IW	000200h 0007FEh 000800h	BS = 3840 IW	001FFEh 002000h	BS = 7936 IW	003FFEh 004000h
GS = 21760 IW	00ABFEh	GS = 20992 IW	00ABFEh	GS = 17920 IW	00ABFEh	GS = 13824 IW	00ABFEh

**TABLE 24-4: CODE FLASH SECURITY SEGMENT SIZES FOR 32K BYTE DEVICES**

BSS<2:0> = x11 0K		BSS<2:0> = x10 1K		BSS<2:0> = x01 4K		BSS<2:0> = x00 8K	
VS = 256 IW	000000h 0001FEh 000200h	VS = 256 IW	000000h 0001FEh 000200h	VS = 256 IW	000000h 0001FEh 000200h	VS = 256 IW	000000h 0001FEh 000200h
		BS = 768 IW	000200h 0007FEh 000800h	BS = 3840 IW	001FFEh 002000h	BS = 7936 IW	003FFEh 004000h 0057FEh
GS = 11008 IW	0057FEh	GS = 10240 IW	0057FEh	GS = 7168 IW	0057FEh	GS = 3072 IW	00ABFEh
	00ABFEh		00ABFEh		00ABFEh		00ABFEh

## 25.0 INSTRUCTION SET SUMMARY

**Note:** This data sheet summarizes the features of the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 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”. Please see the Microchip web site ([www.microchip.com](http://www.microchip.com)) for the latest “dsPIC33F/PIC24H Family Reference Manual” sections.

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 25-1 shows the general symbols used in describing the instructions.

The dsPIC33F instruction set summary in Table 25-2 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 be either 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 can use some of the following operands:

- A literal value to be loaded into a W register or file register (specified by ‘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 can 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 can 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 can use some of the following operands:

- A program memory address
- The mode of the table read and table write instructions

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

Most instructions are a single word. Certain double-word instructions are designed to provide all the required information 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 `NOB`.

The double-word instructions execute in two instruction cycles.

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 `NOB`. 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.

**Note:** For more details on the instruction set, refer to the "16-bit MCU and DSC Programmer's Reference Manual" (DS70157).

**TABLE 25-1: 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>	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 $\in \{W13, [W13]+ = 2\}$
bit4	4-bit bit selection field (used in word-addressed instructions) $\in \{0..15\}$
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 $\in \{0x0000..0x1FFF\}$
lit1	1-bit unsigned literal $\in \{0,1\}$
lit4	4-bit unsigned literal $\in \{0..15\}$
lit5	5-bit unsigned literal $\in \{0..31\}$
lit8	8-bit unsigned literal $\in \{0..255\}$
lit10	10-bit unsigned literal $\in \{0..255\}$ for Byte mode, $\{0:1023\}$ for Word mode
lit14	14-bit unsigned literal $\in \{0..16384\}$
lit16	16-bit unsigned literal $\in \{0..65535\}$
lit23	23-bit unsigned literal $\in \{0..8388608\}$ ; LSb must be '0'
None	Field does not require an entry, can 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 $\in \{-512..511\}$
Slit16	16-bit signed literal $\in \{-32768..32767\}$
Slit6	6-bit signed literal $\in \{-16..16\}$
Wb	Base W register $\in \{W0..W15\}$
Wd	Destination W register $\in \{Wd, [Wd], [Wd++] , [Wd--], [++Wd], [--Wd] \}$
Wdo	Destination W register $\in \{Wnd, [Wnd], [Wnd++] , [Wnd--], [++Wnd], [--Wnd], [Wnd+Wb] \}$
Wm,Wn	Dividend, Divisor Working register pair (Direct Addressing)

**TABLE 25-1: SYMBOLS USED IN OPCODE DESCRIPTIONS (CONTINUED)**

Field	Description
Wm*Wm	Multiplicand and Multiplier Working register pair for Square instructions $\in \{W4 * W4, W5 * W5, W6 * W6, W7 * W7\}$
Wm*Wn	Multiplicand and Multiplier Working register pair for DSP instructions $\in \{W4 * W5, W4 * W6, W4 * W7, W5 * W6, W5 * W7, W6 * W7\}$
Wn	One of 16 Working registers $\in \{W0..W15\}$
Wnd	One of 16 Destination Working registers $\in \{W0..W15\}$
Wns	One of 16 Source Working registers $\in \{W0..W15\}$
WREG	W0 (Working register used in file register instructions)
Ws	Source W register $\in \{Ws, [Ws], [Ws++] , [Ws--], [++Ws], [--Ws] \}$
Wso	Source W register $\in \{Wns, [Wns], [Wns++] , [Wns--], [++Wns], [--Wns], [Wns+Wb] \}$
Wx	X Data Space Prefetch Address register for DSP instructions $\in \{[W8] + = 6, [W8] + = 4, [W8] + = 2, [W8], [W8] - = 6, [W8] - = 4, [W8] - = 2, [W9] + = 6, [W9] + = 4, [W9] + = 2, [W9], [W9] - = 6, [W9] - = 4, [W9] - = 2, [W9 + W12], \text{none}\}$
Wxd	X Data Space Prefetch Destination register for DSP instructions $\in \{W4..W7\}$
Wy	Y Data Space Prefetch Address register for DSP instructions $\in \{[W10] + = 6, [W10] + = 4, [W10] + = 2, [W10], [W10] - = 6, [W10] - = 4, [W10] - = 2, [W11] + = 6, [W11] + = 4, [W11] + = 2, [W11], [W11] - = 6, [W11] - = 4, [W11] - = 2, [W11 + W12], \text{none}\}$
Wyd	Y Data Space Prefetch Destination register for DSP instructions $\in \{W4..W7\}$

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

**TABLE 25-2: INSTRUCTION SET OVERVIEW**

Base Instr #	Assembly Mnemonic	Assembly Syntax	Description	# of Words	# of Cycles	Status Flags Affected
1	ADD	ADD <i>Acc</i>	Add Accumulators	1	1	OA,OB,SA,SB
		ADD <i>f</i>	$f = f + \text{WREG}$	1	1	C,DC,N,OV,Z
		ADD <i>f, WREG</i>	$\text{WREG} = f + \text{WREG}$	1	1	C,DC,N,OV,Z
		ADD <i>#lit10, Wn</i>	$\text{Wd} = \text{lit10} + \text{Wd}$	1	1	C,DC,N,OV,Z
		ADD <i>Wb, Ws, Wd</i>	$\text{Wd} = \text{Wb} + \text{Ws}$	1	1	C,DC,N,OV,Z
		ADD <i>Wb, #lit5, Wd</i>	$\text{Wd} = \text{Wb} + \text{lit5}$	1	1	C,DC,N,OV,Z
		ADD <i>Wso, #Slit4, Acc</i>	16-Bit Signed Add to Accumulator	1	1	OA,OB,SA,SB
2	ADDC	ADDC <i>f</i>	$f = f + \text{WREG} + (C)$	1	1	C,DC,N,OV,Z
		ADDC <i>f, WREG</i>	$\text{WREG} = f + \text{WREG} + (C)$	1	1	C,DC,N,OV,Z
		ADDC <i>#lit10, Wn</i>	$\text{Wd} = \text{lit10} + \text{Wd} + (C)$	1	1	C,DC,N,OV,Z
		ADDC <i>Wb, Ws, Wd</i>	$\text{Wd} = \text{Wb} + \text{Ws} + (C)$	1	1	C,DC,N,OV,Z
		ADDC <i>Wb, #lit5, Wd</i>	$\text{Wd} = \text{Wb} + \text{lit5} + (C)$	1	1	C,DC,N,OV,Z
3	AND	AND <i>f</i>	$f = f \text{ .AND. } \text{WREG}$	1	1	N,Z
		AND <i>f, WREG</i>	$\text{WREG} = f \text{ .AND. } \text{WREG}$	1	1	N,Z
		AND <i>#lit10, Wn</i>	$\text{Wd} = \text{lit10} \text{ .AND. } \text{Wd}$	1	1	N,Z
		AND <i>Wb, Ws, Wd</i>	$\text{Wd} = \text{Wb} \text{ .AND. } \text{Ws}$	1	1	N,Z
		AND <i>Wb, #lit5, Wd</i>	$\text{Wd} = \text{Wb} \text{ .AND. } \text{lit5}$	1	1	N,Z
4	ASR	ASR <i>f</i>	$f = \text{Arithmetic Right Shift } f$	1	1	C,N,OV,Z
		ASR <i>f, WREG</i>	$\text{WREG} = \text{Arithmetic Right Shift } f$	1	1	C,N,OV,Z
		ASR <i>Ws, Wd</i>	$\text{Wd} = \text{Arithmetic Right Shift } \text{Ws}$	1	1	C,N,OV,Z
		ASR <i>Wb, Wns, Wnd</i>	$\text{Wnd} = \text{Arithmetic Right Shift } \text{Wb} \text{ by } \text{Wns}$	1	1	N,Z
		ASR <i>Wb, #lit5, Wnd</i>	$\text{Wnd} = \text{Arithmetic Right Shift } \text{Wb} \text{ by } \text{lit5}$	1	1	N,Z
5	BCLR	BCLR <i>f, #bit4</i>	Bit Clear <i>f</i>	1	1	None
		BCLR <i>Ws, #bit4</i>	Bit Clear <i>Ws</i>	1	1	None
6	BRA	BRA <i>C, Expr</i>	Branch if Carry	1	1 (2)	None
		BRA <i>GE, Expr</i>	Branch if Greater Than or Equal	1	1 (2)	None
		BRA <i>GEU, Expr</i>	Branch if Unsigned Greater Than or Equal	1	1 (2)	None
		BRA <i>GT, Expr</i>	Branch if Greater Than	1	1 (2)	None
		BRA <i>GTU, Expr</i>	Branch if Unsigned Greater Than	1	1 (2)	None
		BRA <i>LE, Expr</i>	Branch if Less Than or Equal	1	1 (2)	None
		BRA <i>LEU, Expr</i>	Branch if Unsigned Less Than or Equal	1	1 (2)	None
		BRA <i>LT, Expr</i>	Branch if Less Than	1	1 (2)	None
		BRA <i>LTU, Expr</i>	Branch if Unsigned Less Than	1	1 (2)	None
		BRA <i>N, Expr</i>	Branch if Negative	1	1 (2)	None
		BRA <i>NC, Expr</i>	Branch if Not Carry	1	1 (2)	None
		BRA <i>NN, Expr</i>	Branch if Not Negative	1	1 (2)	None
		BRA <i>NOV, Expr</i>	Branch if Not Overflow	1	1 (2)	None
		BRA <i>NZ, Expr</i>	Branch if Not Zero	1	1 (2)	None
		BRA <i>OA, Expr</i>	Branch if Accumulator A Overflow	1	1 (2)	None
		BRA <i>OB, Expr</i>	Branch if Accumulator B Overflow	1	1 (2)	None
		BRA <i>OV, Expr</i>	Branch if Overflow	1	1 (2)	None
		BRA <i>SA, Expr</i>	Branch if Accumulator A Saturated	1	1 (2)	None
		BRA <i>SB, Expr</i>	Branch if Accumulator B Saturated	1	1 (2)	None
		BRA <i>Expr</i>	Branch Unconditionally	1	2	None
BRA <i>Z, Expr</i>	Branch if Zero	1	1 (2)	None		
BRA <i>Wn</i>	Computed Branch	1	2	None		
7	BSET	BSET <i>f, #bit4</i>	Bit Set <i>f</i>	1	1	None
		BSET <i>Ws, #bit4</i>	Bit Set <i>Ws</i>	1	1	None
8	BSW	BSW.C <i>Ws, Wb</i>	Write C bit to <i>Ws</i> < <i>Wb</i> >	1	1	None
		BSW.Z <i>Ws, Wb</i>	Write Z bit to <i>Ws</i> < <i>Wb</i> >	1	1	None
9	BTG	BTG <i>f, #bit4</i>	Bit Toggle <i>f</i>	1	1	None
		BTG <i>Ws, #bit4</i>	Bit Toggle <i>Ws</i>	1	1	None



TABLE 25-2: INSTRUCTION SET OVERVIEW (CONTINUED)

Base Instr #	Assembly Mnemonic	Assembly Syntax	Description	# of Words	# of Cycles	Status Flags Affected
10	BTSC	BTSC <i>f</i> , #bit4	Bit Test <i>f</i> , Skip if Clear	1	1 (2 or 3)	None
		BTSC <i>Ws</i> , #bit4	Bit Test <i>Ws</i> , Skip if Clear	1	1 (2 or 3)	None
11	BTSS	BTSS <i>f</i> , #bit4	Bit Test <i>f</i> , Skip if Set	1	1 (2 or 3)	None
		BTSS <i>Ws</i> , #bit4	Bit Test <i>Ws</i> , Skip if Set	1	1 (2 or 3)	None
12	BTST	BTST <i>f</i> , #bit4	Bit Test <i>f</i>	1	1	Z
		BTST.C <i>Ws</i> , #bit4	Bit Test <i>Ws</i> to C	1	1	C
		BTST.Z <i>Ws</i> , #bit4	Bit Test <i>Ws</i> to Z	1	1	Z
		BTST.C <i>Ws</i> , <i>Wb</i>	Bit Test <i>Ws</i> < <i>Wb</i> > to C	1	1	C
		BTST.Z <i>Ws</i> , <i>Wb</i>	Bit Test <i>Ws</i> < <i>Wb</i> > to Z	1	1	Z
13	BTSTS	BTSTS <i>f</i> , #bit4	Bit Test then Set <i>f</i>	1	1	Z
		BTSTS.C <i>Ws</i> , #bit4	Bit Test <i>Ws</i> to C, then Set	1	1	C
		BTSTS.Z <i>Ws</i> , #bit4	Bit Test <i>Ws</i> to Z, then Set	1	1	Z
14	CALL	CALL <i>lit</i> 23	Call Subroutine	2	2	None
		CALL <i>Wn</i>	Call Indirect Subroutine	1	2	None
15	CLR	CLR <i>f</i>	<i>f</i> = 0x0000	1	1	None
		CLR WREG	WREG = 0x0000	1	1	None
		CLR <i>Ws</i>	<i>Ws</i> = 0x0000	1	1	None
		CLR <i>Acc</i> , <i>Wx</i> , <i>Wxd</i> , <i>Wy</i> , <i>Wyd</i> , <i>AWB</i>	Clear Accumulator	1	1	OA,OB,SA,SB
16	CLRWDT	CLRWDT	Clear Watchdog Timer	1	1	WDTO,Sleep
17	COM	COM <i>f</i>	<i>f</i> = $\bar{f}$	1	1	N,Z
		COM <i>f</i> , WREG	WREG = $\bar{f}$	1	1	N,Z
		COM <i>Ws</i> , <i>Wd</i>	<i>Wd</i> = $\bar{Ws}$	1	1	N,Z
18	CP	CP <i>f</i>	Compare <i>f</i> with WREG	1	1	C,DC,N,OV,Z
		CP <i>Wb</i> , #lit5	Compare <i>Wb</i> with lit5	1	1	C,DC,N,OV,Z
		CP <i>Wb</i> , <i>Ws</i>	Compare <i>Wb</i> with <i>Ws</i> ( <i>Wb</i> – <i>Ws</i> )	1	1	C,DC,N,OV,Z
19	CP0	CP0 <i>f</i>	Compare <i>f</i> with 0x0000	1	1	C,DC,N,OV,Z
		CP0 <i>Ws</i>	Compare <i>Ws</i> with 0x0000	1	1	C,DC,N,OV,Z
20	CPB	CPB <i>f</i>	Compare <i>f</i> with WREG, with Borrow	1	1	C,DC,N,OV,Z
		CPB <i>Wb</i> , #lit5	Compare <i>Wb</i> with lit5, with Borrow	1	1	C,DC,N,OV,Z
		CPB <i>Wb</i> , <i>Ws</i>	Compare <i>Wb</i> with <i>Ws</i> , with Borrow ( <i>Wb</i> – <i>Ws</i> – C)	1	1	C,DC,N,OV,Z
21	CPSEQ	CPSEQ <i>Wb</i> , <i>Wn</i>	Compare <i>Wb</i> with <i>Wn</i> , Skip if =	1	1 (2 or 3)	None
22	CPSGT	CPSGT <i>Wb</i> , <i>Wn</i>	Compare <i>Wb</i> with <i>Wn</i> , Skip if >	1	1 (2 or 3)	None
23	CPSLT	CPSLT <i>Wb</i> , <i>Wn</i>	Compare <i>Wb</i> with <i>Wn</i> , Skip if <	1	1 (2 or 3)	None
24	CPSNE	CPSNE <i>Wb</i> , <i>Wn</i>	Compare <i>Wb</i> with <i>Wn</i> , Skip if ≠	1	1 (2 or 3)	None
25	DAW	DAW <i>Wn</i>	<i>Wn</i> = Decimal Adjust <i>Wn</i>	1	1	C
26	DEC	DEC <i>f</i>	<i>f</i> = <i>f</i> – 1	1	1	C,DC,N,OV,Z
		DEC <i>f</i> , WREG	WREG = <i>f</i> – 1	1	1	C,DC,N,OV,Z
		DEC <i>Ws</i> , <i>Wd</i>	<i>Wd</i> = <i>Ws</i> – 1	1	1	C,DC,N,OV,Z
27	DEC2	DEC2 <i>f</i>	<i>f</i> = <i>f</i> – 2	1	1	C,DC,N,OV,Z
		DEC2 <i>f</i> , WREG	WREG = <i>f</i> – 2	1	1	C,DC,N,OV,Z
		DEC2 <i>Ws</i> , <i>Wd</i>	<i>Wd</i> = <i>Ws</i> – 2	1	1	C,DC,N,OV,Z
28	DISI	DISI #lit14	Disable Interrupts for <i>k</i> Instruction Cycles	1	1	None

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

**TABLE 25-2: INSTRUCTION SET OVERVIEW (CONTINUED)**

Base Instr #	Assembly Mnemonic	Assembly Syntax	Description	# of Words	# of Cycles	Status Flags Affected
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
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	C
36	FF1L	FF1L Ws, Wnd	Find First One from Left (MSb) Side	1	1	C
37	FF1R	FF1R Ws, Wnd	Find First One from Right (LSb) Side	1	1	C
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	N,Z
		MOV f, WREG	Move f to WREG	1	1	None
		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

TABLE 25-2: INSTRUCTION SET OVERVIEW (CONTINUED)

Base Instr #	Assembly Mnemonic	Assembly Syntax	Description	# of Words	# of Cycles	Status Flags Affected
48	MPY	MPY Wm*Wn, Acc, Wx, Wxd, Wy, Wyd	Multiply Wm by Wn to Accumulator	1	1	OA,OB,OAB, SA,SB,SAB
		MPY Wm*Wm, Acc, Wx, Wxd, Wy, Wyd	Square Wm to Accumulator	1	1	OA,OB,OAB, SA,SB,SAB
49	MPY.N	MPY.N Wm*Wn, Acc, 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
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 = $\bar{f} + 1$	1	1	C,DC,N,OV,Z
		NEG Ws, Wd	$Wd = \overline{Ws} + 1$	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	PWRSVAV	PWRSVAV #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

TABLE 25-2: INSTRUCTION SET OVERVIEW (CONTINUED)

Base Instr #	Assembly Mnemonic	Assembly Syntax	Description	# of Words	# of Cycles	Status Flags Affected
66	RRNC	RRNC <i>f</i>	<i>f</i> = Rotate Right (No Carry) <i>f</i>	1	1	N,Z
		RRNC <i>f</i> , WREG	WREG = Rotate Right (No Carry) <i>f</i>	1	1	N,Z
		RRNC <i>Ws</i> , <i>Wd</i>	<i>Wd</i> = Rotate Right (No Carry) <i>Ws</i>	1	1	N,Z
67	SAC	SAC <i>Acc</i> , # <i>Slit</i> 4, <i>Wdo</i>	Store Accumulator	1	1	None
		SAC.R <i>Acc</i> , # <i>Slit</i> 4, <i>Wdo</i>	Store Rounded Accumulator	1	1	None
68	SE	SE <i>Ws</i> , <i>Wnd</i>	<i>Wnd</i> = Sign-Extended <i>Ws</i>	1	1	C,N,Z
69	SETM	SETM <i>f</i>	<i>f</i> = 0xFFFF	1	1	None
		SETM WREG	WREG = 0xFFFF	1	1	None
		SETM <i>Ws</i>	<i>Ws</i> = 0xFFFF	1	1	None
70	SFTAC	SFTAC <i>Acc</i> , <i>Wn</i>	Arithmetic Shift Accumulator by ( <i>Wn</i> )	1	1	OA,OB,OAB,SA,SB,SAB
		SFTAC <i>Acc</i> , # <i>Slit</i> 6	Arithmetic Shift Accumulator by <i>Slit</i> 6	1	1	OA,OB,OAB,SA,SB,SAB
71	SL	SL <i>f</i>	<i>f</i> = Left Shift <i>f</i>	1	1	C,N,OV,Z
		SL <i>f</i> , WREG	WREG = Left Shift <i>f</i>	1	1	C,N,OV,Z
		SL <i>Ws</i> , <i>Wd</i>	<i>Wd</i> = Left Shift <i>Ws</i>	1	1	C,N,OV,Z
		SL <i>Wb</i> , <i>Wns</i> , <i>Wnd</i>	<i>Wnd</i> = Left Shift <i>Wb</i> by <i>Wns</i>	1	1	N,Z
		SL <i>Wb</i> , # <i>lit</i> 5, <i>Wnd</i>	<i>Wnd</i> = Left Shift <i>Wb</i> by <i>lit</i> 5	1	1	N,Z
72	SUB	SUB <i>Acc</i>	Subtract Accumulators	1	1	OA,OB,OAB,SA,SB,SAB
		SUB <i>f</i>	<i>f</i> = <i>f</i> – WREG	1	1	C,DC,N,OV,Z
		SUB <i>f</i> , WREG	WREG = <i>f</i> – WREG	1	1	C,DC,N,OV,Z
		SUB # <i>lit</i> 10, <i>Wn</i>	<i>Wn</i> = <i>Wn</i> – <i>lit</i> 10	1	1	C,DC,N,OV,Z
		SUB <i>Wb</i> , <i>Ws</i> , <i>Wd</i>	<i>Wd</i> = <i>Wb</i> – <i>Ws</i>	1	1	C,DC,N,OV,Z
		SUB <i>Wb</i> , # <i>lit</i> 5, <i>Wd</i>	<i>Wd</i> = <i>Wb</i> – <i>lit</i> 5	1	1	C,DC,N,OV,Z
73	SUBB	SUBB <i>f</i>	<i>f</i> = <i>f</i> – WREG – ( $\bar{C}$ )	1	1	C,DC,N,OV,Z
		SUBB <i>f</i> , WREG	WREG = <i>f</i> – WREG – ( $\bar{C}$ )	1	1	C,DC,N,OV,Z
		SUBB # <i>lit</i> 10, <i>Wn</i>	<i>Wn</i> = <i>Wn</i> – <i>lit</i> 10 – ( $\bar{C}$ )	1	1	C,DC,N,OV,Z
		SUBB <i>Wb</i> , <i>Ws</i> , <i>Wd</i>	<i>Wd</i> = <i>Wb</i> – <i>Ws</i> – ( $\bar{C}$ )	1	1	C,DC,N,OV,Z
		SUBB <i>Wb</i> , # <i>lit</i> 5, <i>Wd</i>	<i>Wd</i> = <i>Wb</i> – <i>lit</i> 5 – ( $\bar{C}$ )	1	1	C,DC,N,OV,Z
74	SUBR	SUBR <i>f</i>	<i>f</i> = WREG – <i>f</i>	1	1	C,DC,N,OV,Z
		SUBR <i>f</i> , WREG	WREG = WREG – <i>f</i>	1	1	C,DC,N,OV,Z
		SUBR <i>Wb</i> , <i>Ws</i> , <i>Wd</i>	<i>Wd</i> = <i>Ws</i> – <i>Wb</i>	1	1	C,DC,N,OV,Z
		SUBR <i>Wb</i> , # <i>lit</i> 5, <i>Wd</i>	<i>Wd</i> = <i>lit</i> 5 – <i>Wb</i>	1	1	C,DC,N,OV,Z
75	SUBBR	SUBBR <i>f</i>	<i>f</i> = WREG – <i>f</i> – ( $\bar{C}$ )	1	1	C,DC,N,OV,Z
		SUBBR <i>f</i> , WREG	WREG = WREG – <i>f</i> – ( $\bar{C}$ )	1	1	C,DC,N,OV,Z
		SUBBR <i>Wb</i> , <i>Ws</i> , <i>Wd</i>	<i>Wd</i> = <i>Ws</i> – <i>Wb</i> – ( $\bar{C}$ )	1	1	C,DC,N,OV,Z
		SUBBR <i>Wb</i> , # <i>lit</i> 5, <i>Wd</i>	<i>Wd</i> = <i>lit</i> 5 – <i>Wb</i> – ( $\bar{C}$ )	1	1	C,DC,N,OV,Z
76	SWAP	SWAP.b <i>Wn</i>	<i>Wn</i> = Nibble Swap <i>Wn</i>	1	1	None
		SWAP <i>Wn</i>	<i>Wn</i> = Byte Swap <i>Wn</i>	1	1	None
77	TBLRDH	TBLRDH <i>Ws</i> , <i>Wd</i>	Read Prog<23:16> to <i>Wd</i> <7:0>	1	2	None
78	TBLRDL	TBLRDL <i>Ws</i> , <i>Wd</i>	Read Prog<15:0> to <i>Wd</i>	1	2	None
79	TBLWTH	TBLWTH <i>Ws</i> , <i>Wd</i>	Write <i>Ws</i> <7:0> to Prog<23:16>	1	2	None
80	TBLWTL	TBLWTL <i>Ws</i> , <i>Wd</i>	Write <i>Ws</i> to Prog<15:0>	1	2	None
81	ULNK	ULNK	Unlink Frame Pointer	1	1	None
82	XOR	XOR <i>f</i>	<i>f</i> = <i>f</i> .XOR. WREG	1	1	N,Z
		XOR <i>f</i> , WREG	WREG = <i>f</i> .XOR. WREG	1	1	N,Z
		XOR # <i>lit</i> 10, <i>Wn</i>	<i>Wd</i> = <i>lit</i> 10 .XOR. <i>Wd</i>	1	1	N,Z
		XOR <i>Wb</i> , <i>Ws</i> , <i>Wd</i>	<i>Wd</i> = <i>Wb</i> .XOR. <i>Ws</i>	1	1	N,Z
		XOR <i>Wb</i> , # <i>lit</i> 5, <i>Wd</i>	<i>Wd</i> = <i>Wb</i> .XOR. <i>lit</i> 5	1	1	N,Z
83	ZE	ZE <i>Ws</i> , <i>Wnd</i>	<i>Wnd</i> = Zero-Extend <i>Ws</i>	1	1	C,Z,N

## 26.0 DEVELOPMENT SUPPORT

The PIC® microcontrollers and dsPIC® 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™ Object Linker/  
MPLIB™ 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

## 26.1 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.

## 26.2 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.

## 26.3 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, pre-processor, and one-step driver, and can run on multiple platforms.

## 26.4 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

## 26.5 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

## 26.6 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

## **26.7 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.

## **26.8 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® Flash MCUs and dsPIC® 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 in-circuit 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.

## **26.9 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.

## **26.10 PICkit 3 In-Circuit Debugger/Programmer and PICkit 3 Debug Express**

The MPLAB PICkit 3 allows debugging and programming of PIC® and dsPIC® Flash microcontrollers at a most affordable price point using the powerful graphical user interface of the MPLAB 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.

## 26.11 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.

## 26.12 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.

## 26.13 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™ and dsPICDEM™ demonstration/development board series of circuits, Microchip has a line of evaluation kits and demonstration software for analog filter design, KEELOQ® security ICs, CAN, IrDA®, PowerSmart battery management, SEEVAL® evaluation system, Sigma-Delta ADC, flow rate sensing, plus many more.

Also available are starter kits that contain everything needed to experience the specified device. This usually includes a single application and debug capability, all on one board.

Check the Microchip web page ([www.microchip.com](http://www.microchip.com)) for the complete list of demonstration, development and evaluation kits.



## 27.0 ELECTRICAL CHARACTERISTICS

This section provides an overview of dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 electrical characteristics. Additional information will be provided in future revisions of this document as it becomes available.

Absolute maximum ratings for the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 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<sup>(1)</sup>

Ambient temperature under bias	-40°C to +125°C
Storage temperature	-65°C to +150°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>(3)</sup>	-0.3V to (VDD + 0.3V)
Voltage on any 5V tolerant pin with respect to VSS, when VDD ≥ 3.0V <sup>(3)</sup>	-0.3V to +5.6V
Voltage on any 5V tolerant pin with respect to VSS, when VDD < 3.0V <sup>(3)</sup>	-0.3V to (VDD + 0.3V)
Maximum current out of VSS pin	300 mA
Maximum current into VDD pin <sup>(2)</sup>	250 mA
Maximum current sourced/sunk by any 4x I/O pin	15 mA
Maximum current sourced/sunk by any 8x I/O pin	25 mA
Maximum current sourced/sunk by any 16x I/O pin	45 mA
Maximum current sunk by all ports	200 mA
Maximum current sourced by all ports <sup>(2)</sup>	200mA

**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 27-2](#)).

**3:** See the “[Pin Diagrams](#)” section for 5V tolerant pins.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## 27.1 DC Characteristics

**TABLE 27-1: OPERATING MIPS VS. VOLTAGE**

Characteristic	VDD Range (in Volts)	Temp Range (in °C)	Max MIPS
			dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610
—	3.0-3.6V <sup>(1)</sup>	-40°C to +85°C	40
—	3.0-3.6V <sup>(1)</sup>	-40°C to +125°C	40

**Note 1:** Overall functional device operation at  $V_{BORMIN} < V_{DD} < V_{DDMIN}$  is tested but not characterized. All device analog modules such as the ADC, etc., will function but with degraded performance below  $V_{DDMIN}$ . See parameter BO10 in [Table 27-11](#) for the BOR values.

**TABLE 27-2: THERMAL OPERATING CONDITIONS**

Rating	Symbol	Min	Typ	Max	Unit
Industrial Temperature Devices					
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	—	+140	°C
Operating Ambient Temperature Range	TA	-40	—	+125	°C
Power Dissipation: Internal chip power dissipation: $P_{INT} = V_{DD} \times (I_{DD} - \Sigma I_{OH})$ I/O Pin Power Dissipation: $I/O = \Sigma (\{V_{DD} - V_{OH}\} \times I_{OH}) + \Sigma (V_{OL} \times I_{OL})$	PD	PINT + PI/O			W
Maximum Allowed Power Dissipation	PDMAX	$(T_J - T_A)/\theta_{JA}$			W

**TABLE 27-3: THERMAL PACKAGING CHARACTERISTICS**

Characteristic	Symbol	Typ	Max	Unit	Notes
Package Thermal Resistance, 64-Pin QFN (9x9x0.9 mm)	$\theta_{JA}$	28	—	°C/W	1
Package Thermal Resistance, 64-Pin TQFP (10x10x1 mm)	$\theta_{JA}$	39	—	°C/W	1
Package Thermal Resistance, 80-Pin TQFP (12x12x1 mm)	$\theta_{JA}$	53.1	—	°C/W	1
Package Thermal Resistance, 100-Pin TQFP (12x12x1 mm)	$\theta_{JA}$	43	—	°C/W	1
Package Thermal Resistance, 100-Pin TQFP (14x14x1 mm)	$\theta_{JA}$	43	—	°C/W	1

**Note 1:** Junction to ambient thermal resistance, Theta-JA ( $\theta_{JA}$ ) numbers are achieved by package simulations.

