dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 Data Sheet

High-Performance, 16-bit Digital Signal Controllers

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.

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
ISO/TS 16949:2009

Trademarks

The Microchip name and logo, the Microchip logo, dsPIC, KEELOQ, KEELOQ logo, MPLAB, PIC, PICmicro, PICSTART, PIC^{32} logo, rfPIC and UNI/O are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

FilterLab, Hampshire, HI-TECH C, Linear Active Thermistor, MXDEV, MXLAB, SEEVAL and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Analog-for-the-Digital Age, Application Maestro, chipKIT, chipKIT logo, CodeGuard, dsPICDEM, dsPICDEM.net, dsPICworks, dsSPEAK, ECAN, ECONOMONITOR, FanSense, HI-TIDE, In-Circuit Serial Programming, ICSP, Mindi, MiWi, MPASM, MPLAB Certified logo, MPLIB, MPLINK, mTouch, Omniscient Code Generation, PICC, PICC-18, PICDEM, PICDEM.net, PICkit, PICtail, REAL ICE, rfLAB, Select Mode, Total Endurance, TSHARC, UniWinDriver, WiperLock and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

All other trademarks mentioned herein are property of their respective companies.

Printed on recycled paper.

ISBN: 978-1-61341-369-2

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.

dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304

High-Performance, 16-bit Digital Signal Controllers

Operating Range:

- Up to 40 MIPS operation (@ 3.0-3.6V):
  - Industrial temperature range (-40°C to +85°C)
  - Extended temperature range (-40°C to +125°C)
- Up to 20 MIPS operation (@ 3.0-3.6V):
  - High temperature range (-40°C to +150°C)

High-Performance DSC CPU:

- Modified Harvard architecture
- C compiler optimized instruction set
- 16-bit wide data path
- 24-bit wide instructions
- Linear program memory addressing up to 4M instruction words
- Linear data memory addressing up to 64 Kbytes
- 83 base instructions, mostly one word/one cycle
- Sixteen 16-bit General Purpose Registers
- Two 40-bit accumulators with rounding and saturation options
- Flexible and powerful addressing modes:
  - Indirect
  - Modulo
  - Bit-Reversed
- Software stack
- 16 x 16 fractional/integer multiply operations
- 32/16 and 16/16 divide operations
- Single-cycle multiply and accumulate:
  - Accumulator write back for DSP operations
  - Dual data fetch
- Up to ±16-bit shifts for up to 40-bit data

Interrupt Controller:

- 5-cycle latency
- Up to 21 available interrupt sources
- Up to three external interrupts
- Seven programmable priority levels
- Four processor exceptions

On-Chip Flash and SRAM:

- Flash program memory (up to 32 Kbytes)
- Data SRAM (2 Kbytes)
- Boot and General Security for Program Flash

Digital I/O:

- Peripheral Pin Select Functionality
- Up to 35 programmable digital I/O pins
- Wake-up/Interrupt-on-Change for up to 31 pins
- Output pins can drive from 3.0V to 3.6V
- Up to 5.5V output with open drain configuration on 5V tolerant pins with external pull-up
- 4 mA sink on all I/O pins

System Management:

- Flexible clock options:
  - External, crystal, resonator, internal RC
  - Fully integrated Phase-Locked Loop (PLL)
  - Extremely low jitter PLL
- Power-up Timer
- Oscillator Start-up Timer/Stabilizer
- Watchdog Timer with its own RC oscillator
- Fail-Safe Clock Monitor
- Reset by multiple sources

Power Management:

- On-chip 2.5V voltage regulator
- Switch between clock sources in real time
- Idle, Sleep and Doze modes with fast wake-up

Timers/Capture/Compare:

- Timer/Counters, up to three 16-bit timers:
  - Can pair up to make one 32-bit timer
  - One timer runs as Real-Time Clock with external 32.768 kHz oscillator
  - Programmable prescaler
- Input Capture (up to four channels):
  - Capture on up, down or both edges
  - 16-bit capture input functions
  - 4-deep FIFO on each capture
- Output Compare (up to 2 channels):
  - Single or Dual 16-bit Compare mode
  - 16-bit Glitchless PWM Mode

Communication Modules:

- 4-wire SPI:
  - Framing supports I/O interface to simple codecs
  - Supports 8-bit and 16-bit data
  - Supports all serial clock formats and sampling modes
- I^2C™:
  - Full Multi-Master Slave mode support
  - 7-bit and 10-bit addressing
  - Bus collision detection and arbitration
  - Integrated signal conditioning
  - Slave address masking
- UART:
  - Interrupt on address bit detect
  - Interrupt on UART error
  - Wake-up on Start bit from Sleep mode
  - 4-character TX and RX FIFO buffers
  - LIN bus support
  - IrDA® encoding and decoding in hardware
  - High-Speed Baud mode
  - Hardware Flow Control with CTS and RTS

Analog-to-Digital Converters (ADCs):

- 10-bit, 1.1 Msps or 12-bit, 500 ksps conversion:
  - Two and four simultaneous samples (10-bit ADC)
  - Up to 13 input channels with auto-scanning
  - Conversion start can be manual or synchronized with one of four trigger sources
  - Conversion possible in Sleep mode
  - ±2 LSb max integral nonlinearity
  - ±1 LSb max differential nonlinearity

CMOS Flash Technology:

- Low-power, high-speed Flash technology
- Fully static design
- 3.3V (±10%) operating voltage
- Industrial and extended temperature
- Low-power consumption

Packaging:

- 28-pin SPDIP/SOIC/SSOP/QFN-S
- 44-pin QFN/TQFP

Note: See Table 1 for the exact peripheral features per device.

dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 Product Families

The device names, pin counts, memory sizes and peripheral availability of each family are listed below, followed by their pinout diagrams.

TABLE 1: dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 CONTROLLER FAMILIES

Device	Pins	Program Flash Memory (Kbyte)	RAM (Kbyte)	Remappable Peripherals							10-Bit/12-Bit ADC	I²C™	I/O Pins (Max)	Packages
				Remappable Pins	16-bit Timer	Input Capture	Output Compare Std. PWM	UART	External Interrupts(2)	SPI				
dsPIC33FJ32GP202	28	32	2	16	3(1)	4	2	1	3	1	1 ADC, 10 ch	1	21	SPDIP SOIC SSOP QFN-S
dsPIC33FJ32GP204	44	32	2	26	3(1)	4	2	1	3	1	1 ADC, 13 ch	1	35	QFN TQFP
dsPIC33FJ16GP304	44	16	2	26	3(1)	4	2	1	3	1	1 ADC, 13 ch	1	35	QFN TQFP

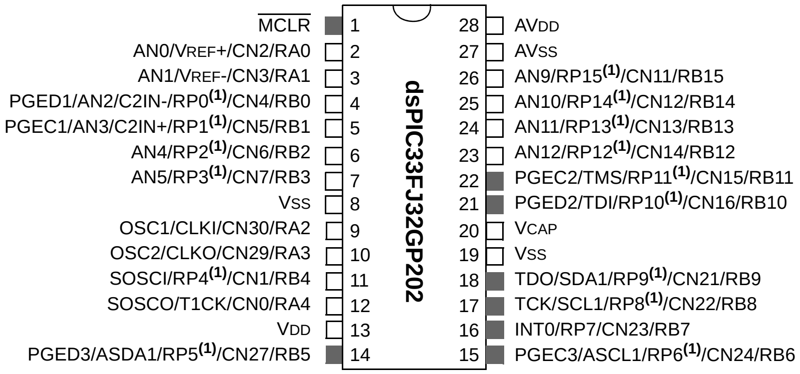
Note 1: Only two out of three timers are remappable.

2: Only two out of three interrupts are remappable.