**TABLE 27-4: DC TEMPERATURE AND VOLTAGE SPECIFICATIONS**

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended				
Param No.	Symbol	Characteristic	Min	Typ <sup>(1)</sup>	Max	Units	Conditions
<b>Operating Voltage</b>							
DC10	VDD	Supply Voltage <sup>(4)</sup>	3.0	—	3.6	V	Industrial and extended
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 <sup>(3)</sup> 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.
- 2:** This is the limit to which VDD may be lowered without losing RAM data.
- 3:** These parameters are characterized but not tested in manufacturing.
- 4:** Overall functional device operation at VBORMIN < VDD < VDDMIN is tested but not characterized. All device analog modules such as the ADC, etc., will function but with degraded performance below VDDMIN. See parameter BO10 in [Table 27-11](#) for the BOR values.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

**TABLE 27-5: DC CHARACTERISTICS: OPERATING CURRENT (IDD)**

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended	
Parameter No.	Typical <sup>(1)</sup>	Max	Units	Conditions
<b>Operating Current (IDD)<sup>(2)</sup></b>				
DC20d	21	30	mA	-40°C
DC20a	21	30	mA	+25°C
DC20b	21	30	mA	+85°C
DC20c	22	30	mA	+125°C
DC21d	28	40	mA	-40°C
DC21a	28	40	mA	+25°C
DC21b	28	40	mA	+85°C
DC21c	29	40	mA	+125°C
DC22d	35	45	mA	-40°C
DC22a	35	45	mA	+25°C
DC22b	35	45	mA	+85°C
DC22c	36	45	mA	+125°C
DC23d	49	60	mA	-40°C
DC23a	49	60	mA	+25°C
DC23b	49	60	mA	+85°C
DC23c	50	60	mA	+125°C
DC24d	66	75	mA	-40°C
DC24a	66	75	mA	+25°C
DC24b	66	75	mA	+85°C
DC24c	67	75	mA	+125°C
DC25d	153	170	mA	-40°C
DC25a	154	170	mA	+25°C
DC25b	155	170	mA	+85°C
DC25c	156	170	mA	+125°C
DC26d	122	135	mA	-40°C
DC26a	123	135	mA	+25°C
DC26b	124	135	mA	+85°C
DC26c	125	135	mA	+125°C

**Note 1:** Data in "Typical" column is at 3.3V, +25°C unless otherwise stated.

**2:** 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)
- 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 (all PMDx bits are zeroed)
- CPU executing `while(1)` statement
- JTAG disabled

**3:** These parameters are characterized but not tested in manufacturing.

TABLE 27-5: DC CHARACTERISTICS: OPERATING CURRENT (IDD) (CONTINUED)

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended		
Parameter No.	Typical <sup>(1)</sup>	Max	Units	Conditions	
<b>Operating Current (IDD)<sup>(2)</sup></b>					
DC27d	107	120	mA	-40°C	3.3V 40 MIPS See <b>Note 2 and 3</b> , except PWM is operating at 1/4 speed (PTCON2 = 0x0002)
DC27a	108	120	mA	+25°C	
DC27b	109	120	mA	+85°C	
DC27c	110	120	mA	+125°C	
DC28d	88	100	mA	-40°C	3.3V 40 MIPS See <b>Note 2 and 3</b> , except PWM is operating at 1/8 speed (PTCON2 = 0x0003)
DC28a	89	100	mA	+25°C	
DC28b	89	100	mA	+85°C	
DC28c	89	100	mA	+125°C	

**Note 1:** Data in “Typical” column is at 3.3V, +25°C unless otherwise stated.

- 2:** 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)
  - 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 (all PMDx bits are zeroed)
  - CPU executing `while(1)` statement
  - JTAG disabled
- 3:** These parameters are characterized but not tested in manufacturing.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

**TABLE 27-6: DC CHARACTERISTICS: IDLE CURRENT (IDLE)**

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended			
Parameter No.	Typical <sup>(1)</sup>	Max	Units	Conditions		
<b>Idle Current (IDLE): Core Off Clock On Base Current<sup>(2)</sup></b>						
DC40d	8	15	mA	-40°C	3.3V	10 MIPS
DC40a	9	15	mA	+25°C		
DC40b	9	15	mA	+85°C		
DC40c	10	15	mA	+125°C		
DC41d	11	20	mA	-40°C	3.3V	16 MIPS <sup>(3)</sup>
DC41a	11	20	mA	+25°C		
DC41b	11	20	mA	+85°C		
DC41c	12	20	mA	+125°C		
DC42d	14	25	mA	-40°C	3.3V	20 MIPS <sup>(3)</sup>
DC42a	14	25	mA	+25°C		
DC42b	14	25	mA	+85°C		
DC42c	15	25	mA	+125°C		
DC43d	20	30	mA	-40°C	3.3V	30 MIPS <sup>(3)</sup>
DC43a	20	30	mA	+25°C		
DC43b	21	30	mA	+85°C		
DC43c	22	30	mA	+125°C		
DC44d	29	40	mA	-40°C	3.3V	40 MIPS
DC44a	29	40	mA	+25°C		
DC44b	30	40	mA	+85°C		
DC44c	31	40	mA	+125°C		

**Note 1:** Data in "Typical" column is at 3.3V, +25°C unless otherwise stated.

**2:** Base Idle current (IDLE) 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 (all PMDx bits are zeroed)
- The NVMSIDL bit (NVMCON<12>) = 1 (i.e., Flash regulator is set to stand-by while the device is in Idle mode)
- JTAG disabled

**3:** These parameters are characterized but not tested in manufacturing.

**TABLE 27-7: DC CHARACTERISTICS: POWER-DOWN CURRENT (IPD)**

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended		
Parameter No.	Typical <sup>(1)</sup>	Max	Units	Conditions	
<b>Power-Down Current (IPD)<sup>(2,4)</sup></b>					
DC60d	50	200	μA	-40°C	3.3V Base Power-Down Current
DC60a	50	200	μA	+25°C	
DC60b	200	500	μA	+85°C	
DC60c	600	1000	μA	+125°C	
DC61d	8	13	μA	-40°C	3.3V Watchdog Timer Current: ΔI <sub>WDT</sub> <sup>(3)</sup>
DC61a	10	15	μA	+25°C	
DC61b	12	20	μA	+85°C	
DC61c	13	25	μA	+125°C	

**Note 1:** Data in the Typical column is at 3.3V, +25°C unless otherwise stated.

**2:** 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 V<sub>SS</sub>
- $\overline{\text{MCLR}} = \text{VDD}$ , WDT and FSCM are disabled
- All peripheral modules are disabled (PMDx bits are all ones)
- The VREGS bit (RCON<8>) = 0 (i.e., core regulator is set to stand-by while the device is in Sleep mode)
- JTAG disabled

**3:** The Δ 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.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

**TABLE 27-8: DC CHARACTERISTICS: DOZE CURRENT (IDOZE)**

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended				
Parameter No.	Typical <sup>(1)</sup>	Max	Doze Ratio	Units	Conditions		
DC73a	45	60	1:2	mA	-40°C	3.3V	40 MIPS
DC73f	40	60	1:64	mA			
DC73g	40	60	1:128	mA			
DC70a	43	60	1:2	mA	+25°C	3.3V	40 MIPS
DC70f	38	60	1:64	mA			
DC70g	38	60	1:128	mA			
DC71a	42	60	1:2	mA	+85°C	3.3V	40 MIPS
DC71f	37	60	1:64	mA			
DC71g	37	60	1:128	mA			
DC72a	41	60	1:2	mA	+125°C	3.3V	40 MIPS
DC72f	36	60	1:64	mA			
DC72g	36	60	1:128	mA			

**Note 1:** Data in the Typical column is at 3.3V, +25°C unless otherwise stated.

**2:** 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 (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
- CPU, SRAM, program memory and data memory are operational
- No peripheral modules are operating; however, every peripheral is being clocked (all PMDx bits are zeroed)
- CPU executing `while(1)` statement
- JTAG disabled



TABLE 27-9: DC CHARACTERISTICS: I/O PIN INPUT SPECIFICATIONS

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended				
Param No.	Symbol	Characteristic	Min	Typ <sup>(1)</sup>	Max	Units	Conditions
	V <sub>IL</sub>	<b>Input Low Voltage</b>					
DI10		I/O Pins	V <sub>SS</sub>	—	0.2 V <sub>DD</sub>	V	—
DI15		MCLR	V <sub>SS</sub>	—	0.2 V <sub>DD</sub>	V	—
DI16		I/O Pins with OSC1 or SOSCI	V <sub>SS</sub>	—	0.2 V <sub>DD</sub>	V	—
DI18		I/O Pins with SDAx, SCLx	V <sub>SS</sub>	—	0.3 V <sub>DD</sub>	V	SMBus disabled
DI19		I/O Pins with SDAx, SCLx	V <sub>SS</sub>	—	0.8	V	SMBus enabled
	V <sub>IH</sub>	<b>Input High Voltage</b>					
DI20		I/O Pins Not 5V Tolerant <sup>(4)</sup>	0.7 V <sub>DD</sub>	—	V <sub>DD</sub>	V	—
DI21		I/O Pins 5V Tolerant <sup>(4)</sup>	0.7 V <sub>DD</sub>	—	5.5	V	—
DI28		SDAx, SCLx	0.7 V <sub>DD</sub>	—	5.5	V	SMBus disabled
DI29		SDAx, SCLx	2.1	—	5.5	V	SMBus enabled
	IC <sub>NPU</sub>	<b>CNx Pull-up Current</b>					
DI30			—	250	—	μA	V <sub>DD</sub> = 3.3V, V <sub>PIN</sub> = V <sub>SS</sub>
	I <sub>IL</sub>	<b>Input Leakage Current<sup>(2,3,4)</sup></b>					
DI50		I/O Pins with: 4x Driver Pins - RA0-RA7, RA14, RA15, RB0-RB15, RC1-RC4, RC12-RC14, RD0-RD2, RD8- RD12, RD14, RD15, RE8, RE9, RF0-RF8, RF12, RF13, RG0-RG3, RG6-RG9, RG14, RG15	—	—	±2	μA	V <sub>SS</sub> ≤V <sub>PIN</sub> ≤V <sub>DD</sub> , Pin at high-impedance
		8x Driver Pins - RC15	—	—	±4	μA	V <sub>SS</sub> ≤V <sub>PIN</sub> ≤V <sub>DD</sub> , Pin at high-impedance
		16x Driver Pins - RA9, RA10, RD3-RD7, RD13, RE0-RE7, RG12, RG13	—	—	±8	μA	V <sub>SS</sub> ≤V <sub>PIN</sub> ≤V <sub>DD</sub> , Pin at high-impedance
DI55		MCLR	—	—	±2	μA	V <sub>SS</sub> ≤V <sub>PIN</sub> ≤V <sub>DD</sub>
DI56		OSC1	—	—	±2	μA	V <sub>SS</sub> ≤V <sub>PIN</sub> ≤V <sub>DD</sub> , XT and HS modes

**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 the list of 5V tolerant I/O pins.
- 5:** V<sub>IL</sub> source < (V<sub>SS</sub> – 0.3). Characterized but not tested.
- 6:** Non-5V tolerant pins V<sub>IH</sub> source > (V<sub>DD</sub> + 0.3), 5V tolerant pins V<sub>IH</sub> 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 I<sub>ICL</sub> or I<sub>ICH</sub> 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 27-9: DC CHARACTERISTICS: I/O PIN INPUT SPECIFICATIONS (CONTINUED)

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended				
Param No.	Symbol	Characteristic	Min	Typ <sup>(1)</sup>	Max	Units	Conditions
DI60a	I <sub>ICL</sub>	Input Low Injection Current	0	—	-5 <sup>(3,5,8)</sup>	mA	All pins except V <sub>DD</sub> , V <sub>SS</sub> , AV <sub>DD</sub> , AV <sub>SS</sub> , MCLR, V <sub>CAP</sub> , SOSCI, SOSCO, and RB11
DI60b	I <sub>ICH</sub>	Input High Injection Current	0	—	+5 <sup>(6,7,8)</sup>	mA	All pins except V <sub>DD</sub> , V <sub>SS</sub> , AV <sub>DD</sub> , AV <sub>SS</sub> , MCLR, V <sub>CAP</sub> , SOSCI, SOSCO, RB11, and digital 5V tolerant designated pins <sup>(3)</sup>
DI60c	$\Sigma I_{ICT}$	Total Input Injection Current (sum of all I/O and control pins)	-20 <sup>(9)</sup>	—	+20 <sup>(9)</sup>	mA	Absolute instantaneous sum of all $\pm$ input injection currents from all I/O pins ( $ I_{ICL}  +  I_{ICH}  \leq \Sigma I_{ICT}$ )

- 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 the list of 5V tolerant I/O pins.
- 5:** V<sub>IL</sub> source < (V<sub>SS</sub> – 0.3). Characterized but not tested.
- 6:** Non-5V tolerant pins V<sub>IH</sub> source > (V<sub>DD</sub> + 0.3), 5V tolerant pins V<sub>IH</sub> 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 I<sub>ICL</sub> or I<sub>ICH</sub> 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 27-10: DC CHARACTERISTICS: I/O PIN OUTPUT SPECIFICATIONS

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param.	Symbol	Characteristic	Min.	Typ.	Max.	Units	Conditions
DO10	VOL	<b>Output Low Voltage</b> I/O Pins: 4x Sink Driver Pins - RA0-RA7, RA14, RA15, RB0-RB15, RC1- RC4, RC12-RC14, RD0-RD2, RD8-RD12, RD14, RD15, RE8, RE9, RF0-RF8, RF12, RF13, RG0- RG3, RG6-RG9, RG14, RG15	—	—	0.4	V	IOL ≤ 6 mA, VDD = 3.3V See <b>Note 1</b>
		<b>Output Low Voltage</b> I/O Pins: 8x Sink Driver Pin - RC15	—	—	0.4	V	IOL ≤ 10 mA, VDD = 3.3V See <b>Note 1</b>
		<b>Output Low Voltage</b> I/O Pins: 16x Sink Driver Pins - RA9, RA10, RD3-RD7, RD13, RE0-RE7, RG12, RG13	—	—	0.4	V	IOL ≤ 18 mA, VDD = 3.3V See <b>Note 1</b>
DO20	VOH	<b>Output High Voltage</b> I/O Pins: 4x Sink Driver Pins - RA0-RA7, RA14, RA15, RB0-RB15, RC1- RC4, RC12-RC14, RD0-RD2, RD8-RD12, RD14, RD15, RE8, RE9, RF0-RF8, RF12, RF13, RG0- RG3, RG6-RG9, RG14, RG15	2.4	—	—	V	IOH ≥ -6 mA, VDD = 3.3V See <b>Note 1</b>
		<b>Output High Voltage</b> I/O Pins: 8x Sink Driver Pin - RC15	2.4	—	—	V	IOH ≥ -10 mA, VDD = 3.3V See <b>Note 1</b>
		<b>Output High Voltage</b> I/O Pins: 16x Sink Driver Pins - RA9, RA10, RD3-RD7, RD13, RE0-RE7, RG12, RG13	2.4	—	—	V	IOH ≥ -18 mA, VDD = 3.3V See <b>Note 1</b>

**Note 1:** Parameters are characterized, but not tested.

TABLE 27-10: DC CHARACTERISTICS: I/O PIN OUTPUT SPECIFICATIONS (CONTINUED)

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended						
Param.	Symbol	Characteristic	Min.	Typ.	Max.	Units	Conditions		
DO20A	VOH1	<b>Output High Voltage</b> I/O Pins: 4x Sink Driver Pins - RA0-RA7, RA14, RA15, RB0-RB15, RC1- RC4, RC12-RC14, RD0-RD2, RD8-RD12, RD14, RD15, RE8, RE9, RF0-RF8, RF12, RF13, RG0- RG3, RG6-RG9, RG14, RG15	1.5	—	—	V	IOH ≥ -12 mA, VDD = 3.3V See <b>Note 1</b>		
			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>		
				<b>Output High Voltage</b> I/O Pins: 8x Sink Driver Pin - RC15	1.5	—	—	V	IOH ≥ -16 mA, VDD = 3.3V See <b>Note 1</b>
					2.0	—	—		IOH ≥ -12 mA, VDD = 3.3V See <b>Note 1</b>
					3.0	—	—		IOH ≥ -4 mA, VDD = 3.3V See <b>Note 1</b>
				<b>Output High Voltage</b> I/O Pins: 16x Sink Driver Pins - RA9, RA10, RD3-RD7, RD13, RE0-RE7, RG12, RG13	1.5	—	—	V	IOH ≥ -30 mA, VDD = 3.3V See <b>Note 1</b>
					2.0	—	—		IOH ≥ -25 mA, VDD = 3.3V See <b>Note 1</b>
					3.0	—	—		IOH ≥ -8 mA, VDD = 3.3V See <b>Note 1</b>

**Note 1:** Parameters are characterized, but not tested.

TABLE 27-11: ELECTRICAL CHARACTERISTICS: BOR

DC CHARACTERISTICS			Standard Operating Conditions (see Note 3): 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	Symbol	Characteristic	Min <sup>(1)</sup>	Typ	Max	Units	Conditions
BO10	VBOR	BOR Event on VDD Transition High-to-Low	2.6	—	2.95	V	See <b>Note 2</b>

**Note 1:** Parameters are for design guidance only and are not tested in manufacturing.

**2:** The device will operate as normal until the VDDMIN threshold is reached.

**3:** Overall functional device operation at VBORMIN < VDD < VDDMIN is tested but not characterized. All device analog modules such as the ADC, etc., will function but with degraded performance below VDDMIN.

TABLE 27-12: DC CHARACTERISTICS: PROGRAM MEMORY

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended				
Param No.	Symbol	Characteristic	Min	Typ <sup>(1)</sup>	Max	Units	Conditions
<b>Program Flash Memory</b>							
D130	EP	Cell Endurance	10,000	—	—	E/W	-40°C to +125°C
D131	VPR	VDD for Read	VMIN	—	3.6	V	VMIN = Minimum operating voltage
D132B	VPEW	VDD for Self-Timed Write	VMIN	—	3.6	V	VMIN = Minimum operating voltage
D134	TRETD	Characteristic Retention	20	—	—	Year	Provided no other specifications are violated, -40°C to +125°C
D135	IDDP	Supply Current during Programming	—	10	—	mA	—
D136a	TRW	Row Write Time	1.43	—	1.58	ms	TRW = 11064 FRC cycles, TA = +85°C, See <b>Note 2</b>
D136b	TRW	Row Write Time	1.39	—	1.63	ms	TRW = 11064 FRC cycles, TA = +125°C, See <b>Note 2</b>
D137a	TPE	Page Erase Time	21.8	—	24.1	ms	TPE = 168517 FRC cycles, TA = +85°C, See <b>Note 2</b>
D137b	TPE	Page Erase Time	21.1	—	24.8	ms	TPE = 168517 FRC cycles, TA = +125°C, See <b>Note 2</b>
D138a	TWW	Word Write Cycle Time	45.8	—	50.7	µs	TWW = 355 FRC cycles, TA = +85°C, See <b>Note 2</b>
D138b	TWW	Word Write Cycle Time	44.5	—	52.3	µs	TWW = 355 FRC cycles, TA = +125°C, See <b>Note 2</b>

**Note 1:** Data in “Typ” column is at 3.3V, +25°C unless otherwise stated.

**2:** Other conditions: FRC = 7.37 MHz, TUN<5:0> = b'011111 (for Min), TUN<5:0> = b'100000 (for Max). This parameter depends on the FRC accuracy (see [Table 27-20](#)) 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”](#).

TABLE 27-13: INTERNAL VOLTAGE REGULATOR SPECIFICATIONS

Operating Conditions: -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended							
Param No.	Symbol	Characteristics	Min	Typ	Max	Units	Comments
—	CEFC	External Filter Capacitor Value <sup>(1)</sup>	22	—	—	µF	Capacitor must be low series resistance (< 0.5 Ohms)

**Note 1:** Typical VCAP voltage = 2.5 volts when VDD ≥ VDDMIN.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

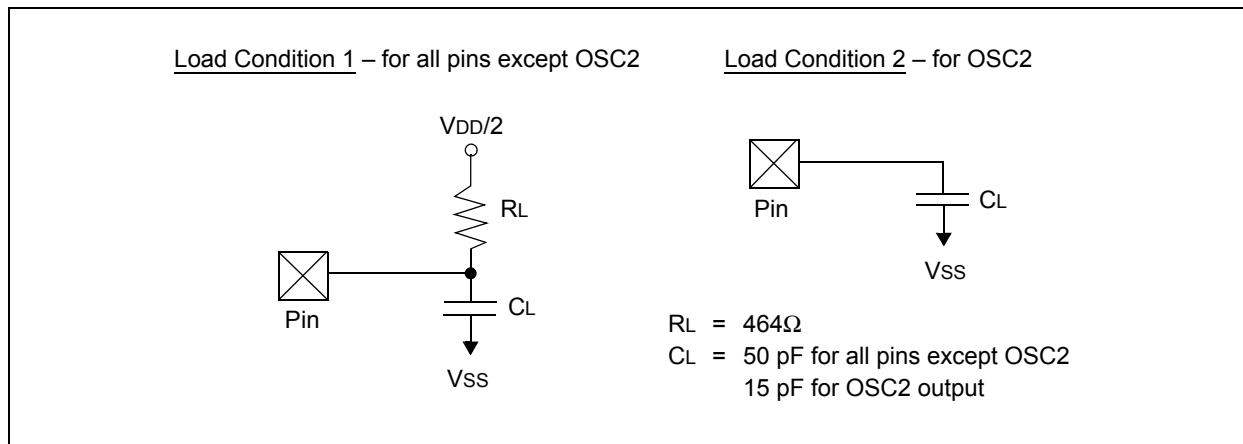
## 27.2 AC Characteristics and Timing Parameters

This section defines dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 AC characteristics and timing parameters.

**TABLE 27-14: TEMPERATURE AND VOLTAGE SPECIFICATIONS – AC**

<b>AC CHARACTERISTICS</b>	<p><b>Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)</b></p> <p>Operating temperature <math>-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}</math> for Industrial  <math>-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}</math> for Extended</p> <p>Operating voltage <math>V_{DD}</math> range as described in <a href="#">Section 27.0 “Electrical Characteristics”</a>.</p>
---------------------------	--

**FIGURE 27-1: LOAD CONDITIONS FOR DEVICE TIMING SPECIFICATIONS**



**TABLE 27-15: CAPACITIVE LOADING REQUIREMENTS ON OUTPUT PINS**

Param No.	Symbol	Characteristic	Min	Typ	Max	Units	Conditions
DO50	Cosco	OSC2 Pin	—	—	15	pF	In XT and HS modes when external clock is used to drive OSC1
DO56	Cio	All I/O Pins and OSC2	—	—	50	pF	EC mode
DO58	CB	SCLx, SDAx	—	—	400	pF	In I <sup>2</sup> C™ mode

FIGURE 27-2: EXTERNAL CLOCK TIMING

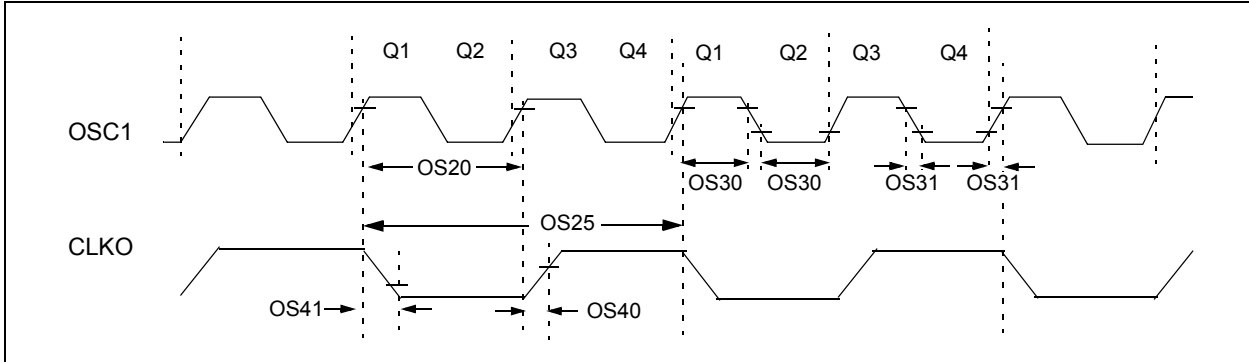


TABLE 27-16: EXTERNAL CLOCK TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	Symb	Characteristic	Min	Typ <sup>(1)</sup>	Max	Units	Conditions
OS10	FIN	External CLKI Frequency (External clocks allowed only in EC and ECPLL modes)	DC	—	40	MHz	EC
		Oscillator Crystal Frequency	3.5	—	10	MHz	XT
			— 10	— —	33 40	kHz MHz	Sosc HS
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	—
OS41	GM	External Oscillator Transconductance	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.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

**TABLE 27-17: PLL CLOCK TIMING SPECIFICATIONS (V<sub>DD</sub> = 3.0V TO 3.6V)**

AC CHARACTERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤T <sub>A</sub> ≤+85°C for Industrial -40°C ≤T <sub>A</sub> ≤+125°C for Extended					
Param No.	Symbol	Characteristic	Min	Typ <sup>(1)</sup>	Max	Units	Conditions
OS50	FPLLI	PLL Voltage Controlled Oscillator (VCO) Input Frequency Range	0.8	—	8	MHz	ECPLL, 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) <sup>(2)</sup>	-3	0.5	3	%	Measured over 100 ms period

- Note 1:** Data in “Typ” column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested in manufacturing.
- 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 bases or communication clocks use this formula:

$$\text{Peripheral Clock Jitter} = \frac{DCLK}{\sqrt{\left(\frac{FOSC}{\text{Peripheral Bit Rate Clock}}\right)}}$$

For example: Fosc = 32 MHz, DCLK = 3%, SPI bit rate clock, (i.e., SCK) is 2 MHz.

$$\text{SPI SCK Jitter} = \left[ \frac{DCLK}{\sqrt{\left(\frac{32 \text{ MHz}}{2 \text{ MHz}}\right)}} \right] = \left[ \frac{3\%}{\sqrt{16}} \right] = \left[ \frac{3\%}{4} \right] = 0.75\%$$

**TABLE 27-18: AUXILIARY PLL CLOCK TIMING SPECIFICATIONS (V<sub>DD</sub> = 3.0V TO 3.6V)**

AC CHARACTERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤T <sub>A</sub> ≤+85°C for Industrial -40°C ≤T <sub>A</sub> ≤+125°C for Extended					
Param No.	Symbol	Characteristic	Min	Typ <sup>(1)</sup>	Max	Units	Conditions
OS56	FHPOUT	On-Chip 16x PLL CCO Frequency	112	118	120	MHz	—
OS57	FHPIN	On-Chip 16x PLL Phase Detector Input Frequency	7.0	7.37	7.5	MHz	—
OS58	Tsu	Frequency Generator Lock Time	—	—	10	μs	—

- Note 1:** Data in “Typ” column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested in manufacturing.



**TABLE 27-19: AC CHARACTERISTICS: INTERNAL FRC ACCURACY**

AC CHARACTERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for industrial -40°C ≤TA ≤+125°C for Extended					
Param No.	Characteristic	Min	Typ	Max	Units	Conditions	
<b>Internal FRC Accuracy @ FRC Frequency = 7.37 MHz<sup>(1,2)</sup></b>							
F20a	FRC	-1	—	+1	%	-40°C ≤TA ≤+85°C	VDD = 3.0-3.6V
F20b	FRC	-2	—	+2	%	-40°C ≤TA ≤+125°C	VDD = 3.0-3.6V

**Note 1:** Frequency calibrated at +25°C and 3.3V. The TUN<5:0> bits can be used to compensate for temperature drift.

**TABLE 27-20: AC CHARACTERISTICS: INTERNAL LPRC ACCURACY**

AC CHARACTERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended					
Param No.	Characteristic	Min	Typ	Max	Units	Conditions	
<b>LPRC @ 32.768 kHz<sup>(1)</sup></b>							
F21a	LPRC	-40	—	+40	%	-40°C ≤TA ≤+85°C	
F21b	LPRC	-70	—	+70	%	-40°C ≤TA ≤+125°C	

**Note 1:** Change of LPRC frequency as VDD changes.

FIGURE 27-3: I/O TIMING CHARACTERISTICS

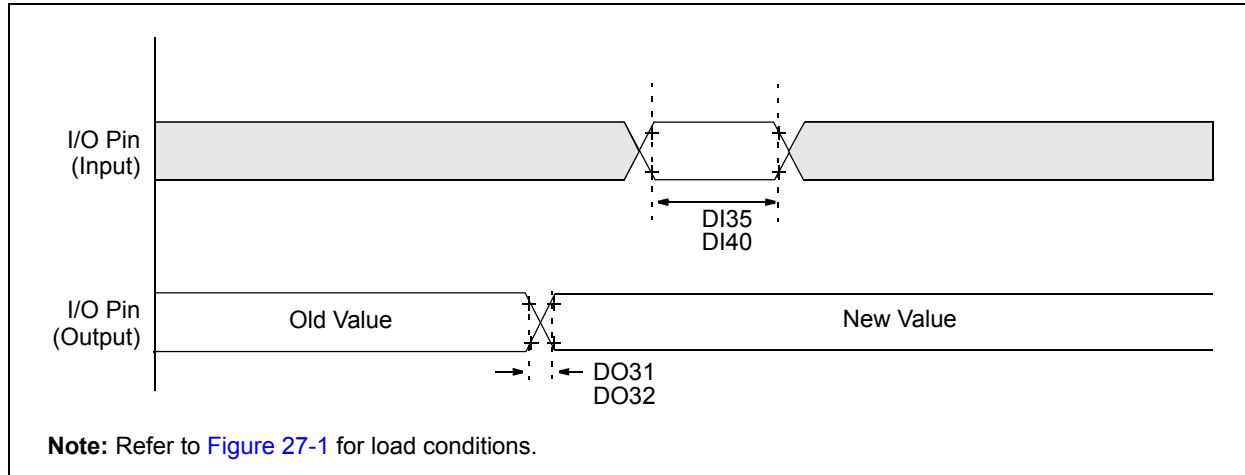
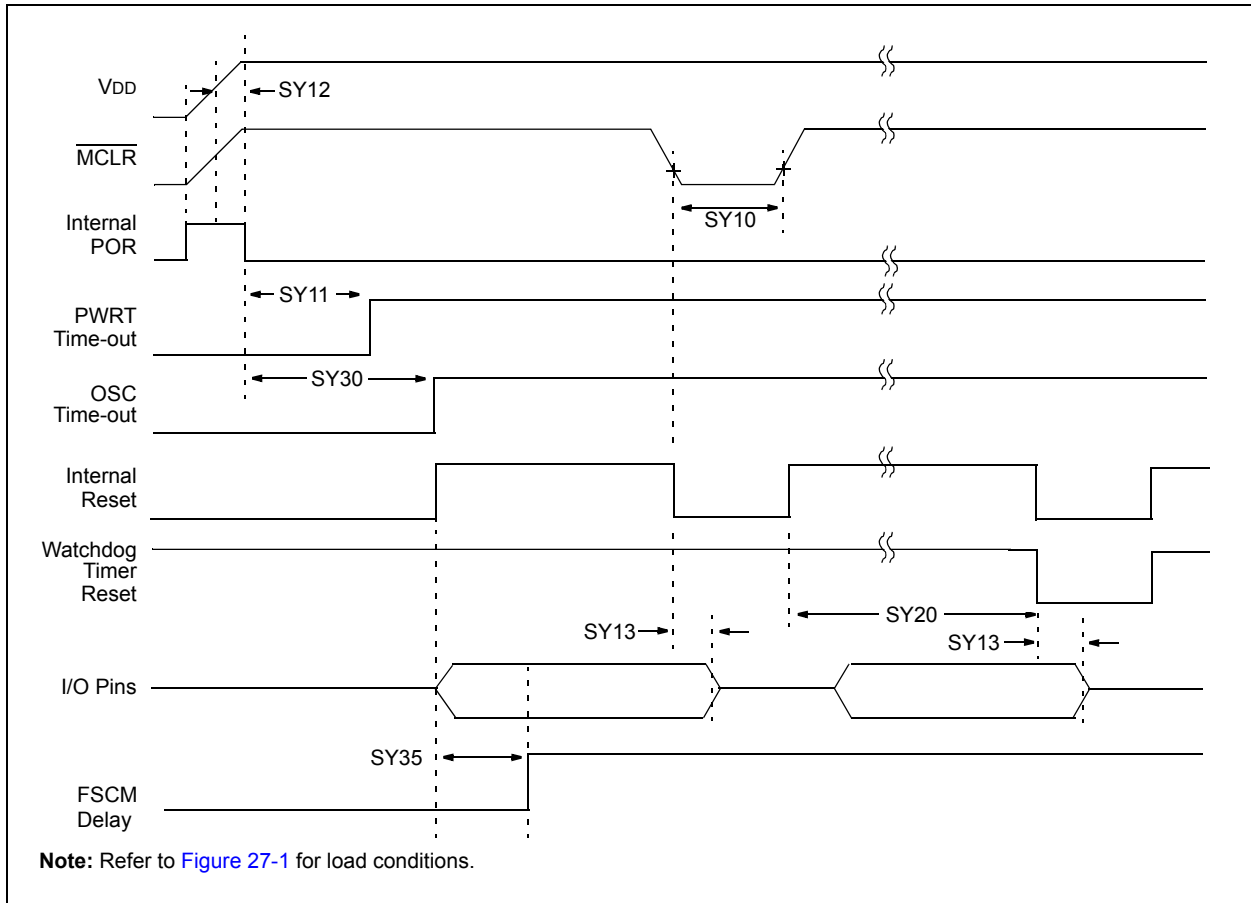


TABLE 27-21: I/O TIMING REQUIREMENTS

AC CHARACTERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended					
Param No.	Symbol	Characteristic	Min	Typ <sup>(1)</sup>	Max	Units	Conditions
DO31	TioR	Port Output Rise Time:					Refer to Figure 27-1 for test conditions
		4x Source Driver Pins - RA0-RA7, RA14, RA15, RB0-RB15, RC1-RC4, RC12-RC14, RD0-RD2, RD8-RD12, RD14, RD15, RE8, RE9, RF0-RF8, RF12, RF13, RG0-RG3, RG6-RG9, RG14, RG15	—	10	25	ns	
		8x Source Driver Pins - RC15	—	8	20	ns	
		16x Source Driver Pins - RE0-RE7, RG12, RG13	—	6	15	ns	
DO32	TioF	Port Output Fall Time:					Refer to Figure 27-1 for test conditions
		4x Source Driver Pins - RA0-RA7, RA14, RA15, RB0-RB15, RC1-RC4, RC12-RC14, RD0-RD2, RD8-RD12, RD14, RD15, RE8, RE9, RF0-RF8, RF12, RF13, RG0-RG3, RG6-RG9, RG14, RG15	—	10	25	ns	
		8x Source Driver Pins - RC15	—	8	20	ns	
		16x Source Driver Pins - RE0-RE7, RG12, RG13	—	6	15	ns	
DI35	TINP	INTx Pin High or Low Time (input)	20	—	—	ns	—
DI40	TRBP	CNx High or Low Time (input)	2	—	—	TCY	—

**Note 1:** Data in “Typ” column is at 3.3V, +25°C unless otherwise stated.

**FIGURE 27-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING CHARACTERISTICS**



# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

**TABLE 27-22: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER, POWER-UP TIMER TIMING REQUIREMENTS**

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended				
Param No.	Symbol	Characteristic <sup>(1)</sup>	Min	Typ <sup>(2)</sup>	Max	Units	Conditions
SY10	TMCL	MCLR Pulse Width (low)	2	—	—	μs	-40°C to +85°C
SY11	TPWRT	Power-up Timer Period	—	2 4 8 16 32 64 128	—	ms	-40°C to +85°C User programmable
SY12	TPOR	Power-on Reset Delay	3	10	30	μs	-40°C to +85°C
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	—	—	—	ms	See <a href="#">Section 24.4 “Watchdog Timer (WDT)”</a> and LPRC parameter <a href="#">F21a (Table 27-20)</a> .
SY30	TOST	Oscillator Start-up Time	—	1024 TOSC	—	—	TOSC = OSC1 period

- Note 1:** These parameters are characterized but not tested in manufacturing.  
**Note 2:** Data in “Typ” column is at 3.3V, +25°C unless otherwise stated.