Pin Diagrams

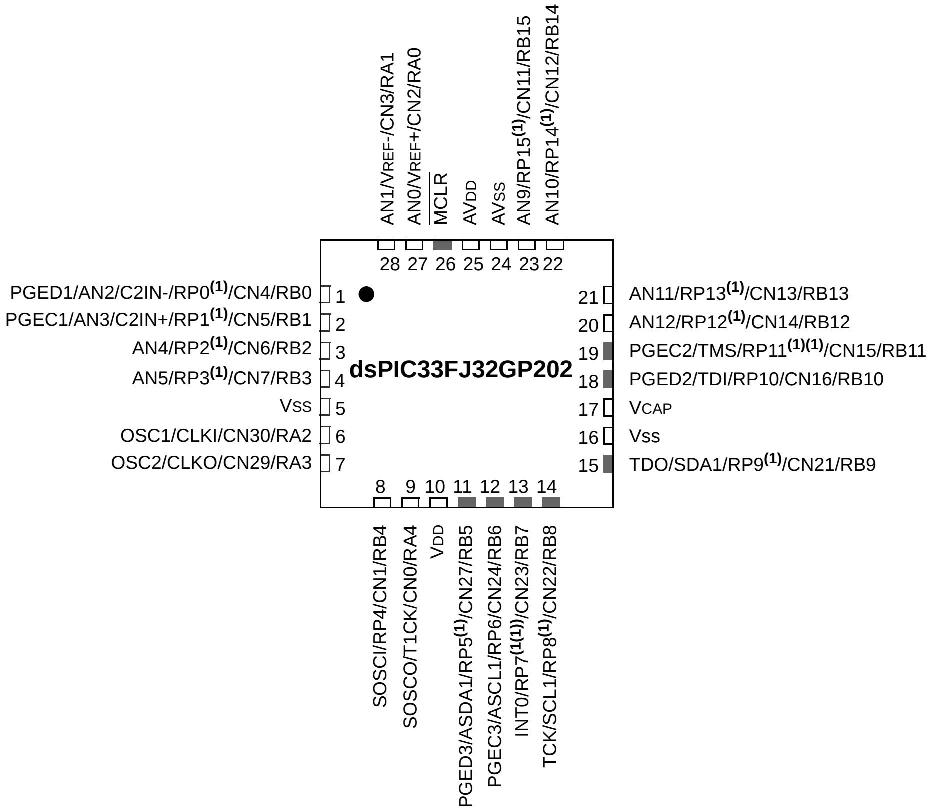
28-Pin SPDIP, SOIC, SSOP

■ = Pins are up to 5V tolerant



28-Pin QFN-S[2]