FIGURE 27-5: TIMER1, 2 AND 3 EXTERNAL CLOCK TIMING CHARACTERISTICS

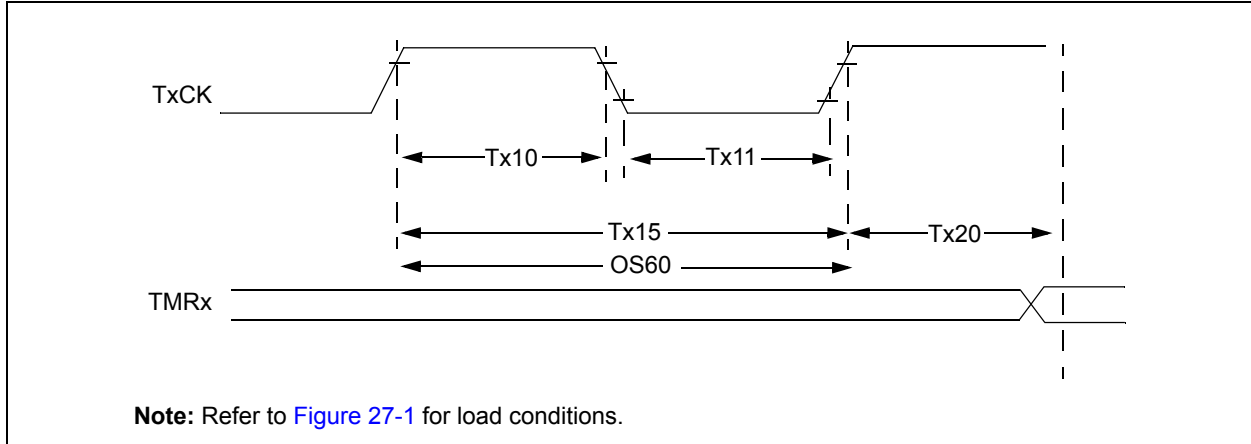


TABLE 27-23: TIMER1 EXTERNAL CLOCK TIMING REQUIREMENTS<sup>(1)</sup>

AC CHARACTERISTICS				Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	Symbol	Characteristic		Min	Typ	Max	Units	Conditions
TA10	TtxH	TxCK High Time	Synchronous, no prescaler	T <sub>cy</sub> + 20	—	—	ns	Must also meet parameter TA15. N = prescale value (1, 8, 64, 256)
			Synchronous, with prescaler	(T <sub>cy</sub> + 20)/N	—	—	ns	
			Asynchronous	20	—	—	ns	
TA11	TtxL	TxCK Low Time	Synchronous, no prescaler	(T <sub>cy</sub> + 20)	—	—	ns	Must also meet parameter TA15. N = prescale value (1, 8, 64, 256)
			Synchronous, with prescaler	(T <sub>cy</sub> + 20)/N	—	—	ns	
			Asynchronous	20	—	—	ns	
TA15	TtxP	TxCK Input Period	Synchronous, no prescaler	2 T <sub>cy</sub> + 40	—	—	ns	—
			Synchronous, with prescaler	Greater of: 40 ns or (2 T <sub>cy</sub> + 40)/N	—	—	—	N = prescale value (1, 8, 64, 256)
			Asynchronous	40	—	—	ns	—
OS60	Ft1	SOSCI/T1CK Oscillator Input frequency Range (oscillator enabled by setting bit TCS (T1CON<1>))		DC	—	50	kHz	—
TA20	TCKEXTMRL	Delay from External TxCK Clock Edge to Timer Increment		0.75 T <sub>cy</sub> + 40	—	1.75 T <sub>cy</sub> + 40	—	—

**Note 1:** Timer1 is a Type A.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

**TABLE 27-24: TIMER2, TIMER4 EXTERNAL CLOCK TIMING REQUIREMENTS**

AC CHARACTERISTICS				Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended				
Param No.	Symbol	Characteristic <sup>(1)</sup>		Min	Typ	Max	Units	Conditions
TB10	TtxH	TxCK High Time	Synchronous mode	Greater of: 20 or (Tcy + 20)/N	—	—	ns	Must also meet parameter TB15 N = prescale value (1, 8, 64, 256)
TB11	TtxL	TxCK Low Time	Synchronous mode	Greater of: 20 or (Tcy + 20)/N	—	—	ns	Must also meet parameter TB15 N = prescale value (1, 8, 64, 256)
TB15	TtxP	TxCK Input Period	Synchronous mode	Greater of: 40 or (2 Tcy + 40)/N	—	—	ns	N = prescale value (1, 8, 64, 256)
TB20	TCKEXTMRL	Delay from External TxCK Clock Edge to Timer Increment		0.75 Tcy + 40	—	1.75 Tcy + 40	ns	—

**Note 1:** These parameters are characterized, but are not tested in manufacturing.

**TABLE 27-25: TIMER3, TIMER5 EXTERNAL CLOCK TIMING REQUIREMENTS**

AC CHARACTERISTICS				Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended				
Param No.	Symbol	Characteristic <sup>(1)</sup>		Min	Typ	Max	Units	Conditions
TC10	TtxH	TxCK High Time	Synchronous	Tcy + 20	—	—	ns	Must also meet parameter TC15
TC11	TtxL	TxCK Low Time	Synchronous	Tcy + 20	—	—	ns	Must also meet parameter TC15
TC15	TtxP	TxCK Input Period	Synchronous, with prescaler	2 Tcy + 40	—	—	ns	—
TC20	TCKEXTMRL	Delay from External TxCK Clock Edge to Timer Increment		0.75 Tcy + 40	—	1.75 Tcy + 40	ns	—

**Note 1:** These parameters are characterized, but are not tested in manufacturing.

FIGURE 27-6: INPUT CAPTURE (CAPx) TIMING CHARACTERISTICS

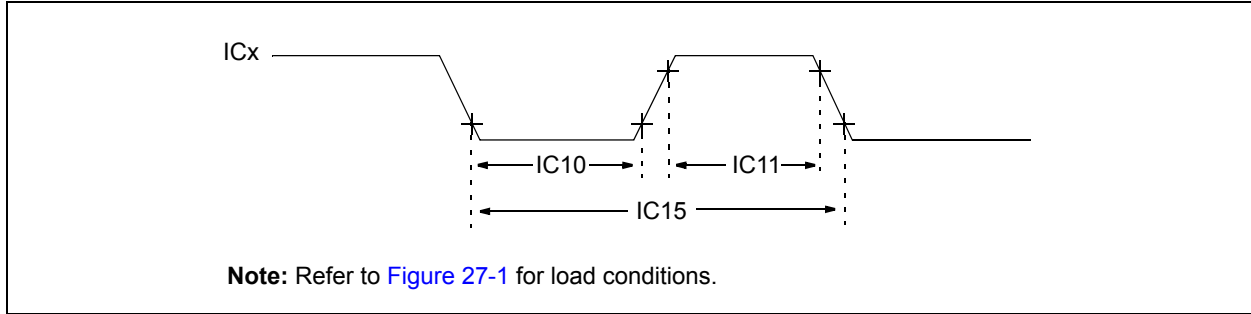


TABLE 27-26: INPUT CAPTURE TIMING REQUIREMENTS

AC CHARACTERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended					
Param No.	Symbol	Characteristic <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	—	ns	N = prescale value (1, 4, 16)

Note 1: These parameters are characterized but not tested in manufacturing.

FIGURE 27-7: OUTPUT COMPARE MODULE (OCx) TIMING CHARACTERISTICS

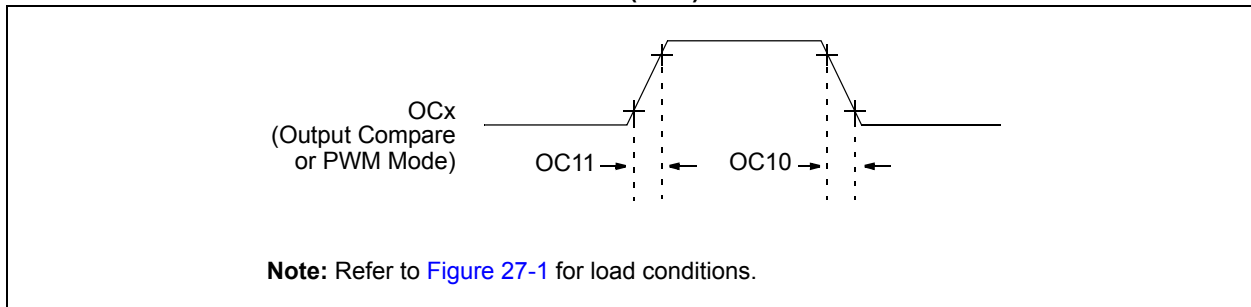
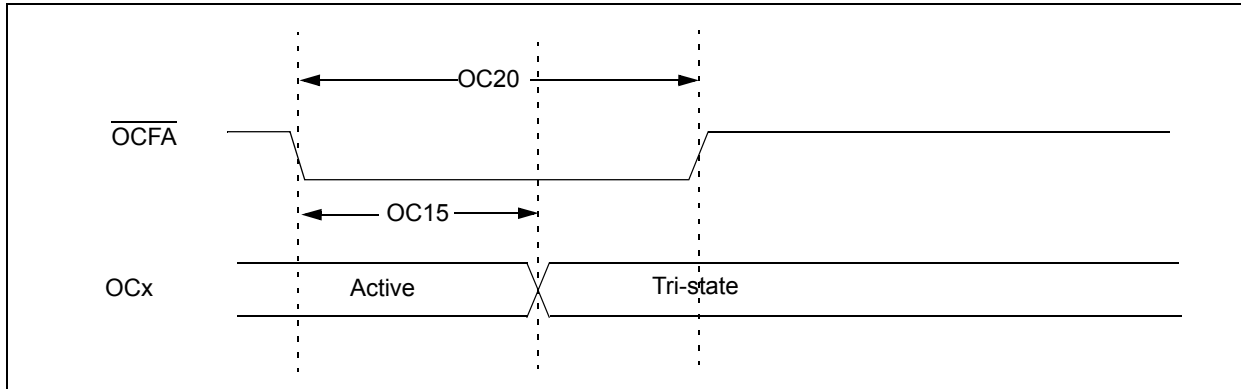


TABLE 27-27: OUTPUT COMPARE MODULE TIMING REQUIREMENTS

AC CHARACTERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended					
Param No.	Symbol	Characteristic <sup>(1)</sup>	Min	Typ	Max	Units	Conditions
OC10	TccF	OCx Output Fall Time	—	—	—	ns	See parameter DO32
OC11	TccR	OCx Output Rise Time	—	—	—	ns	See parameter DO31

Note 1: These parameters are characterized but not tested in manufacturing.

**FIGURE 27-8: OC/PWM MODULE TIMING CHARACTERISTICS**



**TABLE 27-28: SIMPLE OC/PWM MODE TIMING REQUIREMENTS**

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended				
Param No.	Symbol	Characteristic <sup>(1)</sup>	Min	Typ	Max	Units	Conditions
OC15	T <sub>FD</sub>	Fault Input to PWM I/O Change	—	—	T <sub>CY</sub> + 20	ns	—
OC20	T <sub>FLT</sub>	Fault Input Pulse Width	T <sub>CY</sub> + 20	—	—	ns	—

**Note 1:** These parameters are characterized but not tested in manufacturing.



FIGURE 27-9: HIGH-SPEED PWM MODULE FAULT TIMING CHARACTERISTICS

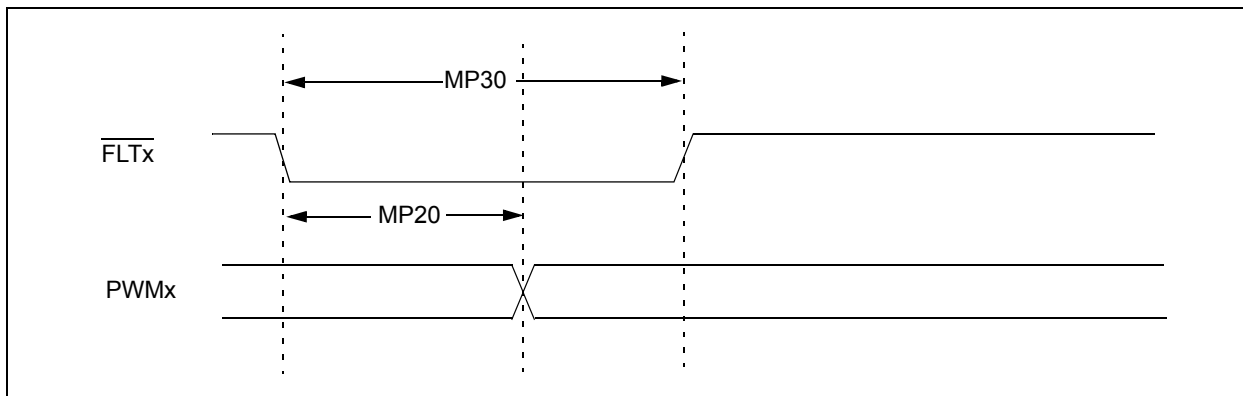


FIGURE 27-10: HIGH-SPEED PWM MODULE TIMING CHARACTERISTICS

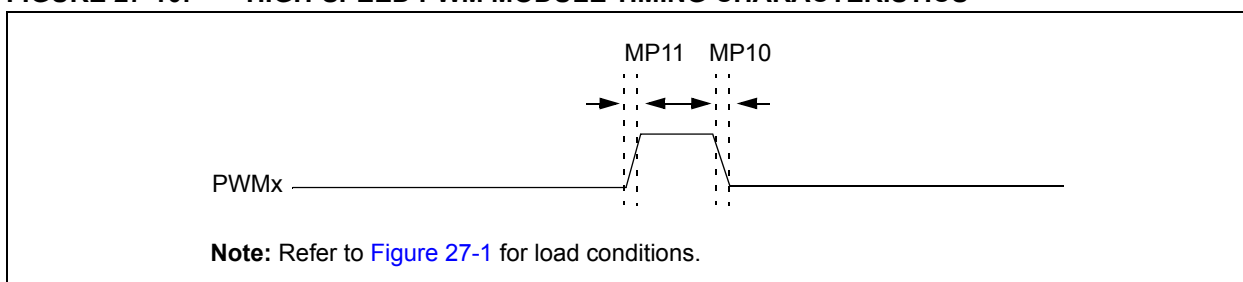


TABLE 27-29: HIGH-SPEED PWM MODULE TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	Symbol	Characteristic <sup>(1)</sup>	Min	Typ	Max	Units	Conditions
MP10	TFPWM	PWM Output Fall Time	—	2.5	—	ns	—
MP11	TRPWM	PWM Output Rise Time	—	2.5	—	ns	—
MP20	T <sub>FD</sub>	Fault Input ↓ to PWM I/O Change	—	—	15	ns	DTC<10> = 10
MP30	T <sub>FH</sub>	Minimum PWM Fault Pulse Width	8	—	—	ns	—
MP31	TPDLY	Tap Delay	1.04	—	—	ns	ACLK = 120 MHz
MP32	ACLK	PWM Input Clock	—	—	120	MHz	See <b>Note 2</b>

**Note 1:** These parameters are characterized but not tested in manufacturing.

**Note 2:** This parameter is a maximum allowed input clock for the PWM module.

TABLE 27-30: SPIx MAXIMUM DATA/CLOCK RATE SUMMARY

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended			
Maximum Data Rate	Master Transmit Only (Half-Duplex)	Master Transmit/Receive (Full-Duplex)	Slave Transmit/Receive (Full-Duplex)	CKE	CKP	SMP
15 MHz	Table 27-31	—	—	0,1	0,1	0,1
10 MHz	—	Table 27-32	—	1	0,1	1
10 MHz	—	Table 27-33	—	0	0,1	1
15 MHz	—	—	Table 27-34	1	0	0
11 MHz	—	—	Table 27-35	1	1	0
15 MHz	—	—	Table 27-36	0	1	0
11 MHz	—	—	Table 27-37	0	0	0

FIGURE 27-11: SPIx MASTER MODE (HALF-DUPLEX, TRANSMIT ONLY CKE = 0) TIMING CHARACTERISTICS

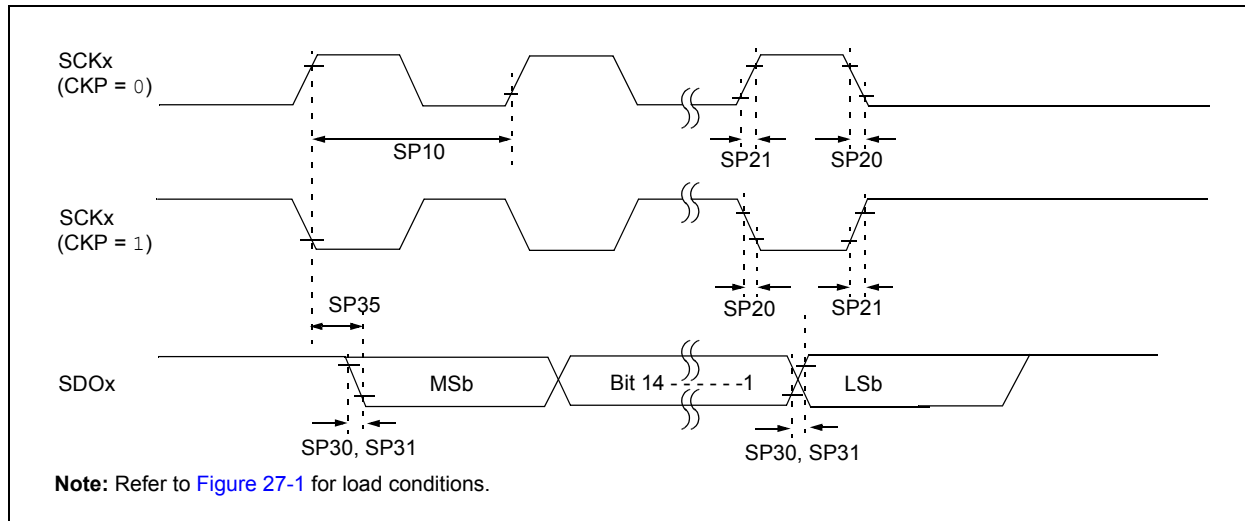
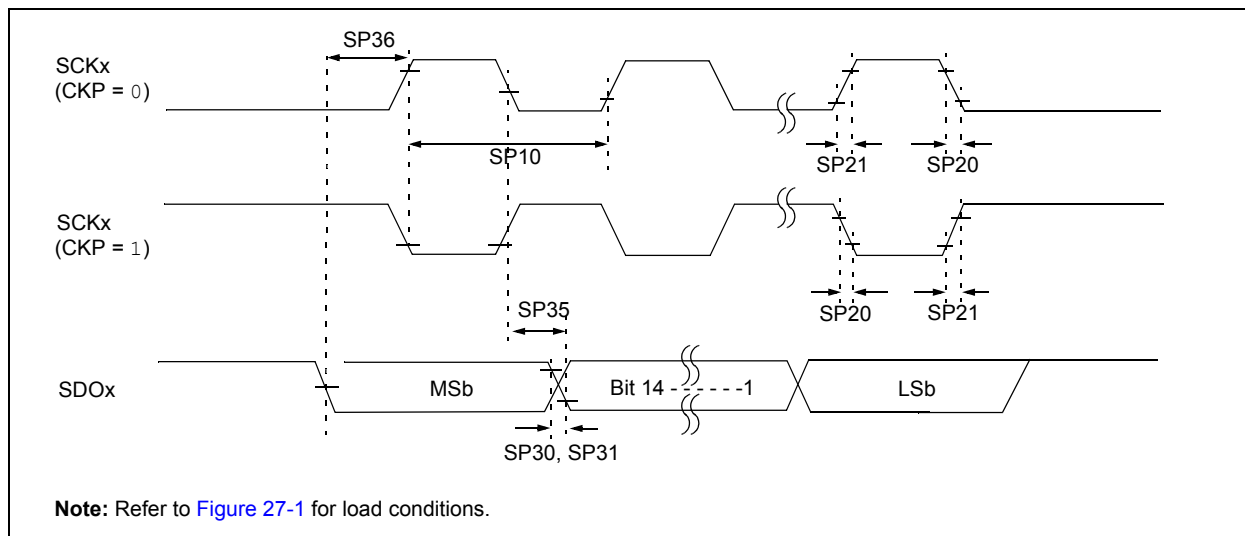


FIGURE 27-12: SPIx MASTER MODE (HALF-DUPLEX, TRANSMIT ONLY CKE = 1) TIMING CHARACTERISTICS

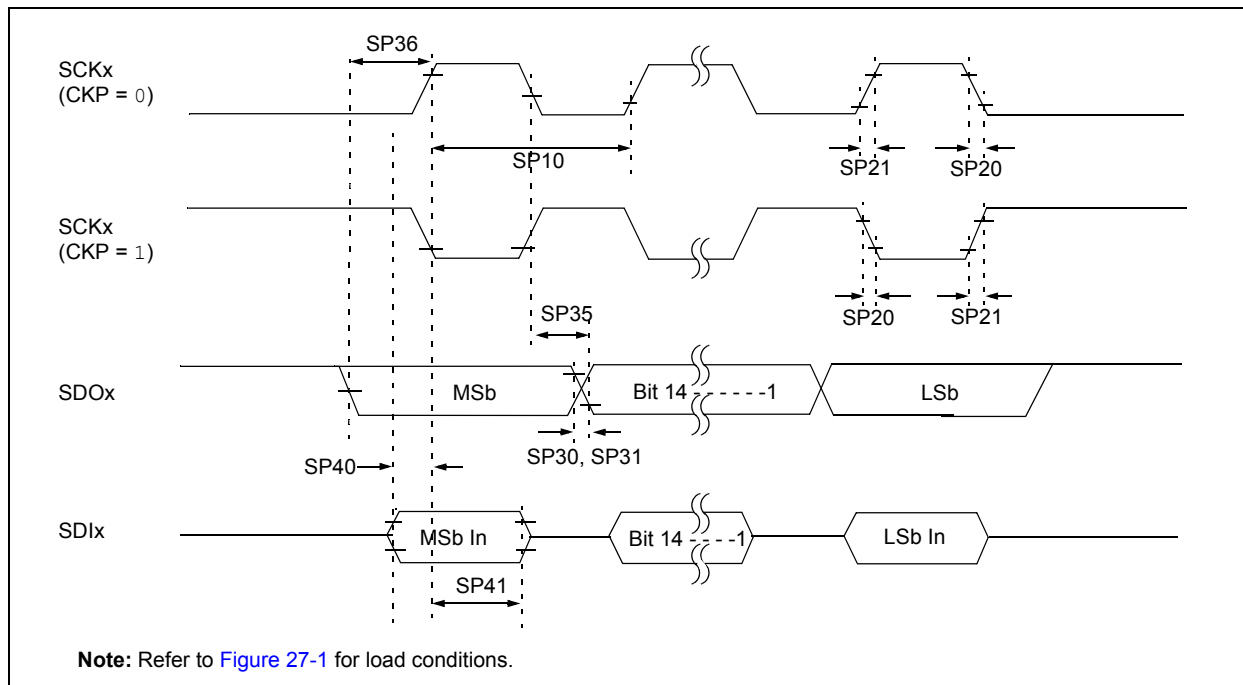


**TABLE 27-31: SPIx MASTER MODE (HALF-DUPLEX, TRANSMIT ONLY) TIMING REQUIREMENTS**

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°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 <b>Note 3</b>
SP20	TscF	SCKx Output Fall Time	—	—	—	ns	See parameter <a href="#">DO32</a> and <b>Note 4</b>
SP21	TscR	SCKx Output Rise Time	—	—	—	ns	See parameter <a href="#">DO31</a> and <b>Note 4</b>
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See parameter <a href="#">DO32</a> and <b>Note 4</b>
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See parameter <a href="#">DO31</a> and <b>Note 4</b>
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.
- Note 2:** Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.
- Note 3:** The minimum clock period for SCKx is 66.7 ns. Therefore, the clock generated in Master mode must not violate this specification.
- Note 4:** Assumes 50 pF load on all SPIx pins.

**FIGURE 27-13: SPIx MASTER MODE (FULL-DUPLEX, CKE = 1, CKP = x, SMP = 1) TIMING CHARACTERISTICS**

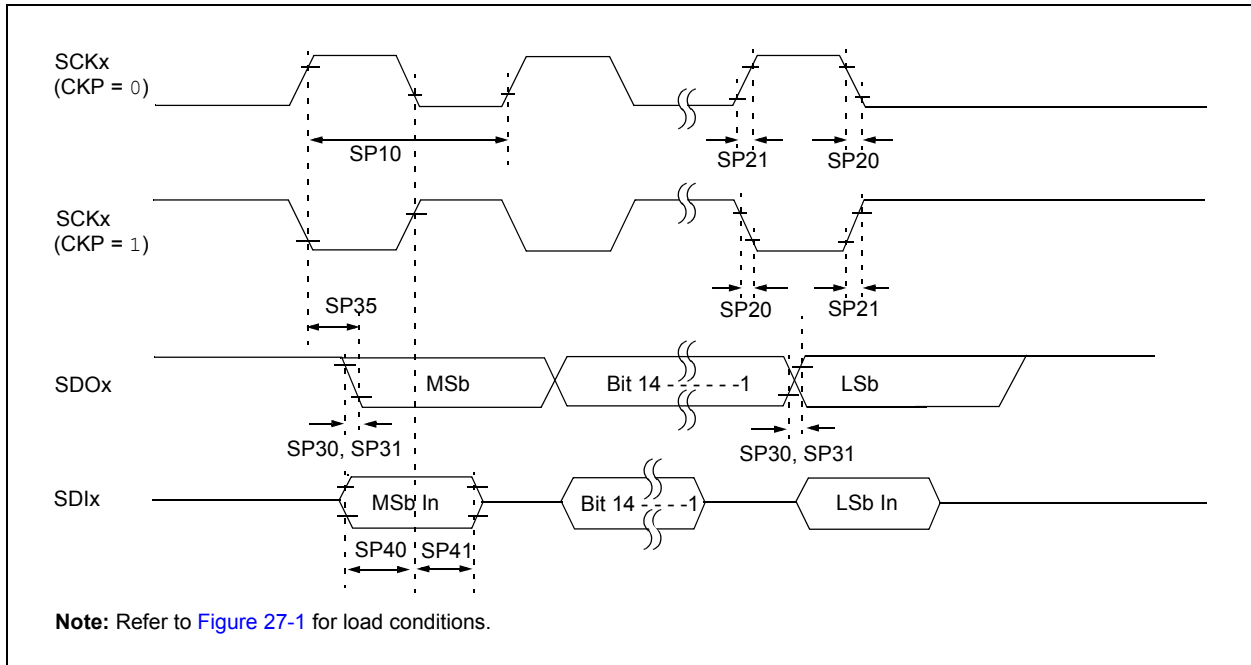


**TABLE 27-32: SPIx MASTER MODE (FULL-DUPLEX, CKE = 1, CKP = x, SMP = 1) TIMING REQUIREMENTS**

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ 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 <b>Note 3</b>
SP20	TscF	SCKx Output Fall Time	—	—	—	ns	See parameter <a href="#">DO32</a> and <b>Note 4</b>
SP21	TscR	SCKx Output Rise Time	—	—	—	ns	See parameter <a href="#">DO31</a> and <b>Note 4</b>
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See parameter <a href="#">DO32</a> and <b>Note 4</b>
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See parameter <a href="#">DO31</a> 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.  
**Note 2:** Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.  
**Note 3:** The minimum clock period for SCKx is 100 ns. The clock generated in Master mode must not violate this specification.  
**Note 4:** Assumes 50 pF load on all SPIx pins.

**FIGURE 27-14: SPIx MASTER MODE (FULL-DUPLEX, CKE = 0, CKP = x, SMP = 1) TIMING CHARACTERISTICS**

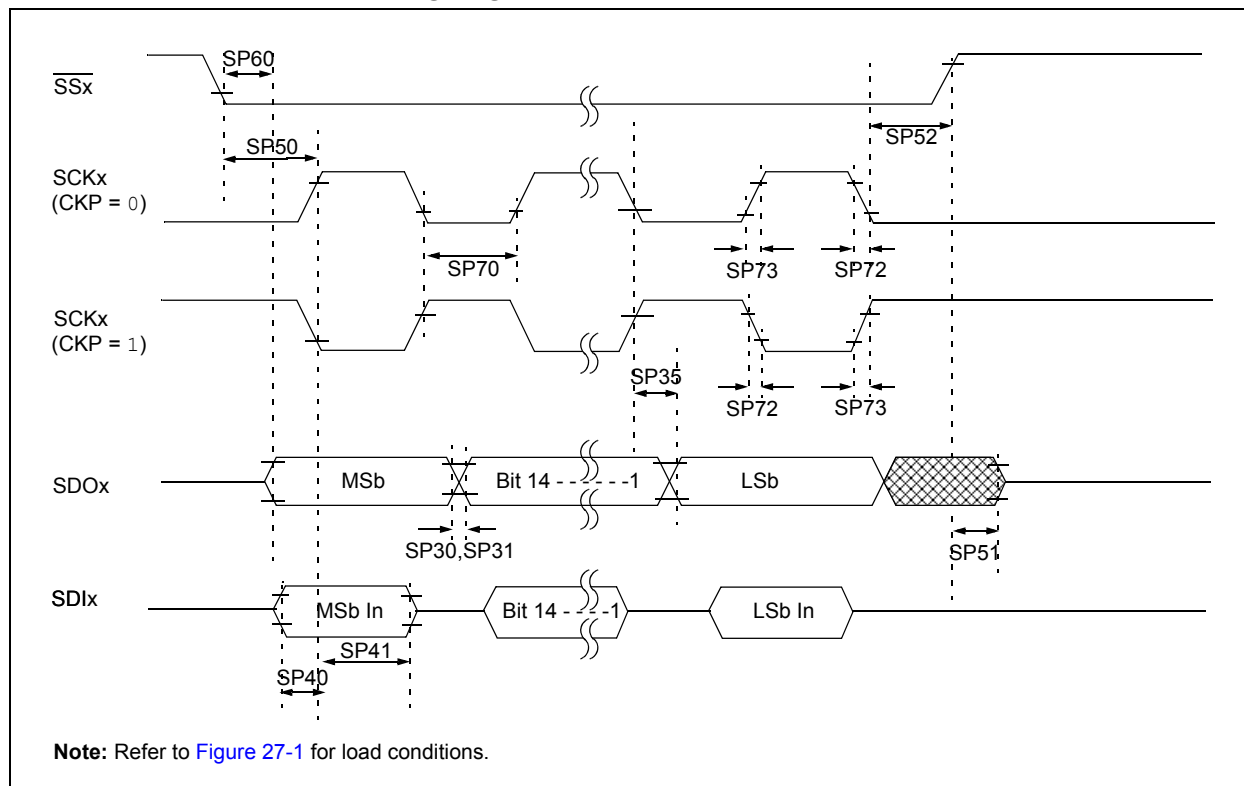


**TABLE 27-33: SPIx MASTER MODE (FULL-DUPLEX, CKE = 0, CKP = x, SMP = 1) TIMING REQUIREMENTS**

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended				
Param No.	Symbol	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 <b>DO32</b> and <b>Note 4</b>
SP21	TscR	SCKx Output Rise Time	—	—	—	ns	See parameter <b>DO31</b> and <b>Note 4</b>
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See parameter <b>DO32</b> and <b>Note 4</b>
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See parameter <b>DO31</b> 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.  
**Note 2:** Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.  
**Note 3:** The minimum clock period for SCKx is 100 ns. The clock generated in Master mode must not violate this specification.  
**Note 4:** Assumes 50 pF load on all SPIx pins.

**FIGURE 27-15: SPIx SLAVE MODE (FULL-DUPLEX, CKE = 1, CKP = 0, SMP = 0) TIMING CHARACTERISTICS**



**TABLE 27-34: SPIx SLAVE MODE (FULL-DUPLEX, CKE = 1, CKP = 0, SMP = 0) TIMING REQUIREMENTS**

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended				
Param No.	Symbol	Characteristic <sup>(1)</sup>	Min	Typ <sup>(2)</sup>	Max	Units	Conditions
SP70	TscP	Maximum SCK Input Frequency	—	—	15	MHz	See <b>Note 3</b>
SP72	TscF	SCKx Input Fall Time	—	—	—	ns	See parameter <a href="#">DO32</a> and <b>Note 4</b>
SP73	TscR	SCKx Input Rise Time	—	—	—	ns	See parameter <a href="#">DO31</a> and <b>Note 4</b>
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See parameter <a href="#">DO32</a> and <b>Note 4</b>
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See parameter <a href="#">DO31</a> 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	—
SP50	TssL2sch, TssL2scL	$\overline{SSx} \downarrow$ to SCKx $\uparrow$ or SCKx Input	120	—	—	ns	—
SP51	TssH2doZ	$\overline{SSx} \uparrow$ to SDOx Output High-Impedance <sup>(4)</sup>	10	—	50	ns	—
SP52	Tsch2ssH TscL2ssH	$\overline{SSx}$ after SCKx Edge	1.5 Tcy + 40	—	—	ns	See <b>Note 4</b>
SP60	TssL2doV	SDOx Data Output Valid after $\overline{SSx}$ Edge	—	—	50	ns	—

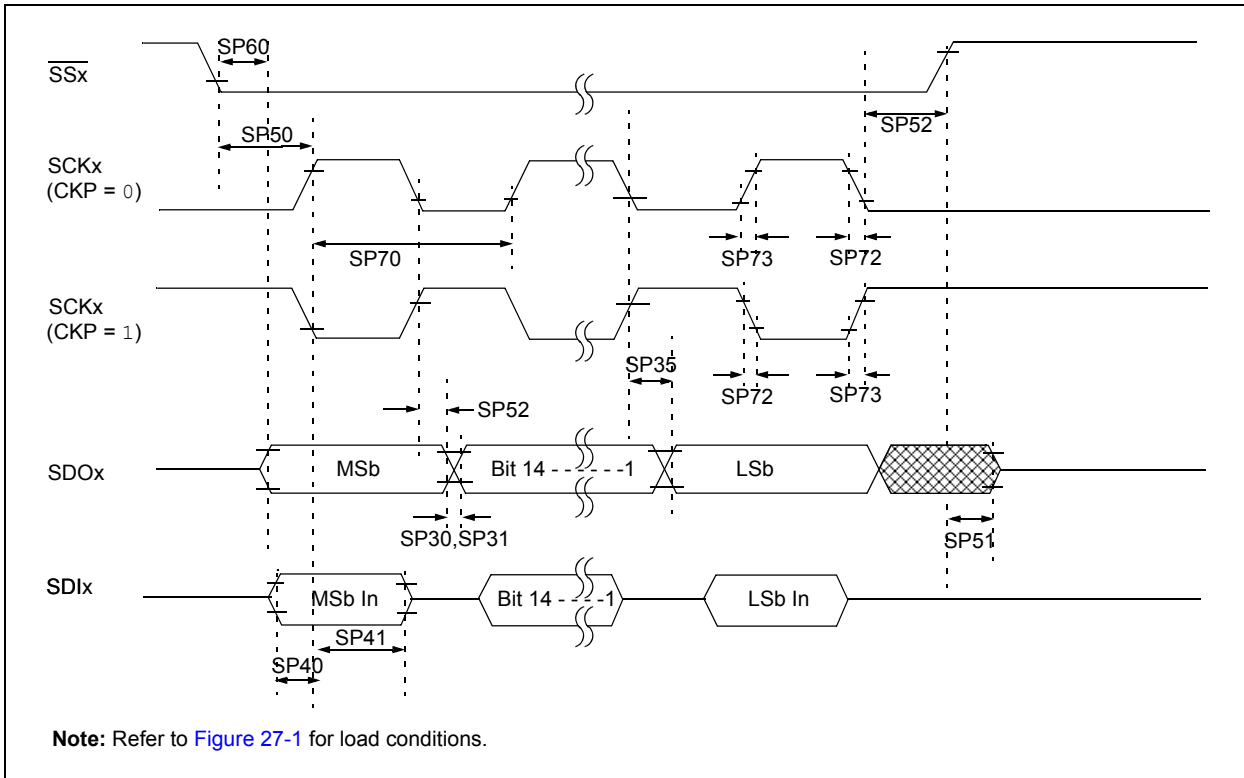
**Note 1:** These parameters are characterized, but are not tested in manufacturing.

**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 specification.

**4:** Assumes 50 pF load on all SPIx pins.

**FIGURE 27-16: SPIx SLAVE MODE (FULL-DUPLEX, CKE = 1, CKP = 1, SMP = 0) TIMING CHARACTERISTICS**





# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

**TABLE 27-35: SPIx SLAVE MODE (FULL-DUPLEX, CKE = 1, CKP = 1, SMP = 0) TIMING REQUIREMENTS**

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	Symbol	Characteristic <sup>(1)</sup>	Min	Typ <sup>(2)</sup>	Max	Units	Conditions
SP70	TscP	Maximum SCK Input Frequency	—	—	11	MHz	See <b>Note 3</b>
SP72	TscF	SCKx Input Fall Time	—	—	—	ns	See parameter <b>DO32</b> and <b>Note 4</b>
SP73	TscR	SCKx Input Rise Time	—	—	—	ns	See parameter <b>DO31</b> and <b>Note 4</b>
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See parameter <b>DO32</b> and <b>Note 4</b>
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See parameter <b>DO31</b> 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	—
SP50	TssL2scH, TssL2scL	$\overline{SSx}$ ↓ to SCKx ↑ or SCKx Input	120	—	—	ns	—
SP51	TssH2doZ	$\overline{SSx}$ ↑ to SDOx Output High-Impedance <sup>(4)</sup>	10	—	50	ns	—
SP52	Tsch2ssH TscL2ssH	$\overline{SSx}$ after SCKx Edge	1.5 Tcy + 40	—	—	ns	See <b>Note 4</b>
SP60	TssL2doV	SDOx Data Output Valid after $\overline{SSx}$ Edge	—	—	50	ns	—

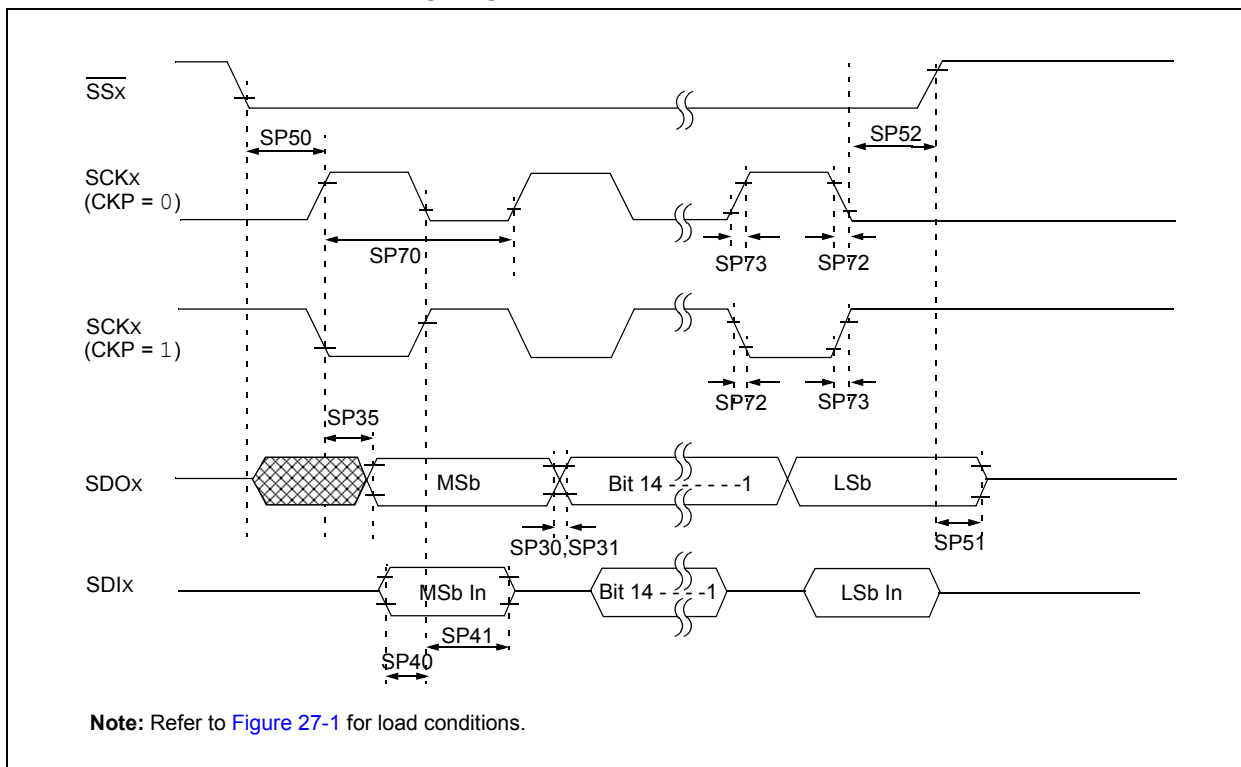
**Note 1:** These parameters are characterized, but are not tested in manufacturing.

**Note 2:** Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.

**Note 3:** The minimum clock period for SCKx is 91 ns. Therefore, the SCK clock generated by the Master must not violate this specification.

**Note 4:** Assumes 50 pF load on all SPIx pins.

**FIGURE 27-17: SPIx SLAVE MODE (FULL-DUPLEX CKE = 0, CKP = 1, SMP = 0) TIMING CHARACTERISTICS**

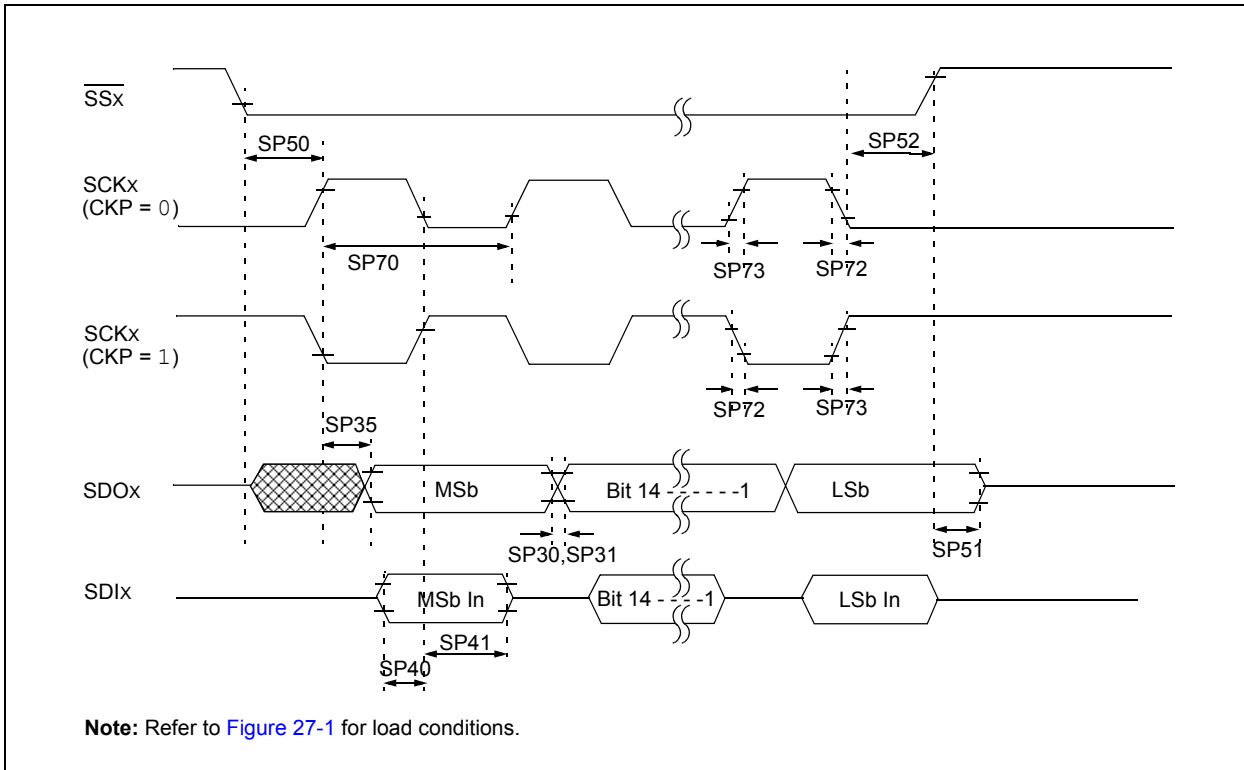


**TABLE 27-36: SPIx SLAVE MODE (FULL-DUPLEX, CKE = 0, CKP = 1, SMP = 0) TIMING REQUIREMENTS**

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	Symbol	Characteristic <sup>(1)</sup>	Min	Typ <sup>(2)</sup>	Max	Units	Conditions
SP70	TscP	Maximum SCK Input Frequency	—	—	15	MHz	See <b>Note 3</b>
SP72	TscF	SCKx Input Fall Time	—	—	—	ns	See parameter <b>DO32</b> and <b>Note 4</b>
SP73	TscR	SCKx Input Rise Time	—	—	—	ns	See parameter <b>DO31</b> and <b>Note 4</b>
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See parameter <b>DO32</b> and <b>Note 4</b>
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See parameter <b>DO31</b> 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	—
SP50	TssL2scH, TssL2scL	$\overline{SSx} \downarrow$ to SCKx $\uparrow$ or SCKx Input	120	—	—	ns	—
SP51	TssH2doZ	$\overline{SSx} \uparrow$ to SDOx Output High-Impedance <sup>(4)</sup>	10	—	50	ns	—
SP52	Tsch2ssH TscL2ssH	$\overline{SSx}$ after SCKx Edge	1.5 TCY + 40	—	—	ns	See <b>Note 4</b>

- Note 1:** These parameters are characterized, but are not tested in manufacturing.  
**Note 2:** Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.  
**Note 3:** The minimum clock period for SCKx is 66.7 ns. Therefore, the SCK clock generated by the Master must not violate this specification.  
**Note 4:** Assumes 50 pF load on all SPIx pins.

**FIGURE 27-18: SPIx SLAVE MODE (FULL-DUPLEX, CKE = 0, CKP = 0, SMP = 0) TIMING CHARACTERISTICS**



**TABLE 27-37: SPIx SLAVE MODE (FULL-DUPLEX, CKE = 0, CKP = 0, SMP = 0) TIMING REQUIREMENTS**

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	Symbol	Characteristic <sup>(1)</sup>	Min	Typ <sup>(2)</sup>	Max	Units	Conditions
SP70	TscP	Maximum SCK Input Frequency	—	—	11	MHz	See <b>Note 3</b>
SP72	TscF	SCKx Input Fall Time	—	—	—	ns	See parameter <b>DO32</b> and <b>Note 4</b>
SP73	TscR	SCKx Input Rise Time	—	—	—	ns	See parameter <b>DO31</b> and <b>Note 4</b>
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See parameter <b>DO32</b> and <b>Note 4</b>
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See parameter <b>DO31</b> 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	—
SP50	TssL2scH, TssL2scL	$\overline{SSx} \downarrow$ to SCKx $\uparrow$ or SCKx Input	120	—	—	ns	—
SP51	TssH2doZ	$\overline{SSx} \uparrow$ to SDOx Output High-Impedance <sup>(4)</sup>	10	—	50	ns	—
SP52	Tsch2ssH TscL2ssH	$\overline{SSx}$ after SCKx Edge	1.5 TCY + 40	—	—	ns	See <b>Note 4</b>

- 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 specification.
- 4:** Assumes 50 pF load on all SPIx pins.

FIGURE 27-19: I2Cx BUS START/STOP BITS TIMING CHARACTERISTICS (MASTER MODE)

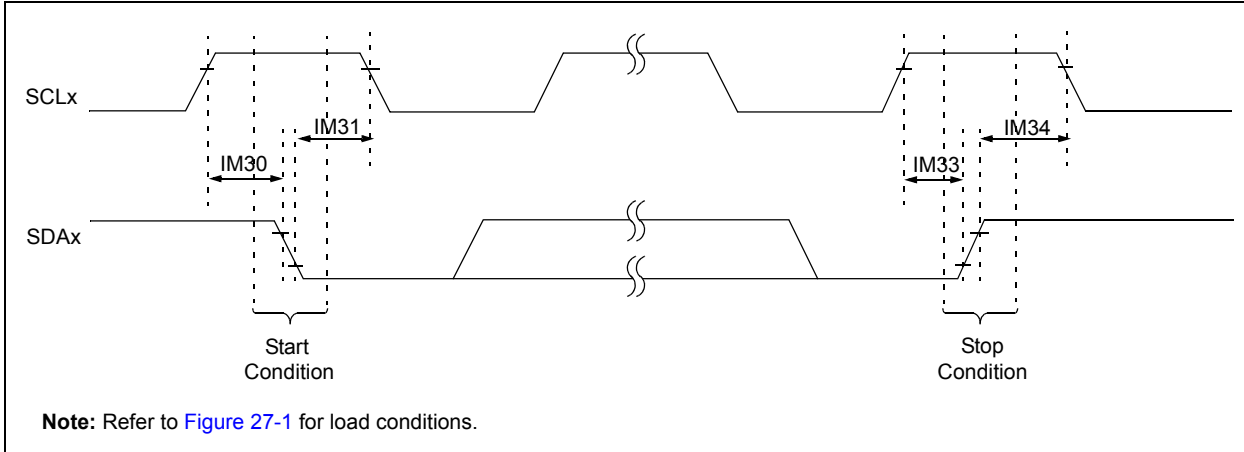


FIGURE 27-20: I2Cx BUS DATA TIMING CHARACTERISTICS (MASTER MODE)

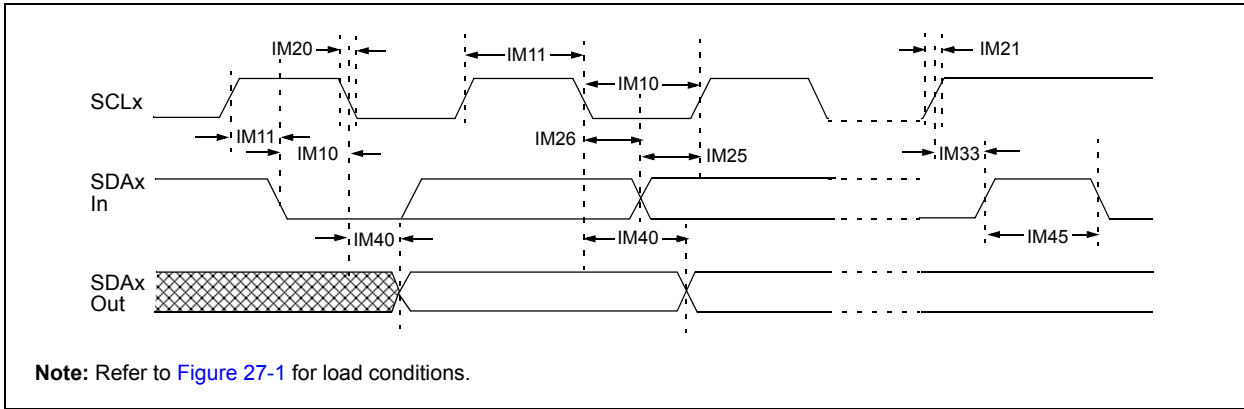


TABLE 27-38: I2Cx BUS DATA TIMING REQUIREMENTS (MASTER MODE)

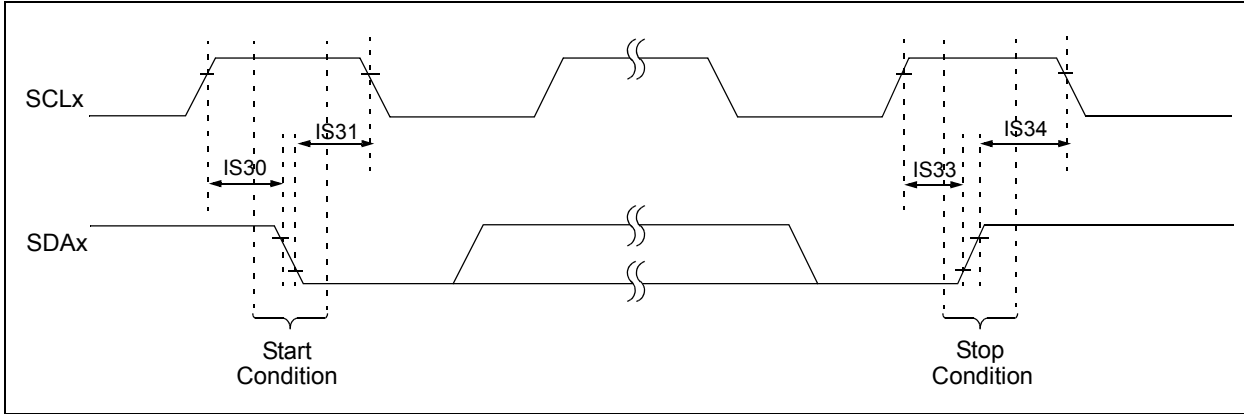
AC CHARACTERISTICS				Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended			
Param No.	Symbol	Characteristic		Min <sup>(1)</sup>	Max	Units	Conditions
IM10	TLO:SCL	Clock Low Time	100 kHz mode	$T_{CY}/2$ (BRG + 1)	—	$\mu\text{s}$	—
			400 kHz mode	$T_{CY}/2$ (BRG + 1)	—	$\mu\text{s}$	—
			1 MHz mode <sup>(2)</sup>	$T_{CY}/2$ (BRG + 1)	—	$\mu\text{s}$	—
IM11	THI:SCL	Clock High Time	100 kHz mode	$T_{CY}/2$ (BRG + 1)	—	$\mu\text{s}$	—
			400 kHz mode	$T_{CY}/2$ (BRG + 1)	—	$\mu\text{s}$	—
			1 MHz mode <sup>(2)</sup>	$T_{CY}/2$ (BRG + 1)	—	$\mu\text{s}$	—
IM20	TF:SCL	SDAx and SCLx Fall Time	100 kHz mode	—	300	ns	Cb is specified to be from 10 to 400 pF
			400 kHz mode	$20 + 0.1 C_B$	300	ns	
			1 MHz mode <sup>(2)</sup>	—	100	ns	
IM21	TR:SCL	SDAx and SCLx Rise Time	100 kHz mode	—	1000	ns	Cb is specified to be from 10 to 400 pF
			400 kHz mode	$20 + 0.1 C_B$	300	ns	
			1 MHz mode <sup>(2)</sup>	—	300	ns	
IM25	TSU:DAT	Data Input Setup Time	100 kHz mode	250	—	ns	—
			400 kHz mode	100	—	ns	
			1 MHz mode <sup>(2)</sup>	40	—	ns	
IM26	THD:DAT	Data Input Hold Time	100 kHz mode	0	—	$\mu\text{s}$	—
			400 kHz mode	0	0.9	$\mu\text{s}$	
			1 MHz mode <sup>(2)</sup>	0.2	—	$\mu\text{s}$	
IM30	TSU:STA	Start Condition Setup Time	100 kHz mode	$T_{CY}/2$ (BRG + 1)	—	$\mu\text{s}$	Only relevant for Repeated Start condition
			400 kHz mode	$T_{CY}/2$ (BRG + 1)	—	$\mu\text{s}$	
			1 MHz mode <sup>(2)</sup>	$T_{CY}/2$ (BRG + 1)	—	$\mu\text{s}$	
IM31	THD:STA	Start Condition Hold Time	100 kHz mode	$T_{CY}/2$ (BRG + 1)	—	$\mu\text{s}$	After this period the first clock pulse is generated
			400 kHz mode	$T_{CY}/2$ (BRG + 1)	—	$\mu\text{s}$	
			1 MHz mode <sup>(2)</sup>	$T_{CY}/2$ (BRG + 1)	—	$\mu\text{s}$	
IM33	TSU:STO	Stop Condition Setup Time	100 kHz mode	$T_{CY}/2$ (BRG + 1)	—	$\mu\text{s}$	—
			400 kHz mode	$T_{CY}/2$ (BRG + 1)	—	$\mu\text{s}$	
			1 MHz mode <sup>(2)</sup>	$T_{CY}/2$ (BRG + 1)	—	$\mu\text{s}$	
IM34	THD:STO	Stop Condition Hold Time	100 kHz mode	$T_{CY}/2$ (BRG + 1)	—	ns	—
			400 kHz mode	$T_{CY}/2$ (BRG + 1)	—	ns	
			1 MHz mode <sup>(2)</sup>	$T_{CY}/2$ (BRG + 1)	—	ns	
IM40	TAA:SCL	Output Valid From Clock	100 kHz mode	—	3500	ns	—
			400 kHz mode	—	1000	ns	—
			1 MHz mode <sup>(2)</sup>	—	400	ns	—
IM45	TBF:SDA	Bus Free Time	100 kHz mode	4.7	—	$\mu\text{s}$	Time the bus must be free before a new transmission can start
			400 kHz mode	1.3	—	$\mu\text{s}$	
			1 MHz mode <sup>(2)</sup>	0.5	—	$\mu\text{s}$	
IM50	CB	Bus Capacitive Loading	—	400	pF	—	
IM51	TPGD	Pulse Gobbler Delay	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/PIC24F Family Reference Manual”.