■ = Pins are up to 5V tolerant

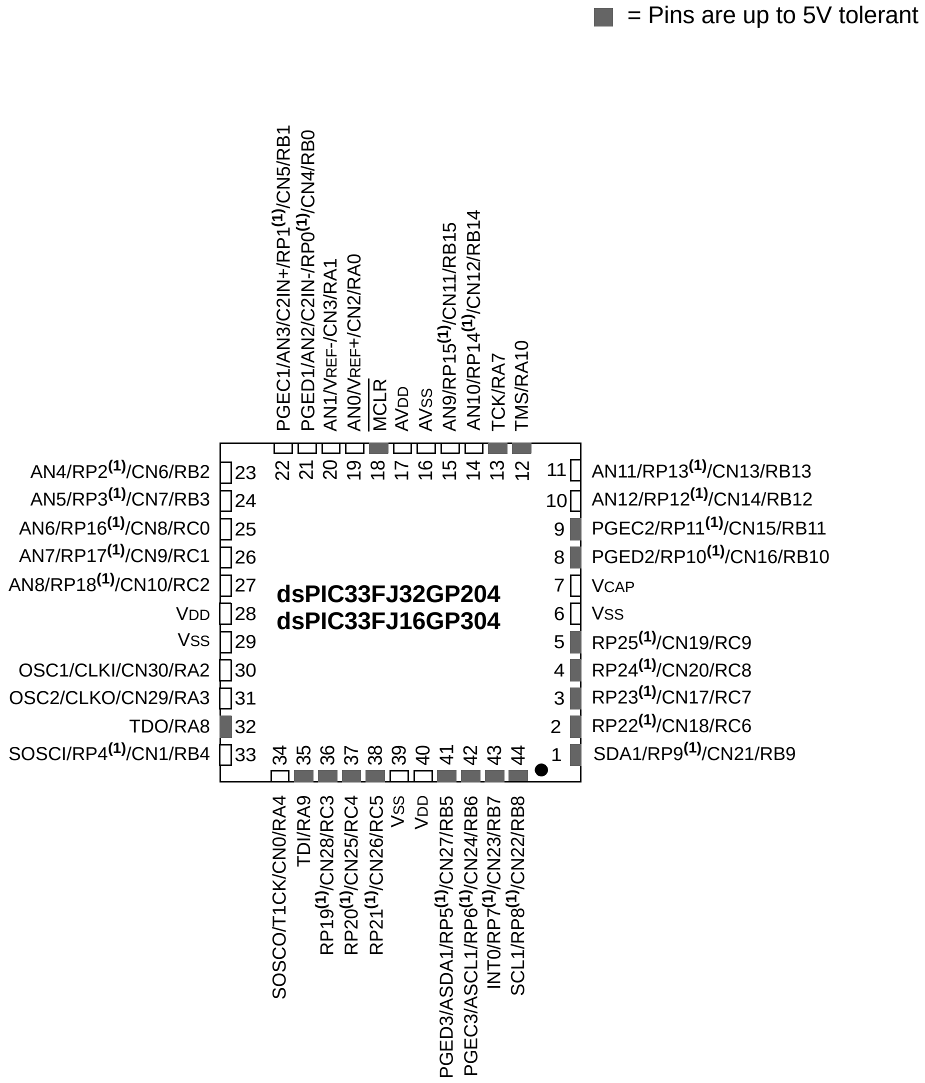


Note 1: The RPn pins can be used by any remappable peripheral. See Table 1 for the list of available peripherals.

2: 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)

44-Pin QFN(2)



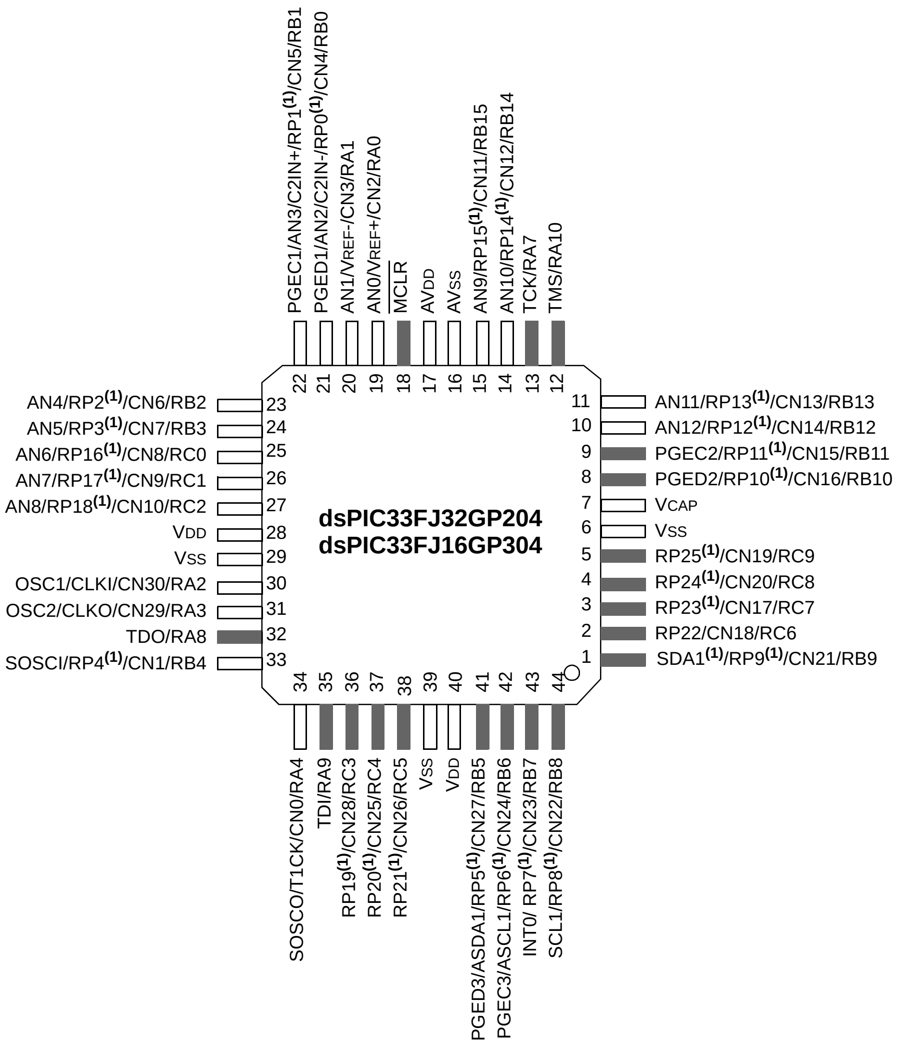
Note 1: The RPn pins can be used by any remappable peripheral. See Table 1 for the list of available peripherals.