**2:** Maximum pin capacitance = 10 pF for all I2Cx pins (for 1 MHz mode only).

**3:** Typical value for this parameter is 130 ns.

**FIGURE 27-21: I2Cx BUS START/STOP BITS TIMING CHARACTERISTICS (SLAVE MODE)**



**FIGURE 27-22: I2Cx BUS DATA TIMING CHARACTERISTICS (SLAVE MODE)**

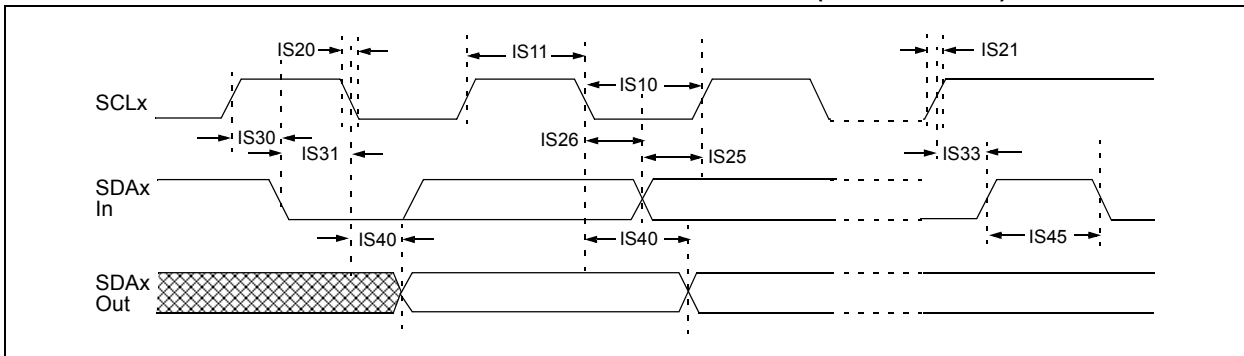




TABLE 27-39: I2Cx BUS DATA TIMING REQUIREMENTS (SLAVE MODE)

AC CHARACTERISTICS				Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended			
Param.	Symbol	Characteristic		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 Fall Time	100 kHz mode	—	300	ns	Cb is specified to be from 10 to 400 pF
			400 kHz mode	20 + 0.1 Cb	300	ns	
			1 MHz mode <sup>(1)</sup>	—	100	ns	
IS21	TR:SCL	SDAx and SCLx Rise Time	100 kHz mode	—	1000	ns	Cb is specified to be from 10 to 400 pF
			400 kHz mode	20 + 0.1 Cb	300	ns	
			1 MHz mode <sup>(1)</sup>	—	300	ns	
IS25	TSU:DAT	Data Input Setup Time	100 kHz mode	250	—	ns	—
			400 kHz mode	100	—	ns	
			1 MHz mode <sup>(1)</sup>	100	—	ns	
IS26	THD:DAT	Data Input Hold Time	100 kHz mode	0	—	μs	—
			400 kHz mode	0	0.9	μs	
			1 MHz mode <sup>(1)</sup>	0	0.3	μs	
IS30	TSU:STA	Start Condition Setup Time	100 kHz mode	4.7	—	μs	Only relevant for Repeated Start condition
			400 kHz mode	0.6	—	μs	
			1 MHz mode <sup>(1)</sup>	0.25	—	μs	
IS31	THD:STA	Start Condition Hold Time	100 kHz mode	4.0	—	μs	After this period, the first clock pulse is generated
			400 kHz mode	0.6	—	μs	
			1 MHz mode <sup>(1)</sup>	0.25	—	μs	
IS33	TSU:STO	Stop Condition Setup Time	100 kHz mode	4.7	—	μs	—
			400 kHz mode	0.6	—	μs	
			1 MHz mode <sup>(1)</sup>	0.6	—	μs	
IS34	THD:STO	Stop Condition Hold Time	100 kHz mode	4000	—	ns	—
			400 kHz mode	600	—	ns	
			1 MHz mode <sup>(1)</sup>	250	—	ns	
IS40	TAA:SCL	Output Valid From Clock	100 kHz mode	0	3500	ns	—
			400 kHz mode	0	1000	ns	
			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 before a new transmission can start
			400 kHz mode	1.3	—	μs	
			1 MHz mode <sup>(1)</sup>	0.5	—	μs	
IS50	CB	Bus Capacitive Loading		—	400	pF	—

**Note 1:** Maximum pin capacitance = 10 pF for all I2Cx pins (for 1 MHz mode only).

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

**TABLE 27-40: 10-BIT HIGH-SPEED ADC MODULE SPECIFICATIONS**

AC CHARACTERISTICS			Standard Operating Conditions (see Note 2): 3.0V and 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended				
Param No.	Symbol	Characteristic	Min	Typ	Max	Units	Conditions
<b>Device Supply</b>							
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	—	VSS + 0.3	V	—
<b>Analog Input</b>							
AD10	V <sub>INH-V<sub>INL</sub></sub>	Full-Scale Input Span	VSS	—	VDD	V	—
AD11	V <sub>IN</sub>	Absolute Input Voltage	AVSS	—	AVDD	V	—
AD12	I <sub>AD</sub>	Operating Current	—	8	—	mA	—
AD13	—	Leakage Current	—	±0.6	—	µA	V <sub>INL</sub> = AVSS = 0V, AVDD = 3.3V Source Impedance = 100Ω
AD17	R <sub>IN</sub>	Recommended Impedance Of Analog Voltage Source	—	—	100	Ω	—
<b>DC Accuracy</b>							
AD20	N <sub>r</sub>	Resolution	10 data bits			bits	—
AD21A	INL	Integral Nonlinearity	> -2	±0.5	< 2	LSb	V <sub>INL</sub> = AVSS = 0V, AVDD = 3.3V
AD22A	DNL	Differential Nonlinearity	> -1	±0.5	< 1	LSb	V <sub>INL</sub> = AVSS = 0V, AVDD = 3.3V
AD23A	GERR	Gain Error	> -5	±2.0	< 5	LSb	V <sub>INL</sub> = AVSS = 0V, AVDD = 3.3V
AD24A	E <sub>OFF</sub>	Offset Error	> -3	±0.75	< 3	LSb	V <sub>INL</sub> = AVSS = VSS = 0V, AVDD = VDD = 3.3V
AD25	—	Monotonicity <sup>(1)</sup>	—	—	—	—	Guaranteed
<b>Dynamic Performance</b>							
AD30	THD	Total Harmonic Distortion	—	-73	—	dB	—
AD31	SINAD	Signal to Noise and Distortion	—	58	—	dB	—
AD32	SFDR	Spurious Free Dynamic Range	—	-73	—	dB	—
AD33	F <sub>NYQ</sub>	Input Signal Bandwidth	—	—	1	MHz	—
AD34	ENOB	Effective Number of Bits	—	9.4	—	bits	—

**Note 1:** The analog-to-digital conversion result never decreases with an increase in the input voltage, and has no missing codes.

**2:** Overall functional device operation at  $V_{BOR} < V_{DD} < V_{DDMIN}$  is guaranteed but not characterized. All device analog modules such as the ADC, etc., will function but with degraded performance below  $V_{DDMIN}$ .

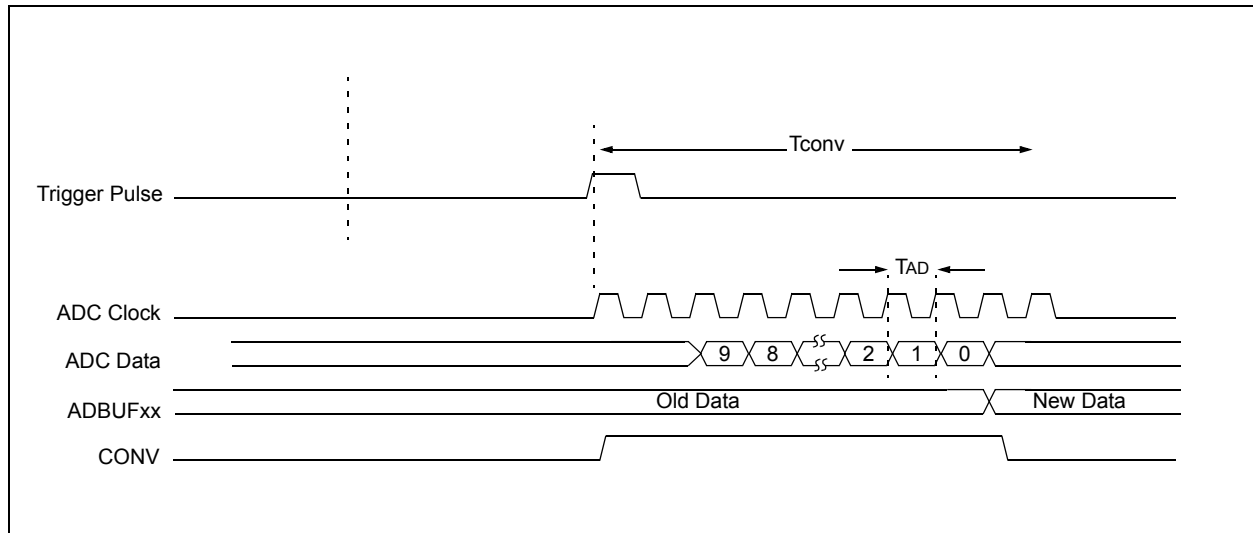
TABLE 27-41: 10-BIT HIGH-SPEED ADC MODULE TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions (see Note 2): 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended				
Param No.	Symbol	Characteristic	Min	Typ <sup>(1)</sup>	Max	Units	Conditions
<b>Clock Parameters</b>							
AD50b	TAD	ADC Clock Period	35.8	—	—	ns	—
<b>Conversion Rate</b>							
AD55b	tCONV	Conversion Time	—	14 TAD	—	—	—
AD56b	FCNV	Throughput Rate	—	—	2.0	Msp/s	—
		Devices with Single SAR	—	—	4.0	Msp/s	—
<b>Timing Parameters</b>							
AD63b	tDPU	Time to Stabilize Analog Stage from ADC Off to ADC On <sup>(1)</sup>	1.0	—	10	μs	—

**Note 1:** These parameters are characterized but not tested in manufacturing.

**Note 2:** Overall functional device operation at  $V_{BOR} < V_{DD} < V_{DDMIN}$  is guaranteed but not characterized. All device analog modules such as the ADC, etc., will function but with degraded performance below  $V_{DDMIN}$ .

FIGURE 27-23: ANALOG-TO-DIGITAL CONVERSION TIMING PER INPUT



# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

**TABLE 27-42: COMPARATOR MODULE SPECIFICATIONS**

AC and DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) Operating temperature: -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended				
Param. No.	Symbol	Characteristic	Min	Typ	Max	Units	Comments
CM10	VIOFF	Input Offset Voltage		±5	±15	mV	—
CM11	VICM	Input Common Mode Voltage Range <sup>(1)</sup>	0	—	AVDD – 1.5	V	—
CM12	VGAIN	Open Loop Gain <sup>(1)</sup>	90	—	—	db	—
CM13	CMRR	Common Mode Rejection Ratio <sup>(1)</sup>	70	—	—	db	—
CM14	TRESP	Large Signal Response		20	30	ns	V+ input step of 100 mv while V- input held at AVDD/2. Delay measured from analog input pin to PWM output pin.

**Note 1:** Parameters are for design guidance only and are not tested in manufacturing.

**TABLE 27-43: DAC MODULE SPECIFICATIONS**

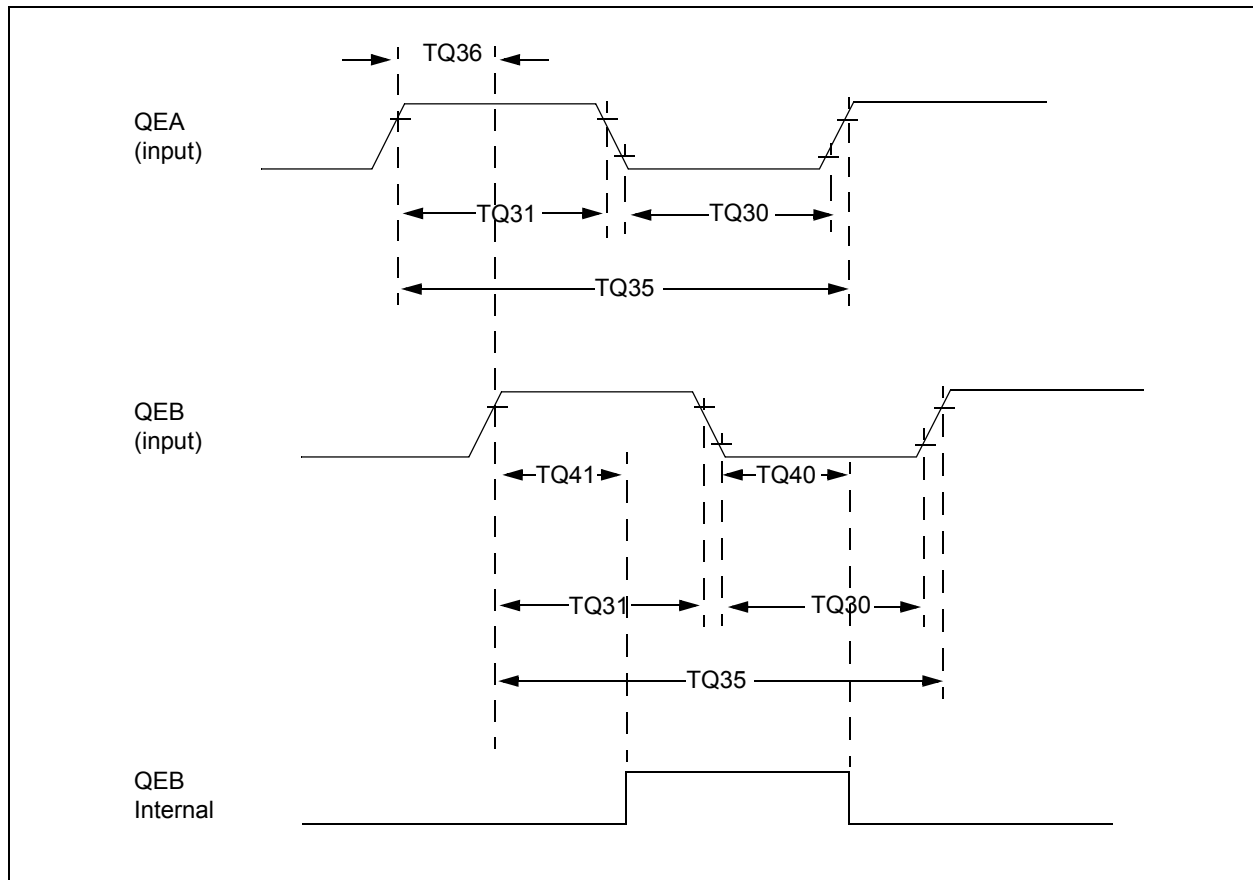
AC and DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) Operating temperature: -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended				
Param. No.	Symbol	Characteristic	Min	Typ	Max	Units	Comments
DA01	EXTREF	External Reference Voltage <sup>(1)</sup>	0	—	AVDD – 1.6	V	—
DA08	INTREF	Internal Reference Voltage <sup>(1)</sup>	1.25	1.32	1.41	V	—
DA02	CVRES	Resolution	10 data bits			bits	—
DA03	INL	Integral Nonlinearity Error	—	±1.0	—	—	AVDD = 3.3V, DACREF = (AVDD/2)V
DA04	DNL	Differential Nonlinearity Error	—	±0.8	—	LSB	—
DA05	E0FF	Offset Error	—	±2.0	—	LSB	—
DA06	EG	Gain Error	—	±2.0	—	LSB	—
DA07	TSET	Settling Time <sup>(1)</sup>	—	—	650	nsec	Measured when range = 1 (high range), and CMREF<9:0> transitions from 0x1FF to 0x300.

**Note 1:** Parameters are for design guidance only and are not tested in manufacturing.

TABLE 27-44: DAC OUTPUT BUFFER SPECIFICATIONS

DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) Operating temperature: $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended				
Param. No.	Symbol	Characteristic	Min	Typ	Max	Units	Comments
DA10	RLOAD	Resistive Output Load Impedance	3K	—	—	$\Omega$	—
DA11	CLOAD	Output Load Capacitance	—	20	35	pF	—
DA12	IOUT	Output Current Drive Strength	200	300	400	$\mu\text{A}$	Sink and source
DA13	VRANGE	Full Output Drive Strength Voltage Range	$\text{AVSS} + 250 \text{ mV}$	—	$\text{AVDD} - 900 \text{ mV}$	V	—
DA14	VLRANGE	Output Drive Voltage Range at Reduced Current Drive of $50 \mu\text{A}$	$\text{AVSS} + 50 \text{ mV}$	—	$\text{AVDD} - 500 \text{ mV}$	V	—
DA15	IDD	Current Consumed when Module is Enabled, High-Power Mode	—	—	$1.3 \times \text{IOUT}$	$\mu\text{A}$	Module will always consume this current even if no load is connected to the output
DA16	ROUTON	Output Impedance when Module is Enabled	—	500	—	$\Omega$	—

FIGURE 27-24: QEA/QEB INPUT CHARACTERISTICS

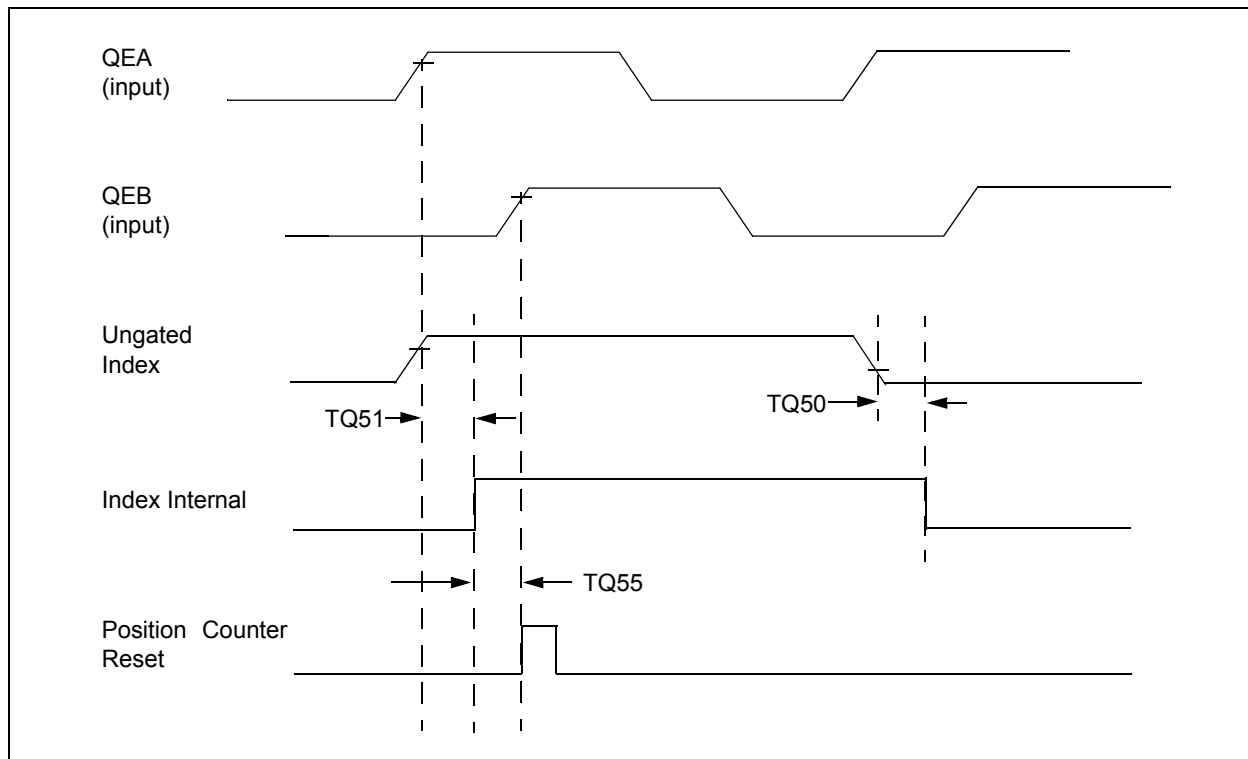


**TABLE 27-45: QUADRATURE DECODER TIMING REQUIREMENTS**

AC CHARACTERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	Symbol	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 Low, with Digital Filter	3 * N * TcY	—	ns	N = 1, 2, 4, 16, 32, 64, 128 and 256 ( <b>Note 3</b> )
TQ41	TQuFH	Filter Time to Recognize High, with Digital Filter	3 * N * TcY	—	ns	N = 1, 2, 4, 16, 32, 64, 128 and 256 ( <b>Note 3</b> )

- Note 1:** These parameters are characterized but not tested in manufacturing.  
**Note 2:** Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.  
**Note 3:** 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 27-25: QEI MODULE INDEX PULSE TIMING CHARACTERISTICS**

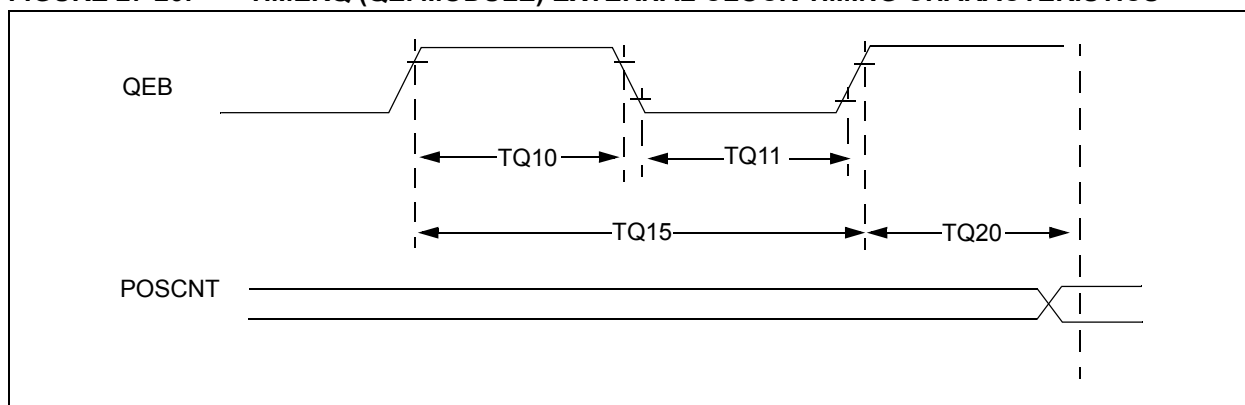


**TABLE 27-46: QEI INDEX PULSE TIMING REQUIREMENTS**

AC CHARACTERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended				
Param No.	Symbol	Characteristic <sup>(1)</sup>	Min	Max	Units	Conditions
TQ50	TqIL	Filter Time to Recognize Low, with Digital Filter	$3 * N * T_{CY}$	—	ns	N = 1, 2, 4, 16, 32, 64, 128 and 256 (Note 2)
TQ51	TqiH	Filter Time to Recognize High, with Digital Filter	$3 * N * T_{CY}$	—	ns	N = 1, 2, 4, 16, 32, 64, 128 and 256 (Note 2)
TQ55	Tqidxr	Index Pulse Recognized to Position Counter Reset (ungated index)	$3 T_{CY}$	—	ns	—

- Note 1:** These parameters are characterized but not tested in manufacturing.
- Note 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 27-26: TIMERQ (QEI MODULE) EXTERNAL CLOCK TIMING CHARACTERISTICS**

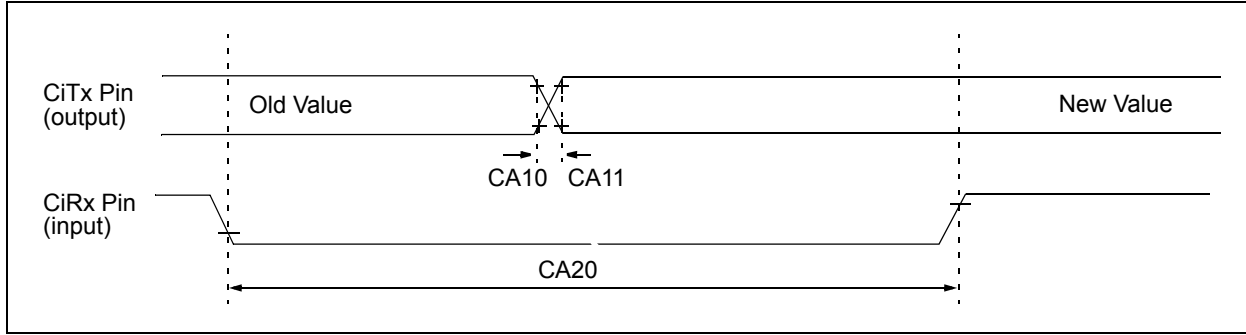


**TABLE 27-47: QEI MODULE EXTERNAL CLOCK TIMING REQUIREMENTS**

AC CHARACTERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended						
Param No.	Symbol	Characteristic <sup>(1)</sup>		Min	Typ	Max	Units	Conditions
TQ10	TtQH	TQCK High Time	Synchronous, with prescaler	$T_{CY} + 20$	—	—	ns	Must also meet parameter TQ15
TQ11	TtQL	TQCK Low Time	Synchronous, with prescaler	$T_{CY} + 20$	—	—	ns	Must also meet parameter TQ15
TQ15	TtQP	TQCP Input Period	Synchronous, with prescaler	$2 * T_{CY} + 40$	—	—	ns	—
TQ20	TCKEXTMRL	Delay from External TxCK Clock Edge to Timer Increment		$0.5 T_{CY}$	—	$1.5 T_{CY}$	—	—

- Note 1:** These parameters are characterized but not tested in manufacturing.

**FIGURE 27-27: CAN MODULE I/O TIMING CHARACTERISTICS**



**TABLE 27-48: ECAN™ MODULE I/O TIMING REQUIREMENTS**

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended				
Param No.	Symbol	Characteristic <sup>(1)</sup>	Min	Typ	Max	Units	Conditions
CA10	TioF	Port Output Fall Time	—	—	—	ns	See parameter <a href="#">DO32</a>
CA11	TioR	Port Output Rise Time	—	—	—	ns	See parameter <a href="#">DO31</a>
CA20	Tcwf	Pulse Width to Trigger CAN Wake-up Filter	120	—	—	ns	—

**Note 1:** These parameters are characterized but not tested in manufacturing.

**TABLE 27-49: DMA READ/WRITE TIMING REQUIREMENTS**

AC CHARACTERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended				
Param No.	Characteristic	Min.	Typ	Max.	Units	Conditions
DM1	DMA Read/Write Cycle Time	—	—	1 T <sub>CY</sub>	ns	—



## 28.0 50 MIPS ELECTRICAL CHARACTERISTICS

This section provides an overview of dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 electrical characteristics for devices operating at 50 MIPS.

Specifications are identical to those shown in [Section 27.0 “Electrical Characteristics”](#), with the exception of the parameters listed in this section.

Absolute maximum ratings for the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 50 MIPS 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 .....	-40°C to +85°C
Storage temperature .....	-65°C to +150°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>(2)</sup> .....	-0.3V to (VDD + 0.3V)
Voltage on any 5V tolerant pin with respect to VSS, when VDD ≥ 3.0V <sup>(2)</sup> .....	-0.3V to +5.6V
Voltage on any 5V tolerant pin with respect to VSS, when VDD < 3.0V <sup>(2)</sup> .....	-0.3V to (VDD + 0.3V)
Maximum current out of VSS pin .....	300 mA
Maximum current into VDD pin <sup>(2)</sup> .....	250 mA
Maximum current sourced/sunk by any 4x I/O pin .....	15 mA
Maximum current sourced/sunk by any 8x I/O pin .....	25 mA
Maximum current sourced/sunk by any 16x I/O pin .....	45 mA
Maximum current sunk by all ports .....	200 mA
Maximum current sourced by all ports <sup>(2)</sup> .....	200mA

**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:** See the “[Pin Diagrams](#)” section for 5V tolerant pins.

## 28.1 DC Characteristics

**TABLE 28-1: OPERATING MIPS VS. VOLTAGE**

Characteristic	VDD Range (in Volts)	Temp Range (in °C)	Max MIPS
			dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610
—	3.0-3.6V <sup>(1)</sup>	-40°C to +85°C	50

**Note 1:** Overall functional device operation at  $V_{BORMIN} < V_{DD} < V_{DDMIN}$  is tested but not characterized. All device analog modules such as the ADC, etc., will function but with degraded performance below  $V_{DDMIN}$ . See parameter BO10 in [Table 27-11](#) for the BOR values.

**TABLE 28-2: DC CHARACTERISTICS: OPERATING CURRENT (IDD)**

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial			
Parameter No.	Typical	Max	Units	Conditions		
<b>Operating Current (IDD)<sup>(1)</sup></b>						
MDC29d	85	100	mA	-40°C	3.3V	50 MIPS See <b>Note 1</b>
MDC29a	85	100	mA	+25°C		
MDC29b	85	100	mA	+85°C		

**Note 1:**  $I_{DD}$  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  $I_{DD}$  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)
- CLKO is configured as an I/O input pin in the Configuration word
- All I/O pins are configured as inputs and pulled to  $V_{SS}$
- $\overline{\text{MCLR}} = V_{DD}$ , 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 (all PMDx bits are zeroed)
- CPU executing `while(1)` statement
- JTAG disabled

**TABLE 28-3: DC CHARACTERISTICS: IDLE CURRENT (IIDLE)**

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial			
Parameter No.	Typical	Max	Units	Conditions		
<b>Idle Current (IIDLE): Core Off Clock On Base Current<sup>(1)</sup></b>						
MDC45d	40	50	mA	-40°C	3.3V	50 MIPS
MDC45a	40	50	mA	+25°C		
MDC45b	40	50	mA	+85°C		

**Note 1:** Base Idle current (IIDLE) 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 (all PMDx bits are zeroed)
- The NVMSIDL bit (NVMCON<12>) = 1 (i.e., Flash regulator is set to stand-by while the device is in Idle mode)
- JTAG disabled

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

**TABLE 28-4: DC CHARACTERISTICS: DOZE CURRENT (IDOZE)**

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial		
Parameter No.	Typical	Max	Doze Ratio	Units	Conditions
MDC74a	49	70	1:2	mA	-40°C    3.3V    50 MIPS
MDC74f	43	70	1:64	mA	
MDC74g	43	70	1:128	mA	
MDC75a	47	70	1:2	mA	+25°C    3.3V    50 MIPS
MDC75f	41	70	1:64	mA	
MDC75g	41	70	1:128	mA	
MDC76a	46	70	1:2	mA	+85°C    3.3V    50 MIPS
MDC76f	40	70	1:64	mA	
MDC76g	40	70	1:128	mA	

**Note 1:** 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 (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
- $\overline{\text{MCLR}} = V_{\text{DD}}$ , 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 (all PMDx bits are zeroed)
- CPU executing `while(1)` statement
- JTAG disabled

## 28.2 AC Characteristics and Timing Parameters

This section defines dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 AC characteristics and timing parameters for 50 MIPS devices.

**TABLE 28-5: EXTERNAL CLOCK TIMING REQUIREMENTS**

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial				
Param No.	Symb	Characteristic	Min	Typ <sup>(1)</sup>	Max	Units	Conditions
MOS10	FIN	External CLKI Frequency (External clocks allowed only in EC and ECPLL modes)	DC	—	50	MHz	EC
		Oscillator Crystal Frequency	3.5 — 10	— — —	10 33 50	MHz kHz MHz	XT Sosc HS
MOS20	Tosc	$T_{osc} = 1/F_{osc}$	10	—	DC	ns	—
MOS25	Tcy	Instruction Cycle Time <sup>(2)</sup>	20	—	DC	ns	—
MOS30	TosL, TosH	External Clock in (OSC1) High or Low Time	$0.375 \times T_{osc}$	—	$0.625 \times T_{osc}$	ns	EC
MOS31	TosR, TosF	External Clock in (OSC1) Rise or Fall Time	—	—	20	ns	EC
MOS40	TckR	CLKO Rise Time <sup>(3)</sup>	—	5.2	—	ns	—
MOS41	TckF	CLKO Fall Time <sup>(3)</sup>	—	5.2	—	ns	—
MOS41	Gm	External Oscillator Transconductance	14	16	18	mA/V	$V_{DD} = 3.3V,$ $T_A = +25^{\circ}\text{C}$

**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.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

**TABLE 28-6: TIMER1 EXTERNAL CLOCK TIMING REQUIREMENTS<sup>(1)</sup>**

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial					
Param No.	Symbol	Characteristic	Min	Typ	Max	Units	Conditions	
MTA10	T <sub>TxH</sub>	TxCK High Time	Synchronous, no prescaler	$T_{CY} + 15$	—	—	ns	Must also meet parameter TA15. N = prescale value (1, 8, 64, 256)
			Synchronous, with prescaler	$(T_{CY} + 15)/N$	—	—	ns	
			Asynchronous	15	—	—	ns	
MTA11	T <sub>TxL</sub>	TxCK Low Time	Synchronous, no prescaler	$(T_{CY} + 15)$	—	—	ns	Must also meet parameter TA15. N = prescale value (1, 8, 64, 256)
			Synchronous, with prescaler	$(T_{CY} + 20)/N$	—	—	ns	
			Asynchronous	15	—	—	ns	
MTA15	T <sub>TxP</sub>	TxCK Input Period	Synchronous, no prescaler	$2 T_{CY} + 30$	—	—	ns	—
			Synchronous, with prescaler	Greater of: 40 ns or $(2 T_{CY} + 30)/N$	—	—	—	N = prescale value (1, 8, 64, 256)
			Asynchronous	30	—	—	ns	—
MOS60	F <sub>t1</sub>	SOSCI/T1CK Oscillator Input frequency Range (oscillator enabled by setting bit TCS (T1CON<1>))	DC	—	50	kHz	—	
MTA20	T <sub>CKEXTMRL</sub>	Delay from External TxCK Clock Edge to Timer Increment	$0.75 T_{CY} + 30$	—	$1.75 T_{CY} + 30$	—	—	

**Note 1:** Timer1 is a Type A.

**TABLE 28-7: TIMER2, TIMER4 EXTERNAL CLOCK TIMING REQUIREMENTS**

AC CHARACTERISTICS				Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial				
Param No.	Symbol	Characteristic <sup>(1)</sup>		Min	Typ	Max	Units	Conditions
MTB10	TtxH	TxCK High Time	Synchronous mode	Greater of: 15 or (Tcy + 15)/N	—	—	ns	Must also meet parameter TB15 N = prescale value (1, 8, 64, 256)
MTB11	TtxL	TxCK Low Time	Synchronous mode	Greater of: 15 or (Tcy + 15)/N	—	—	ns	Must also meet parameter TB15 N = prescale value (1, 8, 64, 256)
MTB15	TtxP	TxCK Input Period	Synchronous mode	Greater of: 40 or (2 Tcy + 30)/N	—	—	ns	N = prescale value (1, 8, 64, 256)
MTB20	TCKEXTMRL	Delay from External TxCK Clock Edge to Timer Increment		0.75 Tcy + 30	—	1.75 Tcy + 30	ns	—

**Note 1:** These parameters are characterized, but are not tested in manufacturing.

**TABLE 28-8: TIMER3, TIMER5 EXTERNAL CLOCK TIMING REQUIREMENTS**

AC CHARACTERISTICS				Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial				
Param No.	Symbol	Characteristic <sup>(1)</sup>		Min	Typ	Max	Units	Conditions
MTC10	TtxH	TxCK High Time	Synchronous	Tcy + 10	—	—	ns	Must also meet parameter TC15
MTC11	TtxL	TxCK Low Time	Synchronous	Tcy + 10	—	—	ns	Must also meet parameter TC15
MTC15	TtxP	TxCK Input Period	Synchronous, with prescaler	2 Tcy + 20	—	—	ns	—
MTC20	TCKEXTMRL	Delay from External TxCK Clock Edge to Timer Increment		0.75 Tcy + 20	—	1.75 Tcy + 20	ns	—

**Note 1:** These parameters are characterized, but are not tested in manufacturing.

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

TABLE 28-9: SIMPLE OC/PWM MODE TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial				
Param No.	Symbol	Characteristic <sup>(1)</sup>	Min	Typ	Max	Units	Conditions
MOC15	TFD	Fault Input to PWM I/O Change	—	—	T <sub>CY</sub> + 10	ns	—
MOC20	TFLT	Fault Input Pulse Width	T <sub>CY</sub> + 10	—	—	ns	—

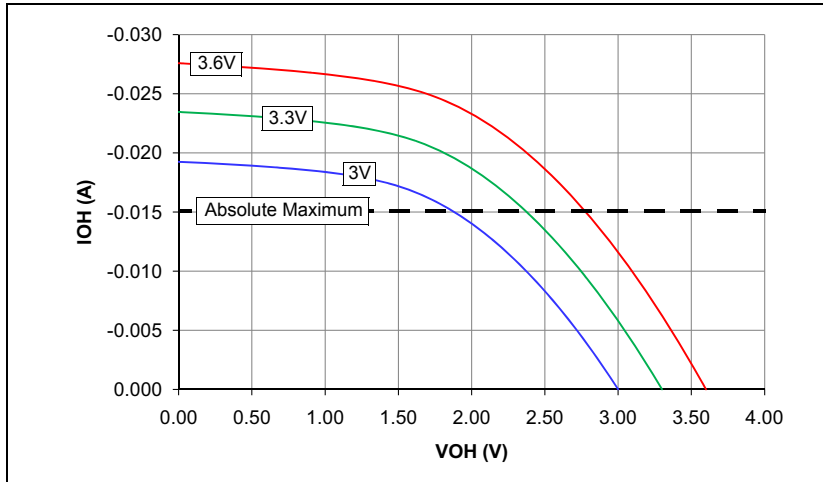
**Note 1:** These parameters are characterized but not tested in manufacturing.



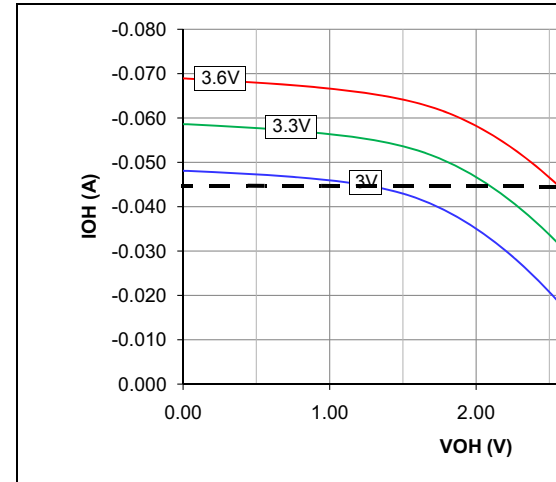
## 29.0 DC AND AC DEVICE CHARACTERISTICS GRAPHS

**Note:** The graphs provided following this note are a statistical summary based on a limited number of samples and are provided for only. The performance characteristics listed herein are not tested or guaranteed. In some graphs, the data presented may be outside range (e.g., outside specified power supply range) and therefore, outside the warranted range.

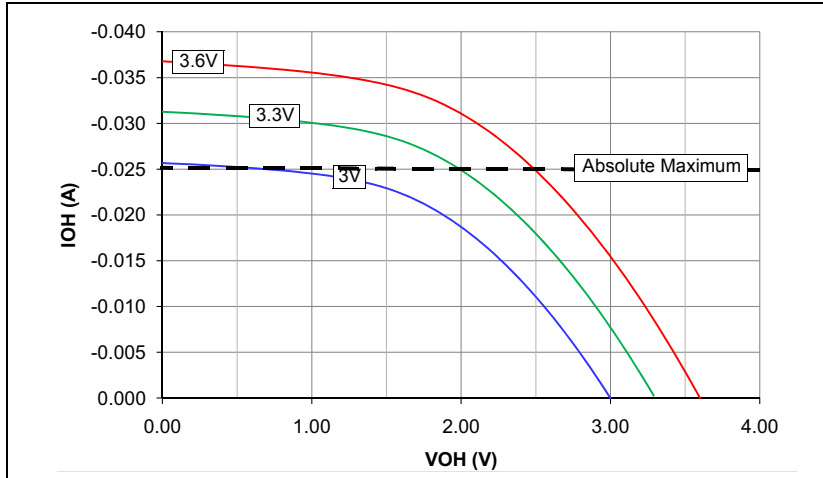
**FIGURE 29-1: VOH – 4x DRIVER PINS**



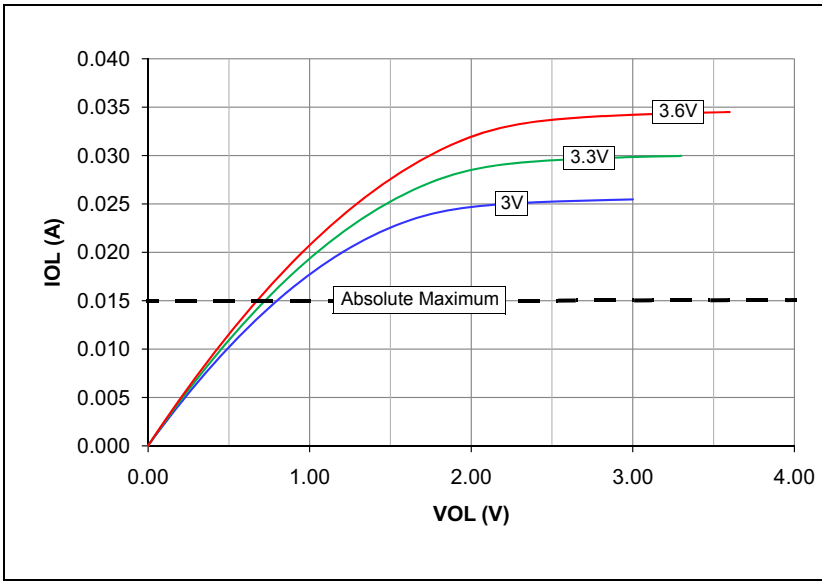
**FIGURE 29-3: VOH – 16x DRIVER PINS**



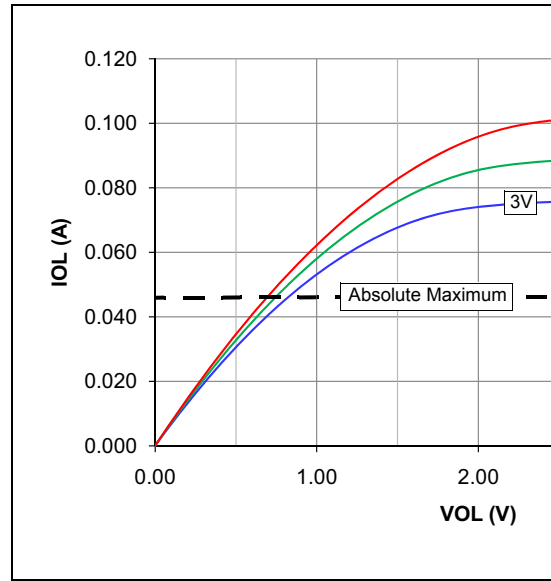
**FIGURE 29-2: VOH – 8x DRIVER PINS**



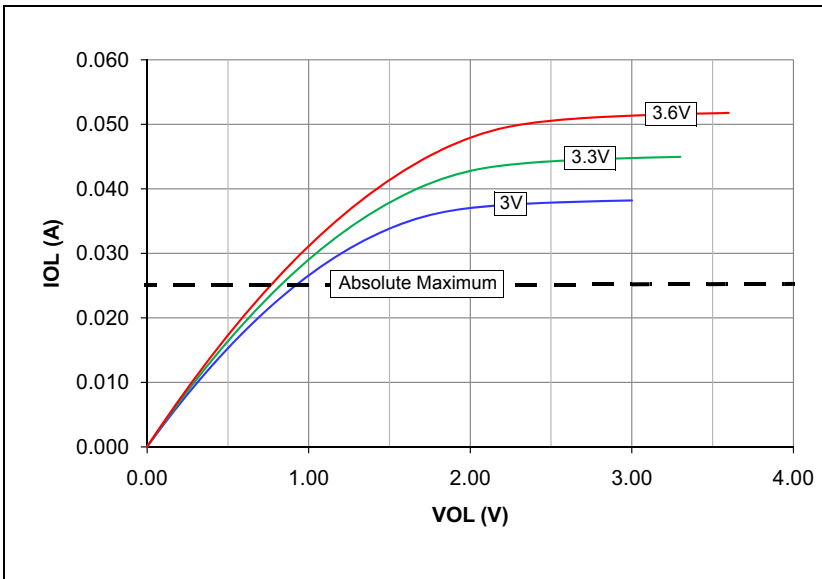
**FIGURE 29-4: VOL – 4x DRIVER PINS**



**FIGURE 29-6: VOL – 16x DRIVER PINS**



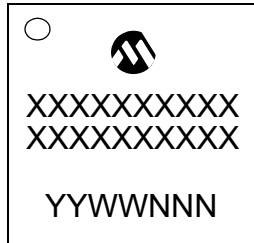
**FIGURE 29-5: VOL – 8x DRIVER PINS**



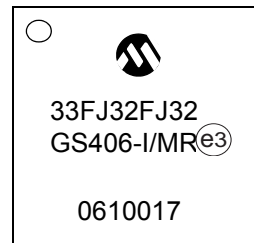
**30.0 PACKAGING INFORMATION**

**30.1 Package Marking Information**

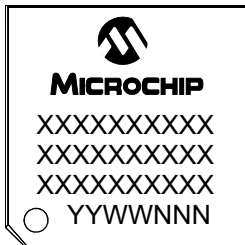
64-Lead QFN (9x9x0.9mm)



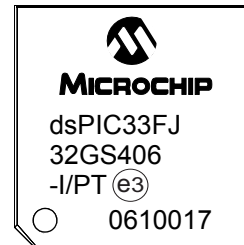
Example



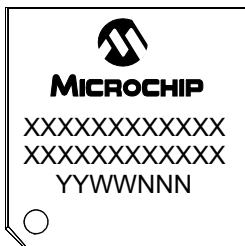
64-Lead TQFP (10x10x1mm)



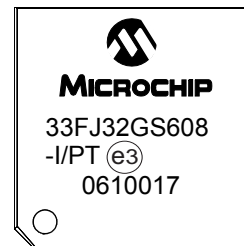
Example



80-Lead TQFP (12x12x1mm)



Example

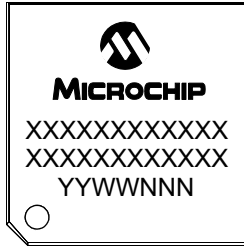


<b>Legend:</b>	XX...X	Customer-specific information
	Y	Year code (last digit of calendar year)
	YY	Year code (last 2 digits of calendar year)
	WW	Week code (week of January 1 is week '01')
	NNN	Alphanumeric traceability code
	(e3)	Pb-free JEDEC designator for Matte Tin (Sn)
	*	This package is Pb-free. The Pb-free JEDEC designator ((e3)) can be found on the outer packaging for this package.

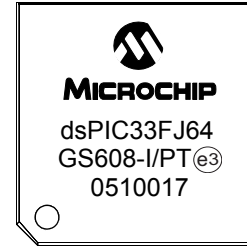
**Note:** If the full Microchip part number cannot be marked on one line, it is carried over to the next line, thus limiting the number of available characters for customer-specific information.

**30.1 Package Marking Information (Continued)**

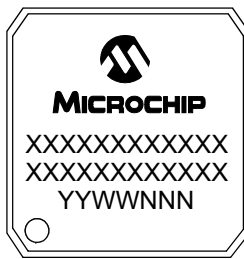
100-Lead TQFP (12x12x1 mm)



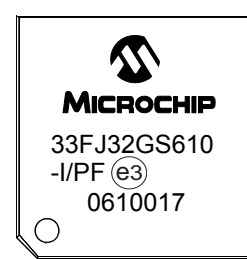
Example



100-Lead TQFP (14x14x1mm)



Example

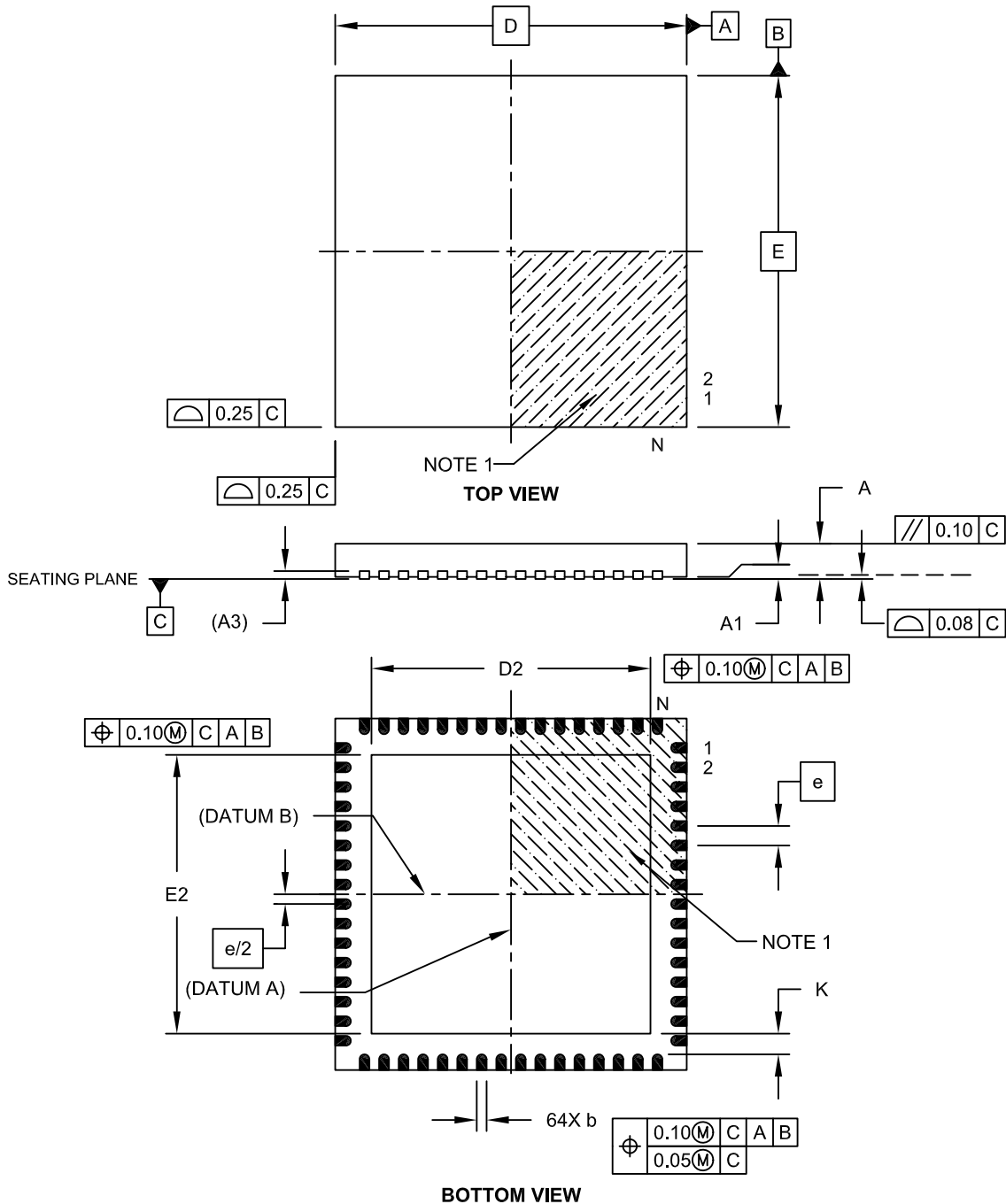


<b>Legend:</b>	XX...X	Customer-specific information
	Y	Year code (last digit of calendar year)
	YY	Year code (last 2 digits of calendar year)
	WW	Week code (week of January 1 is week '01')
	NNN	Alphanumeric traceability code
	(e3)	Pb-free JEDEC designator for Matte Tin (Sn)
	* (e3)	This package is Pb-free. The Pb-free JEDEC designator ((e3)) can be found on the outer packaging for this package.
<b>Note:</b> If the full Microchip part number cannot be marked on one line, it is carried over to the next line, thus limiting the number of available characters for customer-specific information.		

30.2 Package Details

64-Lead Plastic Quad Flat, No Lead Package (MR) – 9x9x0.9 mm Body [QFN]  
With 7.15 x 7.15 Exposed Pad [QFN]

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

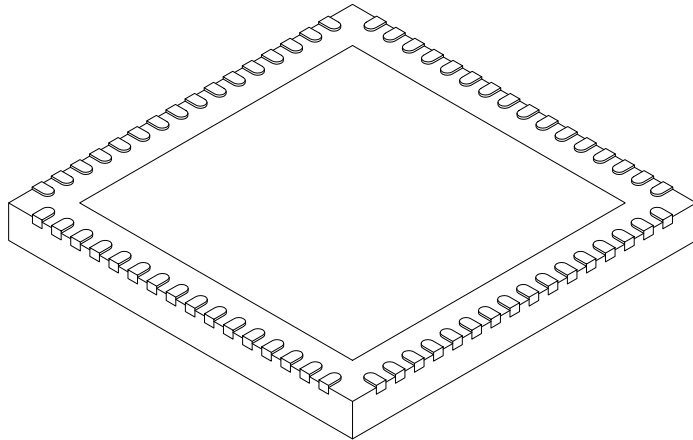


Microchip Technology Drawing C04-149C Sheet 1 of 2

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## 64-Lead Plastic Quad Flat, No Lead Package (MR) – 9x9x0.9 mm Body [QFN] With 7.15 x 7.15 Exposed Pad [QFN]

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



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Number of Pins	N	64		
Pitch	e	0.50 BSC		
Overall Height	A	0.80	0.90	1.00
Standoff	A1	0.00	0.02	0.05
Contact Thickness	A3	0.20 REF		
Overall Width	E	9.00 BSC		
Exposed Pad Width	E2	7.05	7.15	7.50
Overall Length	D	9.00 BSC		
Exposed Pad Length	D2	7.05	7.15	7.50
Contact Width	b	0.18	0.25	0.30
Contact Length	L	0.30	0.40	0.50
Contact-to-Exposed Pad	K	0.20	-	-

**Notes:**

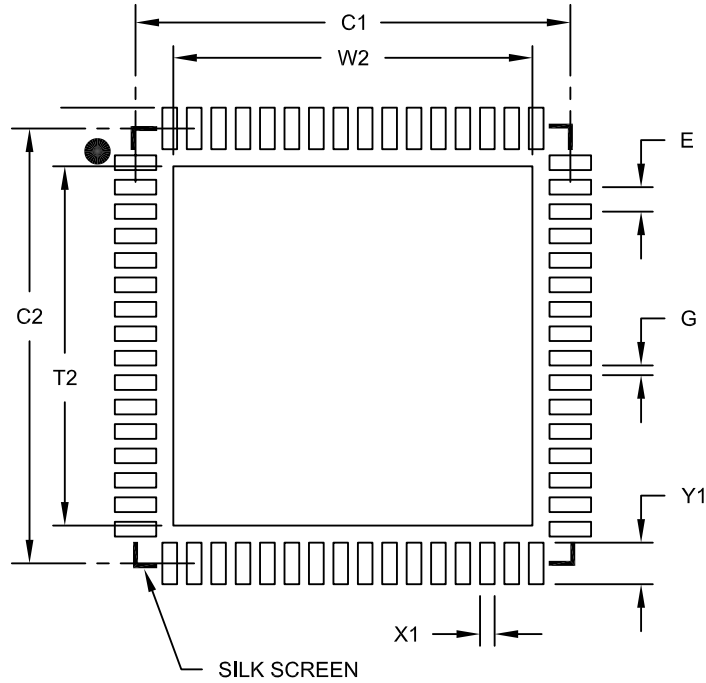
1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated.
3. Dimensioning and tolerancing per ASME Y14.5M.  
BSC: Basic Dimension. Theoretically exact value shown without tolerances.  
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-149C Sheet 2 of 2

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

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

Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Contact Pitch	E	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

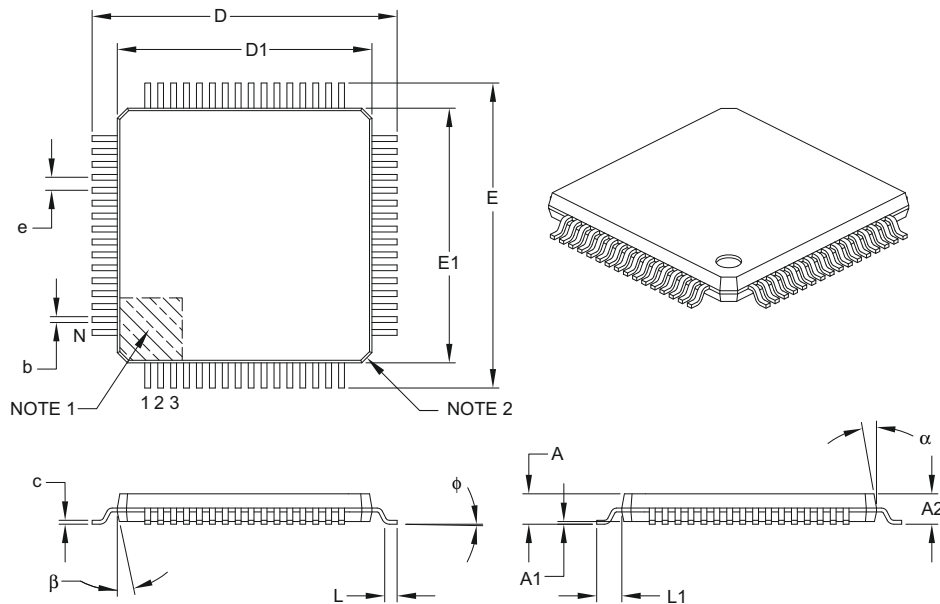
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2149A

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

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

**Notes:**

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

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

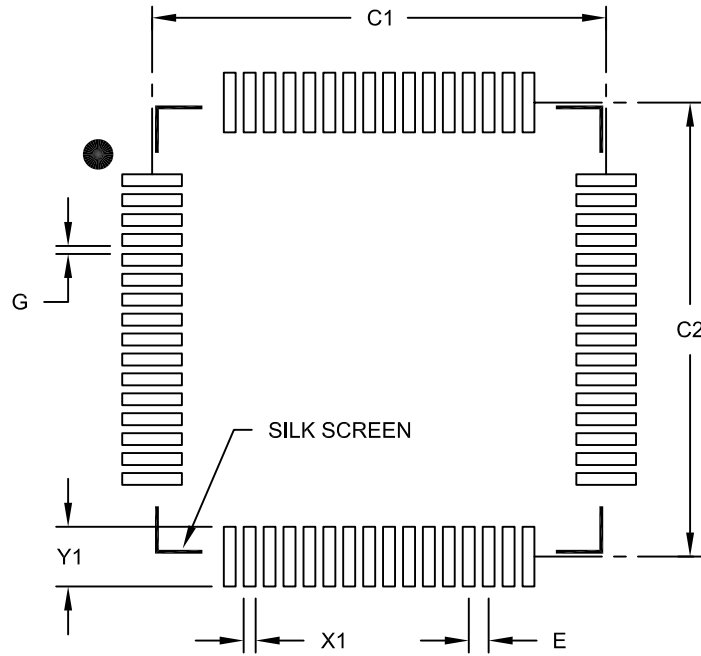
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-085B



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

Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Contact Pitch	E	0.50 BSC		
Contact Pad Spacing	C1		11.40	
Contact Pad Spacing	C2		11.40	
Contact Pad Width (X64)	X1			0.30
Contact Pad Length (X64)	Y1			1.50
Distance Between Pads	G	0.20		

Notes:

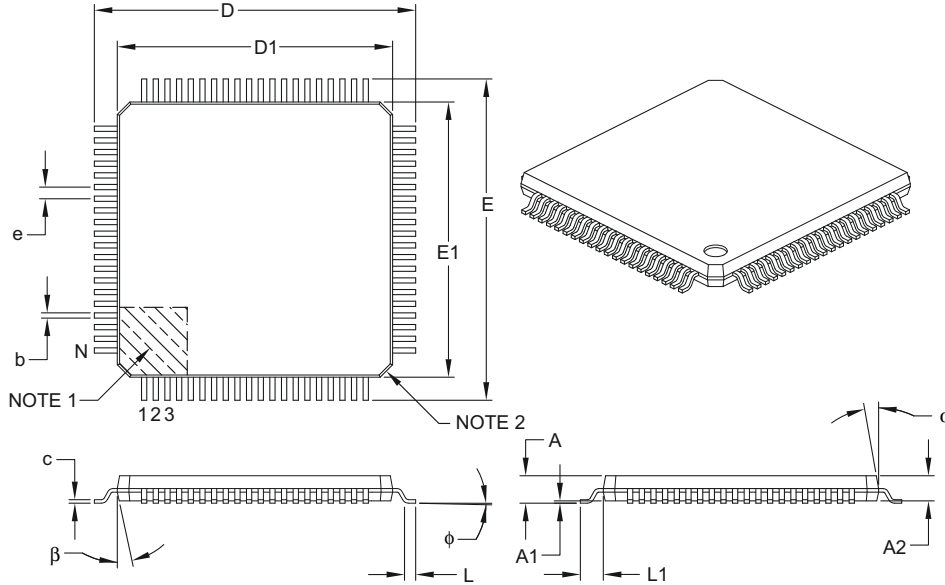
1. Dimensioning and tolerancing per ASME Y14.5M

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>



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Number of Leads	N	80		
Lead Pitch	e	0.50 BSC		
Overall Height	A	–	–	1.20
Molded Package Thickness	A2	0.95	1.00	1.05
Standoff	A1	0.05	–	0.15
Foot Length	L	0.45	0.60	0.75
Footprint	L1	1.00 REF		
Foot Angle	φ	0°	3.5°	7°
Overall Width	E	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	c	0.09	–	0.20
Lead Width	b	0.17	0.22	0.27
Mold Draft Angle Top	α	11°	12°	13°
Mold Draft Angle Bottom	β	11°	12°	13°

**Notes:**

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.

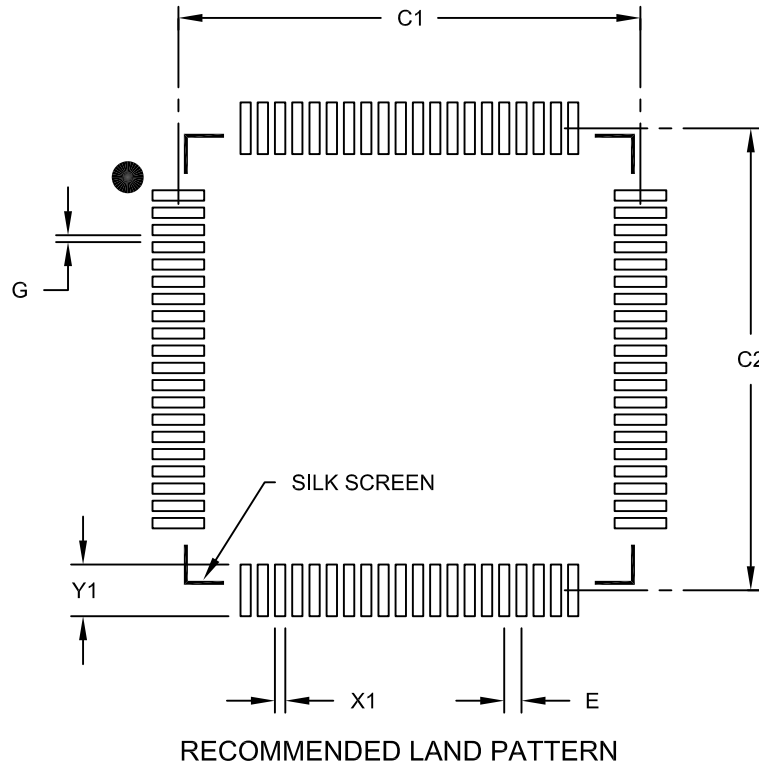
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

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>



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Contact Pitch	E		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:**

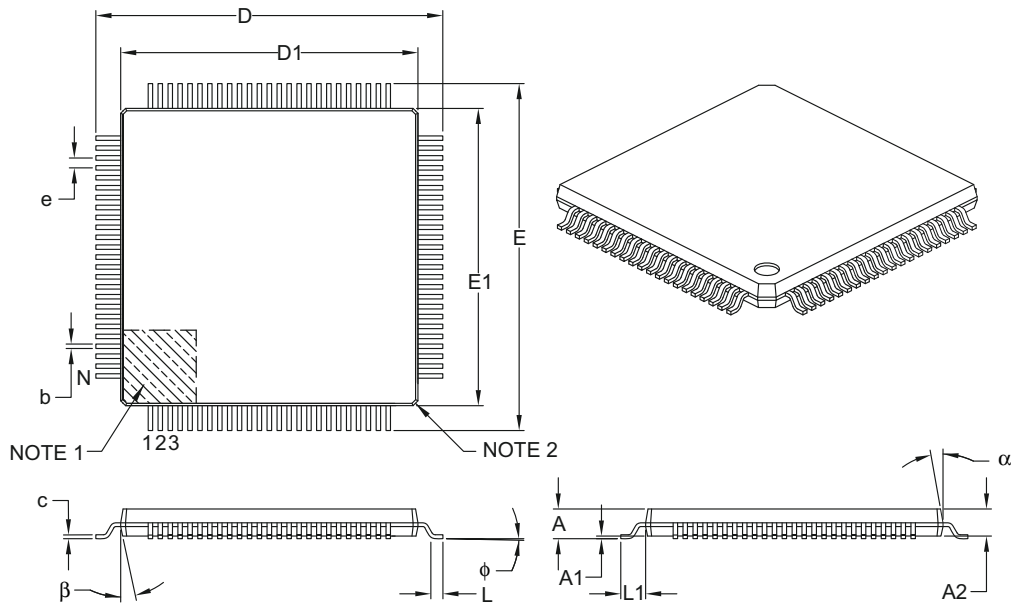
1. Dimensioning and tolerancing per ASME Y14.5M

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>



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Number of Leads	N	100		
Lead Pitch	e	0.40 BSC		
Overall Height	A	–	–	1.20
Molded Package Thickness	A2	0.95	1.00	1.05
Standoff	A1	0.05	–	0.15
Foot Length	L	0.45	0.60	0.75
Footprint	L1	1.00 REF		
Foot Angle	$\phi$	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	c	0.09	–	0.20
Lead Width	b	0.13	0.18	0.23
Mold Draft Angle Top	$\alpha$	11°	12°	13°
Mold Draft Angle Bottom	$\beta$	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.