2: 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)

44-Pin TQFP

■ = Pins are up to 5V tolerant



Note 1: The RPn pins can be used by any remappable peripheral. See Table 1 for the list of available peripherals.

Table of Contents

1.0 Device Overview 11
2.0 Guidelines for Getting Started with 16-bit Digital Signal Controllers 17
3.0 CPU 21
4.0 Memory Organization 33
5.0 Flash Program Memory 59
6.0 Resets 65
7.0 Interrupt Controller 73
8.0 Oscillator Configuration 101
9.0 Power-Saving Features 111
10.0 I/O Ports 115
11.0 Timer1 137
12.0 Timer2/3 Feature 139
13.0 Input Capture 145
14.0 Output Compare 147
15.0 Serial Peripheral Interface (SPI) 151
16.0 Inter-Integrated Circuit™ (I^2C™) 157
17.0 Universal Asynchronous Receiver Transmitter (UART) 165
18.0 10-bit/12-bit Analog-to-Digital Converter (ADC) 171
19.0 Special Features 185
20.0 Instruction Set Summary 193
21.0 Development Support 201
22.0 Electrical Characteristics 205
23.0 High Temperature Electrical Characteristics 249
24.0 Packaging Information 259
Appendix A: Revision History 273
Index 283
The Microchip Web Site 287
Customer Change Notification Service 287
Customer Support 287
Reader Response 288
Product Identification System 289

TO OUR VALUED CUSTOMERS

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

If you have any questions or comments regarding this publication, please contact the Marketing Communications Department via E-mail at **docerrors@microchip.com** or fax the **Reader Response Form** in the back of this data sheet to (480) 792-4150. We welcome your feedback.

Most Current Data Sheet

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

http://www.microchip.com

You can determine the version of a data sheet by examining its literature number found on the bottom outside corner of any page. The last character of the literature number is the version number, (e.g., DS30000A is version A of document DS30000).

Errata

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

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

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

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

Customer Notification System

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

NOTES:

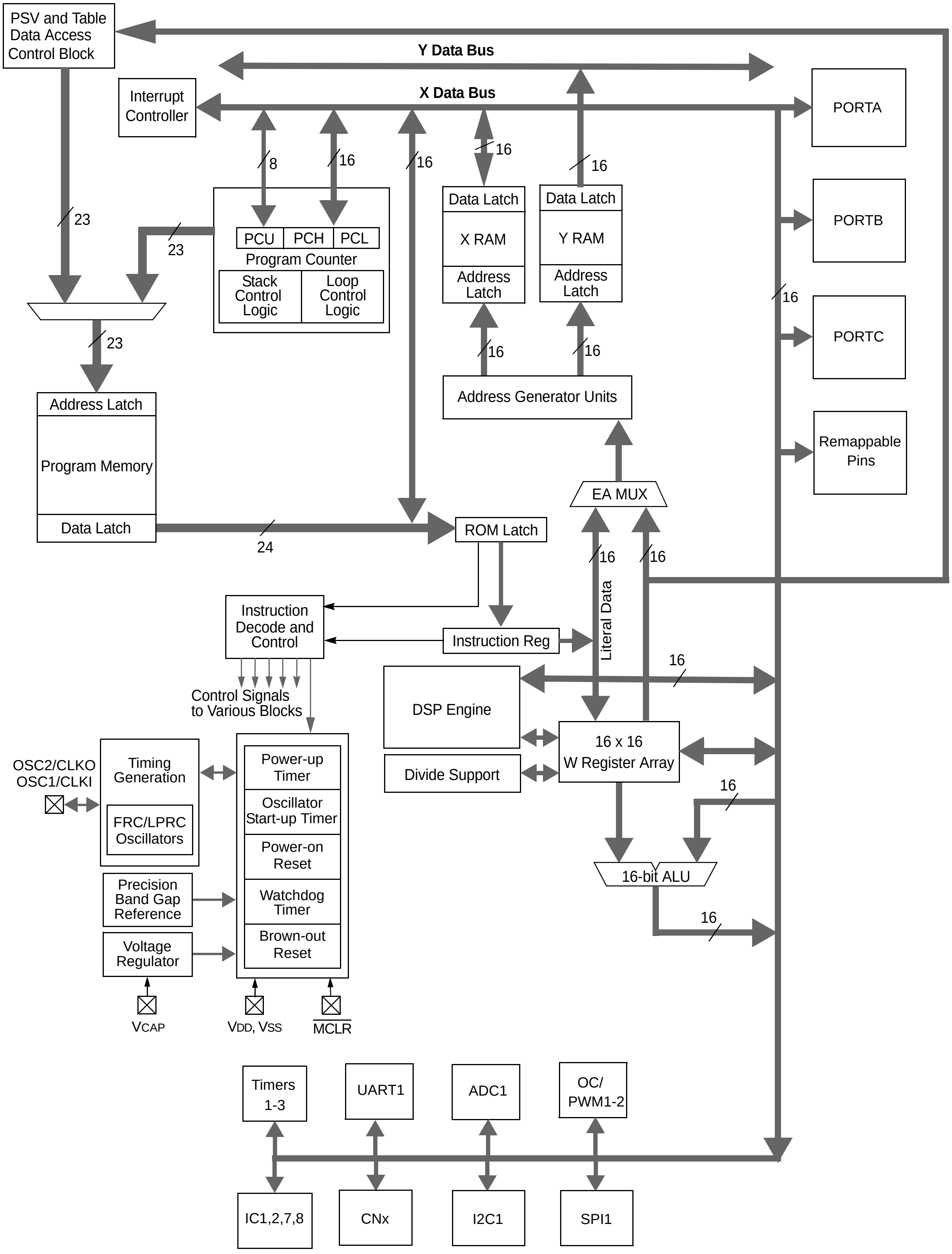
1.0 DEVICE OVERVIEW

> **Note 1:** This data sheet summarizes the features of the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 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) for the latest dsPIC33F/PIC24H Family Reference Manual sections.
>
> **2:** Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

This document contains device-specific information for the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 Digital Signal Controller (DSC) devices. The dsPIC33F 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 dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 family of devices. Table 1-1 lists the functions of the various pins shown in the pinout diagrams.

FIGURE 1-1: dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 BLOCK DIAGRAM



Note: Not all pins or features are implemented on all device pinout configurations. See **"Pin Diagrams"** for the specific pins and features present on each device.