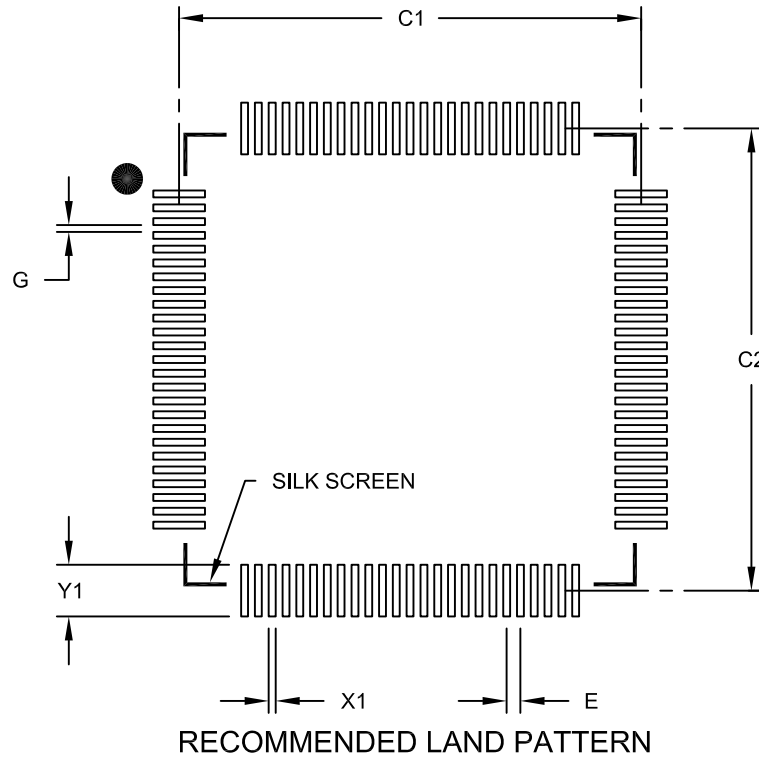
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-100B

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

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>



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

**Notes:**

1. Dimensioning and tolerancing per ASME Y14.5M

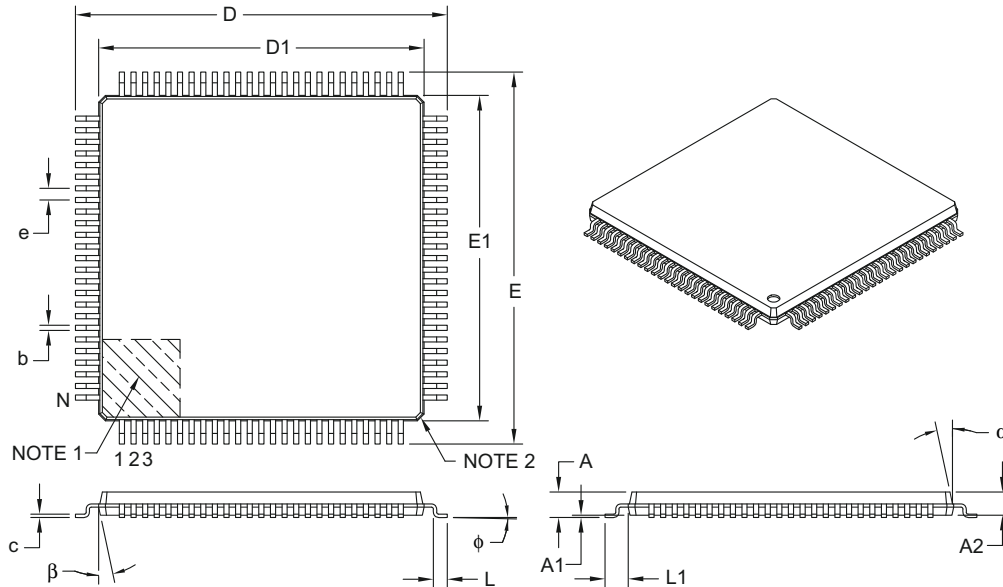
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2100B

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## 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>



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Number of Leads	N	100		
Lead Pitch	e	0.50 BSC		
Overall Height	A	–	–	1.20
Molded Package Thickness	A2	0.95	1.00	1.05
Standoff	A1	0.05	–	0.15
Foot Length	L	0.45	0.60	0.75
Footprint	L1	1.00 REF		
Foot Angle	$\phi$	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	c	0.09	–	0.20
Lead Width	b	0.17	0.22	0.27
Mold Draft Angle Top	$\alpha$	11°	12°	13°
Mold Draft Angle Bottom	$\beta$	11°	12°	13°

**Notes:**

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

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

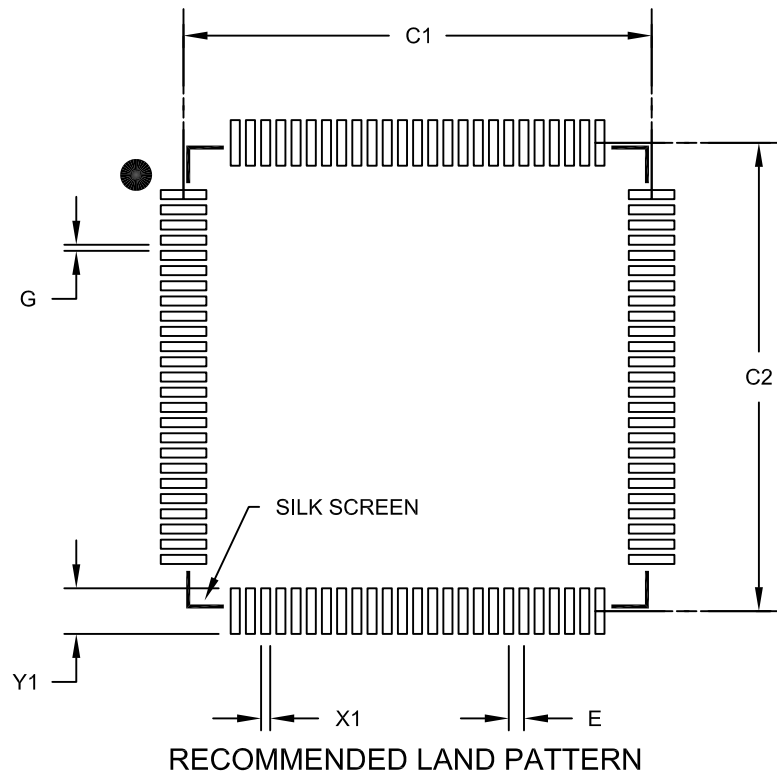
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-110B

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

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>



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Contact Pitch	E	0.50 BSC		
Contact Pad Spacing	C1		15.40	
Contact Pad Spacing	C2		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

NOTES:



## APPENDIX A: MIGRATING FROM dsPIC33FJ06GS101/X02 AND dsPIC33FJ16GSX02/X04 TO dsPIC33FJ32GS406/606/608/610 AND dsPIC33FJ64GS406/606/608/610 DEVICES

This appendix provides an overview of considerations for migrating from the dsPIC33FJ06GS101/X02 and dsPIC33FJ16GSX02/X04 family of devices to the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 family of devices. The code developed for the dsPIC33FJ06GS101/X02 and dsPIC33FJ16GSX02/X04 devices can be ported to the dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 devices after making the appropriate changes outlined below.

### A.1 Device Pins and Peripheral Pin Select (PPS)

On dsPIC33FJ06GS101/X02 and dsPIC33FJ16GSX02/X04 devices, some peripherals such as the Timer, Input Capture, Output Compare, UART, SPI, External Interrupts, Analog Comparator Output, as well as the PWM4 pin pair, were mapped to physical pins via Peripheral Pin Select (PPS) functionality. On dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 devices, these peripherals are hard-coded to dedicated pins. Because of this, as well as pinout differences between the two devices families, software must be updated to utilize peripherals on the desired pin locations.

### A.2 High-Speed PWM

#### A.2.1 FAULT AND CURRENT-LIMIT CONTROL SIGNAL SOURCE SELECTION

Fault and Current-Limit Control Signal Source selection has changed between the two families of devices. On dsPIC33FJ06GS101/X02 and dsPIC33FJ16GSX02/X04 devices, Fault1 through Fault8 were assigned to Fault and Current-Limit Controls with the following values:

- 00000 = Fault 1
- 00001 = Fault 2
- 00010 = Fault 3
- 00011 = Fault 4
- 00100 = Fault 5
- 00101 = Fault 6
- 00110 = Fault 7
- 00111 = Fault 8

On dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 devices, Fault1 through Fault8 were assigned to Fault and Current-Limit Controls with the following values:

- 01000 = Fault 1
- 01001 = Fault 2
- 01010 = Fault 3
- 01011 = Fault 4
- 01100 = Fault 5
- 01101 = Fault 6
- 01110 = Fault 7
- 01111 = Fault 8

#### A.2.2 ANALOG COMPARATORS CONNECTION

Connection of analog comparators to the PWM Fault and Current-Limit Control Signal Sources on dsPIC33FJ06GS101/X02 and dsPIC33FJ16GSX02/X04 devices is performed by assigning a comparator to one of the Fault sources via the virtual PPS pins, and then selecting the desired Fault as the source for Fault and Current-Limit Control. On dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 devices, analog comparators have a direct connection to Fault and Current-Limit Control, and can be selected with the following values for the CLSRC or FLTSRC bits:

- 00000 = Analog Comparator 1
- 00001 = Analog Comparator 2
- 00010 = Analog Comparator 3
- 00011 = Analog Comparator 4

#### A.2.3 LEADING-EDGE BLANKING (LEB)

The Leading-Edge Blanking Delay (LEB) bits have been moved from the LEBCOx register on dsPIC33FJ06GS101/X02 and dsPIC33FJ16GSX02/X04 devices to the LEBDLYx register on dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 devices.

## APPENDIX B: REVISION HISTORY

### Revision B (November 2009)

#### Revision A (March 2009)

This is the initial release of this document.

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 [Table B-1](#).

**TABLE B-1: MAJOR SECTION UPDATES**

Section Name	Update Description
<b>“High-Performance, 16-bit Digital Signal Controllers”</b>	<p>Added “DMA Channels” column and updated the RAM size to 9K for the dsPIC33FJ64GS406 devices in the controller families table (see Table 1).</p> <p>Updated the pin diagrams as follows:</p> <ul style="list-style-type: none"> <li>• 64-pin TQFP and QFN <ul style="list-style-type: none"> <li>- Removed FLT8 from pin 51</li> <li>- Added FLT8 to pin 60</li> <li>- Added FLT17 to pin 31</li> <li>- Added FLT18 to pin32</li> </ul> </li> <li>• 80-pin TQFP <ul style="list-style-type: none"> <li>- Removed FLT8 from pin 63</li> <li>- Added FLT8 to pin 76</li> <li>- Added FLT19 to pin 53</li> <li>- Added FLT20 to pin 52</li> </ul> </li> <li>• 100-pin TQFP <ul style="list-style-type: none"> <li>- Removed FLT8 from pin 78</li> <li>- Added FLT8 to pin 93</li> <li>- Added SYNCO1 to pin 95</li> </ul> </li> </ul>
<b>Section 4.0 “Memory Organization”</b>	<p>Added Data Memory Map for Devices with 8 KB RAM (see Figure 4-4).</p> <p>Removed SFRs IPC25 and IPC26 from the Interrupt Controller Register Map for dsPIC33FJ32GS406 and dsPIC33FJ64GS406 devices (see Table 4-7).</p> <p>The following bits in the Interrupt Controller Register Map for dsPIC33FJ32GS406 and dsPIC33FJ64GS406 devices were changed to unimplemented (see Table 4-7):</p> <ul style="list-style-type: none"> <li>• Bit 2 of IFS1</li> <li>• Bits 9-7 of IFS6</li> <li>• Bit 2 of IEC1</li> <li>• Bits 9-7 of IEC6</li> <li>• Bits 10-8 of IPC4</li> </ul> <p>Removed OSCTUN2 and LFSR, updated OSCCON and OSCTUN, renamed bit 13 of the REFOCON SFR in the System Control Register Map from ROSIDL to ROSSLP and changed the All Resets value from ‘0000’ to ‘2300’ for the ACLKCON SFR (see Table 4-56).</p> <p>Updated bit 1 of the PMD Register Map for dsPIC33FJ64GS608 devices from unimplemented to C1MD (see Table 4-60).</p>

**TABLE B-1: MAJOR SECTION UPDATES (CONTINUED)**

Section Name	Update Description
<b>Section 9.0 “Oscillator Configuration”</b>	<p>Removed Section 9.2 “FRC Tuning”.</p> <p>Removed the PRCDEN, TSEQEN, and LPOSCEN bits from the Oscillator Control Register (see Register 9-1).</p> <p>Updated the Oscillator Tuning Register (see Register 9-4).</p> <p>Removed the Oscillator Tuning Register 2 and the Linear Feedback Shift Register.</p> <p>Updated the default reset values from R/W-0 to R/W-1 for the SELACLK and APSTSCLR&lt;2:0&gt; bits in the ACLKCON register (see Register 9-5).</p> <p>Renamed the ROSIDL bit to ROSSLP in the REFOCON register (see Register 9-6).</p>
<b>Section 10.0 “Power-Saving Features”</b>	<p>Updated the last paragraph of <b>Section 10.2.2 “Idle Mode”</b> to clarify when instruction execution begins.</p> <p>Added Note 1 to the PMD1 register (see Register 10-1).</p>
<b>Section 11.0 “I/O Ports”</b>	<p>Changed the reference to digital-only pins to 5V tolerant pins in the second paragraph of <b>Section 11.2 “Open-Drain Configuration”</b>.</p>
<b>Section 16.0 “High-Speed PWM”</b>	<p>Updated the High-Speed PWM Module Register Interconnect Diagram (see Figure 16-2).</p> <p>Updated the SYNCSRC&lt;2:0&gt; = 111, 101, and 100 definitions to Reserved in the PTCOEN and STCOEN registers (see Register 16-1 and Register 16-5).</p> <p>Updated the PWM time base maximum value from 0xFFFFB to 0xFFFF8 in the PTPER register (Register 16-3).</p> <p>Updated the smallest pulse width value from 0x0008 to 0x0009 in Note 1 of the shaded note that follows the MDC register (see Register 16-10).</p> <p>Updated the smallest pulse width value from 0x0008 to 0x0009 in Note 2 of the shaded note that follows the PDCx and SDCx registers (see Register 16-12 and Register 16-13).</p> <p>Added Note 2 and updated the FLTDAT&lt;1:0&gt; and CLDAT&lt;1:0&gt; bits, changing the word ‘data’ to ‘state’ in the IOCONx register (see Register 16-19).</p>
<b>Section 20.0 “Universal Asynchronous Receiver Transmitter (UART)”</b>	<p>Updated the two baud rate range features to: 10 Mbps to 38 bps at 40 MIPS.</p>
<b>Section 22.0 “High-Speed 10-bit Analog-to-Digital Converter (ADC)”</b>	<p>Updated the TRGSRCx&lt;4:0&gt; = 01101 definition from Reserved to PWM secondary special event trigger selected, and updated Note 1 in the ADCP0-ADCP6 registers (see Register 22-6 through Register 22-12).</p>
<b>Section 24.0 “Special Features”</b>	<p>Updated the second paragraph and removed the fourth paragraph in <b>Section 24.1 “Configuration Bits”</b>.</p> <p>Updated the Device Configuration Register Map (see Table 24-1).</p>

**TABLE B-1: MAJOR SECTION UPDATES (CONTINUED)**

Section Name	Update Description
<p><b>Section 27.0 “Electrical Characteristics”</b></p>	<p>Updated the Absolute Maximum Ratings for high temperature and added Note 4.</p> <p>Updated all Operating Current (IDD) Typical and Max values in Table 27-5.</p> <p>Updated all Idle Current (IDLE) Typical and Max values in Table 27-6.</p> <p>Updated all Power-Down Current (IPD) Typical and Max values in Table 27-7.</p> <p>Updated all Doze Current (IDOZE) Typical and Max values in Table 27-8.</p> <p>Updated the Typ and Max values for parameter D150 and removed parameters DI26, DI28, and DI29 from the I/O Pin Input Specifications (see Table 27-9).</p> <p>Updated the Typ and Max values for parameter DO10 and DO27 and the Min and Typ values for parameter DO20 in the I/O Pin Output Specifications (see Table 27-10).</p> <p>Added parameter numbers to the Auxiliary PLL Clock Timing Specifications (see Table 27-18).</p> <p>Added parameters numbers and updated the Internal RC Accuracy Min, Typ, and Max values (see Table 27-19 and Table 27-20).</p> <p>Added parameter numbers, Note 2, updated the Min and Typ parameter values for MP31 and MP32, and removed the conditions for MP10 and MP11 in the High-Speed PWM Module Timing Requirements (see Table 27-29).</p> <p>Updated the SPIx Module Slave Mode (CKE = 1) Timing Characteristics (see Figure 27-14).</p> <p>Added parameter IM51 to the I2Cx Bus Data Timing Requirements (Master Mode) (see Table 27-34).</p> <p>Updated the Max value for parameter AD33 in the 10-bit High-Speed ADC Module Specifications (see Table 27-36).</p> <p>Updated the titles and added parameter numbers to the Comparator and DAC Module Specifications (see Table 27-38 and Table 27-39) and the DAC Output Buffer Specifications (see Table 27-40).</p>

## Revision C (February 2010)

This revision includes minor typographical and formatting changes throughout the data sheet text.

All other changes are referenced by their respective section in [Table B-2](#).

**TABLE B-2: MAJOR SECTION UPDATES**

Section Name	Update Description
<b>Section 16.0 “High-Speed PWM”</b>	<p>Added Note 2 to PTPER (Register 16-3).</p> <p>Added Note 1 to SEVTCMP (Register 16-4).</p> <p>Updated Note 1 in MDC (Register 16-10).</p> <p>Updated Note 5 and added Note 6 to PWMCONx (Register 16-11).</p> <p>Updated Note 1 in PDCx (Register 16-12).</p> <p>Updated Note 1 in SDCx (Register 16-13).</p> <p>Updated Note 1 and Note 2 in PHASEx (Register 16-14).</p> <p>Updated Note 2 in SPHASEx (Register 16-15).</p> <p>Updated Note 1 in FCLCONx (Register 16-21).</p> <p>Added Note 1 to STRIGx (Register 16-22).</p> <p>Updated Leading-Edge Blanking Delay increment value from 8.4 ns to 8.32 ns and added a shaded note in LEBDLYx (Register 16-24).</p> <p>Added Note 3 and Note 4 to PWMCAPx (Register 16-26).</p>
<b>Section 27.0 “Electrical Characteristics”</b>	<p>Updated the Min and Typ values for the Internal Voltage Regulator specifications in Table 27-13.</p> <p>Updated the Min and Max values for the Internal RC Accuracy specifications in Table 27-20.</p>

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

## Revision D (January 2012)

This revision includes minor typographical and formatting changes throughout the data sheet text.

All occurrences of PGC<sub>n</sub> and PGD<sub>n</sub> (where n = 1, 2, or 3) were updated to: PGEC<sub>n</sub> and PGED<sub>n</sub> throughout the document.

All other changes are referenced by their respective section in [Table B-3](#).

**TABLE B-3: MAJOR SECTION UPDATES**

Section Name	Update Description
<b>“16-bit Digital Signal Controllers (up to 64 KB Flash and 9 KB SRAM) with High-Speed PWM, ADC, and Comparators”</b>	<p>Added 50 MIPS to Operating Range.</p> <p>Changed the Oscillator frequency range in System Management.</p> <p>Added the <b>“Referenced Sources”</b> section.</p>
<b>Section 1.0 “Device Overview”</b>	Updated the block diagram of the core and peripheral modules (see <a href="#">Figure 1-1</a> ).
<b>Section 2.0 “Guidelines for Getting Started with 16-bit Digital Signal Controllers”</b>	<p>Updated the Recommended Minimum Connection diagram (see <a href="#">Figure 2-1</a>).</p> <p>Updated the VCAP pin capacitor specification in <a href="#">Section 2.3 “Capacitor on Internal Voltage Regulator (VCAP)”</a>.</p>
<b>Section 4.0 “Memory Organization”</b>	<p>Removed IPC20 and updated IFS5, IFS7, IEC5, IEC7, and IPC29 in the Interrupt Controller Register Map for dsPIC33FJ64GS606 devices (see <a href="#">Table 4-6</a>).</p> <p>Removed IPC20 and IPC21 and updated IFS5, IFS7, IEC5, IEC7, and IPC29 in the Interrupt Controller Register Map for dsPIC33FJ32GS406 and dsPIC33FJ64GS406 devices (see <a href="#">Table 4-7</a>).</p> <p>Removed IPC20 and updated IFS5, IFS7, IEC5, IEC7, and IPC29 in the Interrupt Controller Register Map for dsPIC33FJ32GS606 devices (see <a href="#">Table 4-10</a>).</p> <p>Added High-Speed 10-bit ADC Register Map for dsPIC33FJ32GS406 and dsPIC33FJ64GS406 devices (see <a href="#">Table 4-35</a>).</p> <p>Updated ODCG in PORTG Register Map for dsPIC33FJ32GS610 and dsPIC33FJ64GS610 devices (see <a href="#">Table 4-54</a>).</p> <p>Updated ODCG in PORTG Register Map for dsPIC33FJ32GS608 and dsPIC33FJ64GS608 devices (see <a href="#">Table 4-55</a>).</p> <p>Updated ODCG in PORTG Register Map for dsPIC33FJ32GS406/606 and dsPIC33FJ64GS406/606 devices (see <a href="#">Table 4-56</a>).</p>
<b>Section 9.0 “Oscillator Configuration”</b>	<p>Changed the High-Speed Crystal (HS) frequency range in <a href="#">Section 9.1.1 “System Clock sources”</a>.</p> <p>Updated the device operating speed to up to 50 MHz in <a href="#">Section 9.1.2 “System Clock Selection”</a>.</p> <p>Updated <a href="#">Section 9.1.3 “PLL Configuration”</a> to reflect the new operating range/speed of 50 MIPS/50 MHz.</p> <p>Updated <a href="#">Section 9.2 “Auxiliary Clock Generation”</a>.</p>

**TABLE B-3: MAJOR SECTION UPDATES (CONTINUED)**

Section Name	Update Description
<p><b>Section 22.0 “High-Speed 10-bit Analog-to-Digital Converter (ADC)”</b></p>	<p>Updated the ADC Block Diagram for dsPIC33FJ32GS406 and dsPIC33FJ64GS406 Devices with one SAR (see <a href="#">Table 22-1</a>).</p> <p>Added Note 2 to ADCPC6: ADC Convert Pair Control Register 6 (see <a href="#">Register 22-12</a>).</p>
<p><b>Section 23.0 “High-Speed Analog Comparator”</b></p>	<p>Added Note 1 to the High-Speed Analog Comparator Module block diagram (see <a href="#">Figure 23-1</a>).</p>
<p><b>Section 24.0 “Special Features”</b></p>	<p>Updated <b>Section 24.1 “Configuration Bits”</b>.</p> <p>Added the RTSP Effect column to the dsPIC33F Configuration Bits Description (see <a href="#">Table 24-2</a>).</p> <p>Added Note 3 to the Connections for the On-chip Voltage Regulator (see <a href="#">Figure 24-1</a>).</p>
<p><b>Section 27.0 “Electrical Characteristics”</b></p>	<p>Updated the Absolute Maximum Ratings.</p> <p>Updated the Operating MIPS vs. Voltage and added Note 1 (see <a href="#">Table 27-1</a>).</p> <p>Updated Note 4 and removed parameter DC18 from the DC Temperature and Voltage Specifications (see <a href="#">Table 27-4</a>).</p> <p>Updated Note 2, Typical and Maximum values for parameters DC20-DC24, and the Conditions for parameters DC25-DC28 in the Operating Current DC Characteristics (see <a href="#">Table 27-5</a>).</p> <p>Updated Note 2 in the Idle Current DC Characteristics (see <a href="#">Table 27-6</a>).</p> <p>Updated Note 2 in the Power-down Current DC Characteristics (see <a href="#">Table 27-7</a>).</p> <p>Added Note 2 to the Doze Current DC Characteristics (see <a href="#">Table 27-8</a>).</p> <p>Added parameters DI60a, DI60b, and DI60c to the I/O Pin Input Specifications (see <a href="#">Table 27-9</a>).</p> <p>Updated all I/O Pin Output Specifications (see <a href="#">Table 27-10</a>).</p> <p>Updated parameter BO10 and added Note 2 and Note 3 to the BOR Electrical Characteristics (see <a href="#">Table 27-11</a>).</p> <p>Added Note 1 to the Internal Voltage Regulator Specifications (see <a href="#">Table 27-13</a>).</p> <p>Updated the OS25 parameter in the External Clock Timing diagram (see <a href="#">Figure 27-2</a>).</p> <p>Added the Secondary Oscillator (Sosc) to parameter OS10, added parameter OS42 (GM), and added Note 2 to the External Clock Timing Requirements (see <a href="#">Table 27-16</a>).</p> <p>Updated Note 2 in the Internal FRC Accuracy AC Characteristics (see <a href="#">Table 27-19</a>).</p> <p>Updated parameters DO31 and DO32 in the I/O Timing Requirements (see <a href="#">Table 27-21</a>).</p>

**TABLE B-3: MAJOR SECTION UPDATES (CONTINUED)**

Section Name	Update Description
<b>Section 27.0 “Electrical Characteristics” (Continued)</b>	Updated the Timer1, Timer2, and Timer3 External Clock Timing Requirements (see <a href="#">Table 27-23</a> , <a href="#">Table 27-24</a> , and <a href="#">Table 27-25</a> ).  Updated the Simple OC/PWM Mode Timing Requirements (see <a href="#">Table 27-28</a> ).  Updated all SPI Timing specifications (see <a href="#">Figure 27-11-Figure 27-18</a> and <a href="#">Table 27-30-Table 27-37</a> ).  Added Note 2 to the 10-bit High-Speed ADC Module Specifications (see <a href="#">Table 27-40</a> ).  Added Note 2 to the 10-bit High-Speed ADC Module Timing Requirements (see <a href="#">Table 27-41</a> ).  Added parameter DA08 to the DAC Module Specifications (see <a href="#">Table 27-43</a> ).  Updated parameter DA16 in the DAC Output Buffer Specifications (see <a href="#">Table 27-44</a> ).  Added DMA Read/Write Timing Requirements (see <a href="#">Table 27-49</a> ).
<b>Section 28.0 “50 MIPS Electrical Characteristics”</b>	Added new chapter with electrical specifications for 50 MIPS devices.
<b>Section 29.0 “DC and AC Device Characteristics Graphs”</b>	Added new chapter.



**INDEX**

**A**

AC Characteristics ..... 374, 413  
 Internal RC Accuracy ..... 377  
 Load Conditions ..... 374  
 Alternate Interrupt Vector Table (AIVT) ..... 123  
 Arithmetic Logic Unit (ALU) ..... 39  
 Assembler  
 MPASM Assembler ..... 358

**B**

Barrel Shifter ..... 43  
 Bit-Reversed Addressing ..... 102  
 Example ..... 103  
 Implementation ..... 102  
 Sequence Table (16-Entry) ..... 103  
 Block Diagrams  
 16-bit Timer1 Module ..... 211  
 Comparator ..... 337  
 Connections for On-Chip Voltage Regulator ..... 345  
 Device Clock ..... 190  
 DSP Engine ..... 40  
 dsPIC33FJ32GS406/606/608/610 and  
 dsPIC33FJ64GS406/606/608/610 ..... 18  
 dsPIC33FJ32GS406/606/608/610 and  
 dsPIC33FJ64GS406/606/608/610 CPU Core ..... 34  
 ECAN Module ..... 280  
 I<sup>2</sup>C ..... 266  
 Input Capture ..... 219  
 Oscillator System ..... 188  
 Output Compare ..... 221  
 PLL ..... 190  
 Quadrature Encoder Interface ..... 255  
 Reset System ..... 115  
 Shared Port Structure ..... 209  
 Simplified Conceptual High-Speed PWM ..... 226  
 SPI ..... 259  
 Timer2/3 (32-bit) ..... 215  
 Type B Timer ..... 213  
 Type C Timer ..... 213  
 UART ..... 273  
 Watchdog Timer (WDT) ..... 346  
 Brown-out Reset (BOR) ..... 341

**C**

C Compilers  
 MPLAB C18 ..... 358  
 Clock Switching ..... 198  
 Enabling ..... 198  
 Sequence ..... 198  
 Code Examples  
 Erasing a Program Memory Page ..... 113  
 Initiating a Programming Sequence ..... 114  
 Loading Write Buffers ..... 114  
 Port Write/Read ..... 210  
 PWRSAV Instruction Syntax ..... 199  
 Code Protection ..... 341, 348  
 CodeGuard Security ..... 341  
 Configuration Bits ..... 341  
 Configuration Register Map ..... 341  
 Configuring Analog Port Pins ..... 210  
 CPU  
 Control Registers ..... 36  
 CPU Clocking System ..... 189

PLL Configuration ..... 190  
 Selection ..... 189  
 Sources ..... 189  
 Customer Change Notification Service ..... 447  
 Customer Notification Service ..... 447  
 Customer Support ..... 447

**D**

DAC ..... 338  
 Output Range ..... 338  
 Data Accumulators and Adder/Subtractor ..... 41  
 Data Space Write Saturation ..... 43  
 Overflow and Saturation ..... 41  
 Round Logic ..... 42  
 Write Back ..... 42  
 Data Address Space ..... 47  
 Alignment ..... 47  
 Memory Map for dsPIC33FJ32GS406/606/608/610 De-  
 vices with 4 KB RAM ..... 48  
 Memory Map for dsPIC33FJ64GS406/606/608/610 De-  
 vices with 8 KB RAM ..... 49  
 Memory Map for dsPIC33FJ64GS406/606/608/610 De-  
 vices with 9 KB RAM ..... 50  
 Near Data Space ..... 47  
 Software Stack ..... 99  
 Width ..... 47  
 DC and AC Characteristics  
 Graphs and Tables ..... 417  
 DC Characteristics ..... 362, 410  
 Doze Current (IDOZE) ..... 368, 412  
 I/O Pin Input Specifications ..... 369  
 I/O Pin Output Specifications ..... 371  
 Idle Current (IDLE) ..... 366, 411  
 Operating Current (IDD) ..... 364, 410  
 Power-Down Current (IPD) ..... 367  
 Program Memory ..... 373  
 Temperature and Voltage Specifications ..... 363  
 Development Support ..... 357  
 DMAC Registers ..... 178  
 DMAxCNT ..... 178  
 DMAxCON ..... 178  
 DMAxPAD ..... 178  
 DMAxREQ ..... 178  
 DMAxSTA ..... 178  
 DMAxSTB ..... 178  
 Doze Mode ..... 200  
 DSP Engine ..... 39  
 Multiplier ..... 41

**E**

ECAN Module  
 CiBUFNT1 register ..... 291  
 CiBUFNT2 register ..... 292  
 CiBUFNT3 register ..... 292  
 CiBUFNT4 register ..... 293  
 CiCFG1 register ..... 289  
 CiCFG2 register ..... 290  
 CiCTRL1 register ..... 282  
 CiCTRL2 register ..... 283  
 CiEC register ..... 289  
 CiFCTRL register ..... 285  
 CiFEN1 register ..... 291  
 CiFIFO register ..... 286  
 CiFMSKSEL1 register ..... 295

CiFMSKSEL2 register .....	296	<b>H</b>	High-Speed Analog Comparator.....	337
CiINTE register .....	288		High-Speed PWM .....	225
CiINTF register.....	287	<b>I</b>	I/O Ports.....	209
CiRXFnEID register .....	295		Parallel I/O (PIO).....	209
CiRXFnSID register .....	294		Write/Read Timing.....	210
CiRXFUL1 register .....	298	<b>I<sup>2</sup>C</b>	Operating Modes .....	265
CiRXFUL2 register .....	298		Registers .....	265
CiRXMnEID register.....	297		In-Circuit Debugger.....	347
CiRXMnSID register.....	297		In-Circuit Emulation .....	341
CiRXOVF1 register .....	299		In-Circuit Serial Programming (ICSP).....	341, 347
CiRXOVF2 register.....	299		Input Capture .....	219
CiTRmnCON register.....	300		Registers .....	220
CiVEC register .....	284		Input Change Notification .....	210
Frame Types .....	279		Instruction Addressing Modes .....	99
Modes of Operation .....	281		File Register Instructions .....	99
Overview .....	279		Fundamental Modes Supported .....	100
<b>ECAN Registers</b>			MAC Instructions .....	100
Acceptance Filter Enable Register (CiFEN1).....	291		MCU Instructions .....	99
Acceptance Filter Extended Identifier Register n (CiRXFnEID).....	295		Move and Accumulator Instructions.....	100
Acceptance Filter Mask Extended Identifier Register n (CiRXMnEID).....	297		Other Instructions .....	100
Acceptance Filter Mask Standard Identifier Register n (CiRXMnSID).....	297		Instruction Set	
Acceptance Filter Standard Identifier Register n (CiRXFnSID).....	294		Overview.....	352
Baud Rate Configuration Register 1 (CiCFG1).....	289		Summary .....	349
Baud Rate Configuration Register 2 (CiCFG2).....	290		Instruction-Based Power-Saving Modes.....	199
Control Register 1 (CiCTRL1).....	282		Idle.....	200
Control Register 2 (CiCTRL2).....	283		Sleep .....	199
FIFO Control Register (CiFCTRL) .....	285		Interfacing Program and Data Memory Spaces.....	104
FIFO Status Register (CiFIFO) .....	286		Internal RC Oscillator	
Filter 0-3 Buffer Pointer Register (CiBUFPTNT1) .....	291		Use with WDT.....	345
Filter 12-15 Buffer Pointer Register (CiBUFPTNT4) .....	293		Internet Address .....	447
Filter 15-8 Mask Selection Register (CiFMSKSEL2).....	296		Interrupt Control and Status Registers .....	127
Filter 4-7 Buffer Pointer Register (CiBUFPTNT2) .....	292		IECx .....	127
Filter 7-0 Mask Selection Register (CiFMSKSEL1).....	295		IFSx .....	127
Filter 8-11 Buffer Pointer Register (CiBUFPTNT3) .....	292		INTCON1 .....	127
Interrupt Code Register (CiVEC) .....	284		INTCON2.....	127
Interrupt Enable Register (CiINTE) .....	288		INTTREG .....	127
Interrupt Flag Register (CiINTF) .....	287		IPCx.....	127
Receive Buffer Full Register 1 (CiRXFUL1).....	298		Interrupt Setup Procedures.....	176
Receive Buffer Full Register 2 (CiRXFUL2).....	298		Initialization .....	176
Receive Buffer Overflow Register 2 (CiRXOVF2).....	299		Interrupt Disable .....	176
Receive Overflow Register (CiRXOVF1) .....	299		Interrupt Service Routine .....	176
<b>ECAN Transmit/Receive Error Count Register (CiEC) .....</b>	<b>289</b>		Trap Service Routine.....	<b>176</b>
<b>ECAN TX/RX Buffer m Control Register (CiTRmnCON) ..</b>	<b>300</b>		Interrupt Vector Table (IVT) .....	<b>123</b>
<b>Electrical Characteristics .....</b>	<b>361, 409</b>		Interrupts Coincident with Power Save Instructions .....	<b>200</b>
AC Characteristics and Timing Parameters .....	374, 413		<b>J</b>	
BOR .....	372		JTAG Boundary Scan Interface .....	341
<b>Enhanced CAN Module.....</b>	<b>279</b>		JTAG Interface.....	347
<b>Equations</b>			<b>L</b>	
Device Operating Frequency .....	189		Leading-Edge Blanking (LEB) .....	225
Fosc Calculation.....	190		<b>M</b>	
XT with PLL Mode Example.....	190		Memory Organization .....	45
<b>Errata .....</b>	<b>14</b>		Microchip Internet Web Site.....	447
<b>F</b>			Migrating from dsPIC33FJ06GS101/X02 and dsPIC33FJ16GSX02/X04 to dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610 Devices .....	433
Fail-Safe Clock Monitor (FSCM) .....	198		Migration	
Flash Program Memory.....	109		Analog Comparators Connection.....	433
Control Registers .....	110		Device Pins and Peripheral Pin Select (PPS).....	433
Operations .....	110			
Programming Algorithm .....	113			
RTSP Operation.....	110			
Table Instructions.....	109			
Flexible Configuration .....	341			

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

Fault and Current-Limit Control Signal Source Selection 433	Change Notification (dsPIC33FJ32GS608/610 and dsPIC33FJ64GS608/601 Devices).....	54
Leading-Edge Blanking (LEB).....	Change Notification (dsPIC33FJ64GS406/606 Devices) 54	54
Modes of Operation	CPU Core .....	52
Disable .....	DMA.....	87
Initialization .....	ECAN1 (C1CTRL1.WIN = 0 or 1).....	88
Listen All Messages .....	ECAN1 (C1CTRL1.WIN = 0).....	88
Listen Only .....	ECAN1 (C1CTRL1.WIN = 1).....	89
Loopback .....	High-Speed 10-bit ADC Module (dsPIC33FJ32GS608 and dsPIC33FJ64GS608 Devices).....	84
Normal Operation.....	High-Speed 10-bit ADC Module (dsPIC33FJ32GS610 and dsPIC33FJ64GS610 Devices).....	82
Modulo Addressing .....	High-Speed 10-bit ADC Module (for dsPIC33FJ32GS606 and dsPIC33FJ64GS606 De- vices).....	85, 86
Applicability .....	High-Speed PWM.....	71
Operation Example .....	High-Speed PWM Generator 1.....	71
Start and End Address.....	High-Speed PWM Generator 2.....	72
W Address Register Selection .....	High-Speed PWM Generator 3.....	73
MPLAB ASM30 Assembler, Linker, Librarian .....	High-Speed PWM Generator 4.....	74
MPLAB Integrated Development Environment Software ..	High-Speed PWM Generator 5.....	75
MPLAB PM3 Device Programmer .....	High-Speed PWM Generator 6.....	76
MPLAB REAL ICE In-Circuit Emulator System.....	High-Speed PWM Generator 7 (All devices except dsPIC33FJ32GS406 and dsPIC33FJ64GS406) 77	77
MPLINK Object Linker/MPLIB Object Librarian .....	High-Speed PWM Generator 8 (All devices except dsPIC33FJ32GS406 and dsPIC33FJ64GS406) 78	78
<b>O</b>	High-Speed PWM Generator 9 (dsPIC33FJ32GS610 and dsPIC33FJ64GS610 Devices).....	79
Open-Drain Configuration .....	I2C1 .....	79
Oscillator Configuration.....	I2C2 .....	80
Oscillator Tuning Register (OSCTUN) .....	Input Capture.....	69
Output Compare .....	Interrupt Controller (dsPIC33FJ32GS406 and dsPIC33FJ64GS406 Devices).....	61
<b>P</b>	Interrupt Controller (dsPIC33FJ32GS606 Devices) ...	67
Packaging .....	Interrupt Controller (dsPIC33FJ32GS608 Devices) ...	65
100-Lead TQFP .....	Interrupt Controller (dsPIC33FJ32GS610 Devices) ...	63
100-Lead TQFP Land Pattern.....	Interrupt Controller (dsPIC33FJ64GS606 Devices) ...	59
64-Lead QFN .....	Interrupt Controller (dsPIC33FJ64GS608 Devices) ...	57
64-Lead QFN Land Pattern.....	Interrupt Controller (dsPIC33FJ64GS610 Devices) ...	55
64-Lead TQFP .....	NVM.....	96
64-Lead TQFP Land Pattern.....	Output Compare .....	70
80-Lead TQFP .....	PMD (dsPIC33FJ64GS606 Devices) .....	97
80-Lead TQFP Land Pattern.....	PMD (dsPIC33FJ32GS406 and dsPIC33FJ64GS406 Devices).....	98
Marking .....	PMD (dsPIC33FJ32GS606 Devices) .....	98
Peripheral Module Disable (PMD) .....	PMD (dsPIC33FJ32GS608 Devices) .....	97
Pinout I/O Descriptions (table) .....	PMD (dsPIC33FJ32GS610 Devices) .....	96
Power-on Reset (POR) .....	PMD (dsPIC33FJ64GS608 Devices) .....	97
Power-Saving Features .....	PMD (dsPIC33FJ64GS610 Devices) .....	96
Clock Frequency and Switching.....	PORTA (dsPIC33FJ32GS608 and dsPIC33FJ64GS608 Devices).....	91
Program Address Space.....	PORTA (dsPIC33FJ32GS610 and dsPIC33FJ64GS610 Devices).....	91
Construction.....	PORTB .....	92
Data Access from Program Memory Using Program Space Visibility.....	PORTC (dsPIC33FJ32GS406/606 and dsPIC33FJ64GS406/606 Devices).....	92
Data Access from Program Memory Using Table Instruc- tions .....	PORTC (dsPIC33FJ32GS608 and dsPIC33FJ64GS608 Devices).....	92
Data Access from, Address Generation.....	PORTC (dsPIC33FJ32GS610 and dsPIC33FJ64GS610 Devices).....	92
Memory Map .....	PORTD (dsPIC33FJ32GS406/606 and dsPIC33FJ64GS406/606 Devices).....	93
Table Read Instructions	PORTD (dsPIC33FJ32GS608/610 and dsPIC33FJ64GS608/610 Devices).....	93
TBLRDH .....	PORTE (dsPIC33FJ32GS406/606 and	
TBLRDL .....		
Visibility Operation .....		
Program Memory		
Interrupt Vector .....		
Organization.....		
Reset Vector .....		
<b>Q</b>		
Quadrature Encoder Interface (QEI).....		
<b>R</b>		
Reader Response .....		
Register Maps		
Analog Comparator.....		

# dsPIC33FJ32GS406/606/608/610 and dsPIC33FJ64GS406/606/608/610

dsPIC33FJ64GS406/606 Devices).....	93	Identifier).....	295
PORTE (dsPIC33FJ32GS608/610 and dsPIC33FJ64GS608/610 Devices).....	93	CiRXFnSID (ECAN Acceptance Filter n Standard Identifier).....	294
PORTF (dsPIC33FJ32GS406/606 and dsPIC33FJ64GS406/606 Devices).....	94	CiRXFUL1 (ECAN Receive Buffer Full 1).....	298
PORTF (dsPIC33FJ32GS608 and dsPIC33FJ64GS608 Devices).....	94	CiRXFUL2 (ECAN Receive Buffer Full 2).....	298
PORTF (dsPIC33FJ32GS610 and dsPIC33FJ64GS610 Devices).....	94	CiRXMnEID (ECAN Acceptance Filter Mask n Extended Identifier).....	297
PORTG (dsPIC33FJ32GS406/606 and dsPIC33FJ64GS406/606 Devices).....	95	CiRXMnSID (ECAN Acceptance Filter Mask n Standard Identifier).....	297
PORTG (dsPIC33FJ32GS608 and dsPIC33FJ64GS608 Devices).....	95	CiRXOVF1 (ECAN Receive Buffer Overflow 1).....	299
PORTG (dsPIC33FJ32GS610 and dsPIC33FJ64GS610 Devices).....	94	CiRXOVF2 (ECAN Receive Buffer Overflow 2).....	299
Quadrature Encoder Interface (QEI) 1.....	70	CiTRBnSID (ECAN Buffer n Standard Identifier).....	301, 302, 304
Quadrature Encoder Interface (QEI) 2.....	70	CiTRmCON (ECAN TX/RX Buffer m Control).....	300
SPI1.....	81	CiVEC (ECAN Interrupt Code).....	284
SPI2.....	81	CLKDIV (Clock Divisor).....	193
System Control.....	95	CMPCONx (Comparator Control).....	339
Timers.....	69	CMPCPNx (Comparator Control).....	339
UART1.....	80	CMPDACx (Comparator DAC Control).....	340
UART2.....	80	CORCON (Core Control).....	38, 128
Registers		DFLTCON (QEI Control).....	258
A/D Convert Pair Control Register 2 (ADCPC2).....	322	DFLTxCN (Digital Filter Control).....	258
ACLKCON (Auxiliary Clock Divisor Control).....	196	DMACS0 (DMA Controller Status 0).....	183
ADBASE (A/D Base).....	314	DMACS1 (DMA Controller Status 1).....	184
ADC Base Register (ADBASE).....	314	DMAxCNT (DMA Channel x Transfer Count).....	182
ADC Control Register (ADCON).....	311	DMAxCON (DMA Channel x Control).....	179
ADC Port Configuration Register (ADPCFG).....	315	DMAxPAD (DMA Channel x Peripheral Address).....	181
ADC Status Register (ADSTAT).....	313	DMAxREQ (DMA Channel x IRQ Select).....	180
ADPCFG (A/D Port Configuration).....	315	DMAxSTA (DMA Channel x RAM Start Address A).....	180
ADPCFG2 (A/D Port Configuration).....	315	DMAxSTB (DMA Channel x RAM Start Address B).....	181
ADSTAT (A/D Status).....	313	DSADR (Most Recent DMA RAM Address).....	185
ALTDTRx (PWM Alternate Dead Time).....	242	DTRx (PWM Dead Time).....	242
Analog-to-Digital Convert Pair Control Register 0 (ADCPC0).....	316	FCLCONx (PWM Fault Current-Limit Control).....	247
Analog-to-Digital Convert Pair Control Register 1 (ADCPC1).....	319	I2CxCON (I2Cx Control).....	267
Analog-to-Digital Convert Pair Control Register 3 (ADCPC3).....	325	I2CxMSK (I2Cx Slave Mode Address Mask).....	271
Analog-to-Digital Convert Pair Control Register 4 (ADCPC4).....	328	I2CxSTAT (I2Cx Status).....	269
Analog-to-Digital Convert Pair Control Register 5 (ADCPC5).....	331	ICxCON (Input Capture x Control, x = 1, 2).....	220
Analog-to-Digital Convert Pair Control Register 6 (ADCPC6).....	334	ICxCON (Input Capture x Control).....	220
AUXCONx (PWM Auxiliary Control).....	252	IEC0 (Interrupt Enable Control 0).....	141
CHOP (PWM Chop Clock Generator).....	235	IEC1 (Interrupt Enable Control 1).....	143
CiBUFNT1 (ECAN Filter 0-3 Buffer Pointer).....	291	IEC2 (Interrupt Enable Control 2).....	144
CiBUFNT2 (ECAN Filter 4-7 Buffer Pointer).....	292	IEC3 (Interrupt Enable Control 3).....	145
CiBUFNT3 (ECAN Filter 8-11 Buffer Pointer).....	292	IEC4 (Interrupt Enable Control 4).....	146
CiBUFNT4 (ECAN Filter 12-15 Buffer Pointer).....	293	IEC5 (Interrupt Enable Control 5).....	147
CiCFG1 (ECAN Baud Rate Configuration 1).....	289	IEC6 (Interrupt Enable Control 6).....	148
CiCFG2 (ECAN Baud Rate Configuration 2).....	290	IEC7 (Interrupt Enable Control 7).....	149
CiCTRL1 (ECAN Control 1).....	282	IFS0 (Interrupt Flag Status 0).....	132
CiCTRL2 (ECAN Control 2).....	283	IFS1 (Interrupt Flag Status 1).....	134
CiEC (ECAN Transmit/Receive Error Count).....	289	IFS2 (Interrupt Flag Status 2).....	135
CiFCTRL (ECAN FIFO Control).....	285	IFS3 (Interrupt Flag Status 3).....	136
CiFEN1 (ECAN Acceptance Filter Enable).....	291	IFS4 (Interrupt Flag Status 4).....	137
CiFIFO (ECAN FIFO Status).....	286	IFS5 (Interrupt Flag Status 5).....	138
CiFMSKSEL1 (ECAN Filter 7-0 Mask Selection).....	295	IFS6 (Interrupt Flag Status 6).....	139
CiFMSKSEL2 (ECAN Filter 15-8 Mask Selection).....	296	IFS7 (Interrupt Flag Status 7).....	140
CiINTE (ECAN Interrupt Enable).....	288	INTCON1 (Interrupt Control 1).....	129
CiINTF (ECAN Interrupt Flag).....	287	INTCON1 (Interrupt Control Register 1).....	129
CiRXFnEID (ECAN Acceptance Filter n Extended Identifier).....	295	INTCON2 (Interrupt Control Register 2).....	131
		INTTREG (Interrupt Control and Status).....	175
		INTTREG Interrupt Control and Status.....	175
		IOCONx (PWM I/O Control).....	244
		IPC0 (Interrupt Priority Control 0).....	150
		IPC1 (Interrupt Priority Control 1).....	151
		IPC12 (Interrupt Priority Control 12).....	160
		IPC13 (Interrupt Priority Control 13).....	161
		IPC14 (Interrupt Priority Control 14).....	162
		IPC16 (Interrupt Priority Control 16).....	163

IPC17 (Interrupt Priority Control 17) .....	164	TRIGx (PWM Primary Trigger Compare Value) .....	246
IPC18 (Interrupt Priority Control 18) .....	165	TxCON (Timer Control, x = 2).....	216
IPC2 (Interrupt Priority Control 2) .....	152	TyCON (Timer Control, y = 3).....	217
IPC20 (Interrupt Priority Control 20) .....	166	UxMODE (UARTx Mode) .....	274
IPC21 (Interrupt Priority Control 21) .....	167	UxSTA (UARTx Status and Control) .....	276
IPC23 (Interrupt Priority Control 23) .....	168	<b>Reset</b>	
IPC24 (Interrupt Priority Control 24) .....	169	Illegal Opcode.....	115, 120
IPC25 (Interrupt Priority Control 25) .....	170	Trap Conflict .....	120
IPC26 (Interrupt Priority Control 26) .....	171	Uninitialized W Register .....	115, 120
IPC27 (Interrupt Priority Control 27) .....	172	Reset Sequence .....	123
IPC28 (Interrupt Priority Control 28) .....	173	Resets .....	115
IPC29 (Interrupt Priority Control 29) .....	174	Resources Required for Digital PFC.....	27, 30
IPC3 (Interrupt Priority Control 3) .....	153	Resources Required for Digital Phase-Shift ZVT Converter32	
IPC4 (Interrupt Priority Control 4) .....	154	<b>S</b>	
IPC5 (Interrupt Priority Control 5) .....	155	Serial Peripheral Interface (SPI).....	259
IPC6 (Interrupt Priority Control 6) .....	156	Software RESET Instruction (SWR) .....	120
IPC7 (Interrupt Priority Control 7) .....	157	Software Simulator (MPLAB SIM) .....	359
IPC8 (Interrupt Priority Control 8) .....	158	Software Stack Pointer, Frame Pointer	
IPC9 (Interrupt Priority Control 9) .....	159	CALL Stack Frame .....	99
LEBCONx (Leading-Edge Blanking Control) .....	250	Special Event Compare Register (SEVTCMP).....	232, 235
LEBDLYx (Leading-Edge Blanking Delay).....	251	Special Features of the CPU .....	341
MDC (PWM Master Duty Cycle) .....	236	Symbols Used in Opcode Descriptions .....	350
NVMCON (Flash Memory Control) .....	111	<b>T</b>	
NVMKEY (Non-Volatile Memory Key).....	112	Temperature and Voltage Specifications	
NVMKEY (Nonvolatile Memory Key) .....	112	AC .....	374
OCxCON (Output Compare x Control, x = 1, 2) .....	223	Timer1 .....	211
OSCCON (Oscillator Control) .....	192	Timer2/3 .....	213
OSCTUN (Oscillator Tuning) .....	195	Timing Diagrams	
PDCx (PWM Generator Duty Cycle).....	239	Analog-to-Digital Conversion per Input.....	403
PHASEx (PWM Primary Phase Shift) .....	240	Brown-out Situations .....	119
PLLFBFD (PLL Feedback Divisor).....	194	CAN I/O .....	408
PMD1 (Peripheral Module Disable Control 1) .....	202	External Clock .....	375
PMD1 (Peripheral Module Disable Control 1) .....	202	High-Speed PWM .....	385
PMD2 (Peripheral Module Disable Control 2) .....	204	High-Speed PWM Fault.....	385
PMD3 (Peripheral Module Disable Control 3).....	205	I/O .....	378
PMD4 (Peripheral Module Disable Control 4).....	205	I2Cx Bus Data (Master Mode).....	398
PMD6 (Peripheral Module Disable Control 6).....	206	I2Cx Bus Data (Slave Mode) .....	400
PMD7 (Peripheral Module Disable Control 7).....	207	I2Cx Bus Start/Stop Bits (Master Mode).....	398
PTCON (PWM Time Base Control) .....	229	I2Cx Bus Start/Stop Bits (Slave Mode).....	400
PTCON2 (PWM Clock Divider Select) .....	231	Input Capture (CAPx) .....	383
PTPER (Primary Master Time Base Period).....	231	OC/PWM .....	384
PWMCAPx (Primary PWM Time Base Capture) .....	253	Output Compare (OCx) .....	383
PWMCONx (PWM Control).....	237	QEA/QEB Input .....	405
QEICON (QEI Control).....	256	QEI Module Index Pulse .....	406
QEIXCON (QEIX Control, x = 1 or 2).....	256	Reset, Watchdog Timer, Oscillator Start-up Timer and	
RCON (Reset Control) .....	116	Power-up Timer .....	379
REFOCON (Reference Oscillator Control) .....	197	Timer1, 2, 3 External Clock .....	381
SDCx (PWM Secondary Duty Cycle).....	239	TimerQ (QEI Module) External Clock .....	407
SEVTCMP .....	235	Timing Requirements	
SEVTCMP (Special Event Compare) .....	232	External Clock .....	375, 413
SPHASEx (PWM Secondary Phase Shift).....	241	I/O .....	378
SPIxCON1 (SPIx Control 1).....	261	Input Capture.....	383
SPIxCON2 (SPIx Control 2).....	263	Timing Specifications	
SPIxSTAT (SPIx Status and Control) .....	260	10-bit Analog-to-Digital Conversion Requirements ..	403
SR (CPU STATUS).....	128	CAN I/O Requirements .....	408
SR (CPU Status).....	36	High-Speed PWM Requirements .....	385
SSEVTCMP (PWM Secondary Special Event Compare)		I2Cx Bus Data Requirements (Master Mode).....	399
235		I2Cx Bus Data Requirements (Slave Mode).....	401
STCON (PWM Secondary Master Time Base Control) ..		Output Compare Requirements.....	383
233		PLL Clock .....	376
STCON2 (PWM Secondary Clock Divider Select)....	234	QEI External Clock Requirements.....	407
STPER (Secondary Master Time Base Period).....	234	QEI Index Pulse Requirements .....	407
STRIGx (PWM Secondary Trigger Compare Value) 249		Quadrature Decoder Requirements .....	406
T1CON (Timer1 Control).....	212		
TRGCONx (PWM Trigger Control) .....	243		

Reset, Watchdog Timer, Oscillator Start-up Timer, Power-up Timer and Brown-out Reset Require- ments .....	380
Simple OC/PWM Mode Requirements .....	384, 416
Timer1 External Clock Requirements .....	381, 414
Timer2 External Clock Requirements .....	382, 415
Timer3 External Clock Requirements .....	382, 415

## U

Universal Asynchronous Receiver Transmitter (UART)....	273
Using the RCON Status Bits .....	121

## V

Voltage Regulator (On-Chip).....	345
----------------------------------	-----

## W

Watchdog Time-out Reset (WDTO) .....	120
Watchdog Timer (WDT) .....	341, 345
Programming Considerations .....	346
WWW Address.....	447
WWW, On-Line Support.....	14

## THE MICROCHIP WEB SITE

Microchip provides online support via our WWW site at [www.microchip.com](http://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](http://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: Technical Publications Manager Total Pages Sent \_\_\_\_\_

RE: Reader Response

From: Name \_\_\_\_\_

Company \_\_\_\_\_

Address \_\_\_\_\_

City / State / ZIP / Country \_\_\_\_\_

Telephone: (\_\_\_\_) \_\_\_\_\_ - \_\_\_\_\_ FAX: (\_\_\_\_) \_\_\_\_\_ - \_\_\_\_\_

Application (optional):

Would you like a reply?  Y  N

Device: dsPIC33FJ32GS406/606/608/610 and  
dsPIC33FJ64GS406/606/608/610

Literature Number: DS70591D

Questions:

1. What are the best features of this document?

\_\_\_\_\_  
\_\_\_\_\_

2. How does this document meet your hardware and software development needs?

\_\_\_\_\_  
\_\_\_\_\_

3. 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 without affecting the overall usefulness?

\_\_\_\_\_  
\_\_\_\_\_

6. Is there any incorrect or misleading information (what and where)?

\_\_\_\_\_  
\_\_\_\_\_

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.

dsPIC 33 FJ 32 GS4 06 T - 50 I / PT - XXX		
<b>Microchip Trademark</b>		
<b>Architecture</b>		
<b>Flash Memory Family</b>		
<b>Program Memory Size (KB)</b>		
<b>Product Group</b>		
<b>Pin Count</b>		
<b>Tape and Reel Flag (if applicable)</b>		
<b>Speed</b>		
<b>Temperature Range</b>		
<b>Package</b>		
<b>Pattern</b>		

<b>Architecture:</b>	33	=	16-bit Digital Signal Controller
<b>Flash Memory Family:</b>	FJ	=	Flash program memory, 3.3V
<b>Product Group:</b>	GS4	=	Switch Mode Power Supply (SMPS) family
	GS6	=	Switch Mode Power Supply (SMPS) family
<b>Pin Count:</b>	06	=	64-pin
	08	=	80-pin
	10	=	100-pin
<b>Speed:</b>	50	=	50 MIPS
		=	40 MIPS (Marking intentionally absent)
<b>Temperature Range:</b>	I	=	-40°C to+85°C (Industrial)
	E	=	-40°C to+125°C (Extended)
<b>Package:</b>	PT	=	Plastic Thin Quad Flatpack - 10x10x1 mm body (TQFP)
	PT	=	Plastic Thin Quad Flatpack - 12x12x1 mm body (TQFP)
	PF	=	Plastic Thin Quad Flatpack - 14x14x1 mm body (TQFP)
	MR	=	Plastic Quad Flat, No Lead Package - 9x9x0.9 mm body (QFN)

**Examples:**

a) dsPIC33FJ32GS406-50I/PT:  
SMPS dsPIC33, 32 KB program memory, 64-pin, 50 MIPS, Industrial temp., TQFP package.

NOTES:

---

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

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

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

---

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

**Trademarks**

The Microchip name and logo, the Microchip logo, dsPIC, KEELOQ, KEELOQ logo, MPLAB, PIC, PICmicro, PICSTART, PIC<sup>32</sup> logo, rPIC 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, rLAB, 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: 978-1-61341-976-2

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

**QUALITY MANAGEMENT SYSTEM**  
**CERTIFIED BY DNV**  
**== ISO/TS 16949:2009 ==**



# MICROCHIP

## 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](http://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

**Korea - Daegu**  
Tel: 82-53-744-4301  
Fax: 82-53-744-4302

**Korea - Seoul**  
Tel: 82-2-554-7200  
Fax: 82-2-558-5932 or  
82-2-558-5934

**Malaysia - Kuala Lumpur**  
Tel: 60-3-6201-9857  
Fax: 60-3-6201-9859

**Malaysia - Penang**  
Tel: 60-4-227-8870  
Fax: 60-4-227-4068

**Philippines - Manila**  
Tel: 63-2-634-9065  
Fax: 63-2-634-9069

**Singapore**  
Tel: 65-6334-8870  
Fax: 65-6334-8850

**Taiwan - Hsin Chu**  
Tel: 886-3-5778-366  
Fax: 886-3-5770-955

**Taiwan - Kaohsiung**  
Tel: 886-7-536-4818  
Fax: 886-7-330-9305

**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

## X-ON Electronics

Largest Supplier of Electrical and Electronic Components

*Click to view similar products for Digital Signal Processors & Controllers - DSP, DSC category:*

*Click to view products by Microchip manufacturer:*

Other Similar products are found below :

[ADSP-2189NBCAZ-320](#) [ADSP-BF703BBCZ-3](#) [dsPIC33EV64GM002-ISP](#) [DSPIC33FJ128MC510-I/PF](#) [MWPR1516CFM](#) [ADSP-2185MKCAZ-300](#) [ADSP-BF701KBCZ-1](#) [dsPIC33EV128GM002-ISP](#) [646890G](#) [ADSP-21587KBCZ-4B](#) [ADSP-BF701KBCZ-2](#) [ADSP-SC582KBCZ-4A](#) [ADSP-SC584BBCZ-4A](#) [ADSP-SC584KBCZ-3A](#) [ADSP-SC589KBCZ-4B](#) [ADSP-SC583KBCZ-4A](#) [ADSP-BF703BBCZ-40W888-002-XTP](#) [ADSP-BF705KBCZ-4](#) [ADSP-BF705BBCZ-3](#) [DSPIC33FJ64GP708-IPT](#) [ADBF707WCBCZ411](#) [ADBF706WCCPZ411](#) [ADSP-21584KBCZ-4A](#) [MC56F82746VLFR](#) [ADBF704WCCPZ311](#) [ADBF702WCCPZ411](#) [AD21571WCSWZ400](#) [MC56F84550VLFR](#) [ADSP-SC571CSWZ-5](#) [ADBF700WCCPZ211](#) [ADSC572WCBCZ302](#) [ADSC572WCBCZ402](#) [ADSC572WCBCZ4202](#) [AD21489WBCPZ402](#) [ADSC572WCBCZ4200](#) [ADSC572WCBCZ400](#) [ADSC573WCBCZ400](#) [ADSC573WCBCZ300](#) [ADSC573WCBCZ500](#) [ADSC571WCSWZ300](#) [ADSC571WCSWZ500](#) [ADSC571WCSWZ400](#) [ADSP-21569KBCZ8](#) [ADSP-21567KBCZ6](#) [ADSP-21567KBCZ8](#) [ADSP-21566KBCZ4](#) [ADSP-21569KBCZ10](#) [MC56F8323VFBE](#) [MC56F8366VFVE](#)