TABLE 1-1: PINOUT I/O DESCRIPTIONS

Pin Name	Pin Type	Buffer Type	PPS	Description
AN0-AN12	I	Analog	No	Analog input channels.
CLKI CLKO	I O	ST/CMOS —	No No	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 —	No No	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 —	No No	32.768 kHz low-power oscillator crystal input; CMOS otherwise. 32.768 kHz low-power oscillator crystal output.
CN0-CN30	I	ST	No	Change notification inputs. Can be software programmed for internal weak pull-ups on all inputs.
IC1-IC2 IC7-IC8	I	ST	Yes Yes	Capture inputs 1/2. Capture inputs 7/8.
OCFA OC1-OC2	I O	ST —	Yes Yes	Compare Fault A input (for Compare Channels 1 and 2). Compare outputs 1 through 2.
INT0 INT1 INT2	I I I	ST ST ST	No Yes Yes	External interrupt 0. External interrupt 1. External interrupt 2.
RA0-RA4 RA7-RA10	I/O	ST	No No	PORTA is a bidirectional I/O port.
RB0-RB15	I/O	ST	No	PORTB is a bidirectional I/O port.
RC0-RC9	I/O	ST	No	PORTC is a bidirectional I/O port.
T1CK T2CK T3CK	I I I	ST ST ST	No Yes Yes	Timer1 external clock input. Timer2 external clock input. Timer3 external clock input.
$\overline{\text{U1CTS}}$ $\overline{\text{U1RTS}}$ U1RX U1TX	I O I O	ST — ST —	Yes Yes Yes Yes	UART1 clear to send. UART1 ready to send. UART1 receive. UART1 transmit.
SCK1 SDI1 SDO1 $\overline{\text{SS1}}$	I/O I O I/O	ST ST — ST	Yes Yes Yes Yes	Synchronous serial clock input/output for SPI1. SPI1 data in. SPI1 data out. SPI1 slave synchronization or frame pulse I/O.
SCL1 SDA1 ASCL1 ASDA1	I/O I/O I/O I/O	ST ST ST ST	No No No No	Synchronous serial clock input/output for I2C1. Synchronous serial data input/output for I2C1. Alternate synchronous serial clock input/output for I2C1. Alternate synchronous serial data input/output for I2C1.
TMS TCK TDI TDO	I I I O	ST ST ST —	No No No No	JTAG Test mode select pin. JTAG test clock input pin. JTAG test data input pin. JTAG test data output pin.

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

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

Pin Name	Pin Type	Buffer Type	PPS	Description
PGED1	I/O	ST	No	Data I/O pin for programming/debugging communication channel 1.
PGEC1	I	ST	No	Clock input pin for programming/debugging communication channel 1.
PGED2	I/O	ST	No	Data I/O pin for programming/debugging communication channel 2.
PGEC2	I	ST	No	Clock input pin for programming/debugging communication channel 2.
PGED3	I/O	ST	No	Data I/O pin for programming/debugging communication channel 3.
PGEC3	I	ST	No	Clock input pin for programming/debugging communication channel 3.
V_{CAP}	P	—	No	CPU logic filter capacitor connection.
V_{SS}	P	—	No	Ground reference for logic and I/O pins.
$V_{REF}+$	I	Analog	No	Analog voltage reference (high) input.
$V_{REF}-$	I	Analog	No	Analog voltage reference (low) input.
AV_{DD}	P	P	No	Positive supply for analog modules. This pin must be connected at all times.
$\overline{MCLR}$	I/P	ST	No	Master Clear (Reset) input. This pin is an active-low Reset to the device.
Avss	P	P	No	Ground reference for analog modules.
V_{DD}	P	—	No	Positive supply for peripheral logic and I/O pins.

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

1.1 Referenced Sources

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

> **Note 1:** To access the documents listed below, browse to the documentation section of the dsPIC33FJ32GP204 product page of the Microchip web site (www.microchip.com) or select a family reference manual section from the following list.
>
> In addition to parameters, features, and other documentation, the resulting page provides links to the related family reference manual sections.

- **Section 1. "Introduction"** (DS70197)
- **Section 2. "CPU"** (DS70204)
- **Section 3. "Data Memory"** (DS70202)
- **Section 4. "Program Memory"** (DS70202)
- **Section 5. "Flash Programming"** (DS70191)
- **Section 6. "Interrupts** (DS70184)
- **Section 7. "Oscillator"** (DS70186)
- **Section 8. "Reset"** (DS70192)
- **Section 9. "Watchdog Timer and Power-Saving Modes"** (DS70196)
- **Section 10. "I/O Ports"** (DS70193)
- **Section 11. "Timers"** (DS70205)
- **Section 12. "Input Capture"** (DS70198)
- **Section 13. "Output Compare"** (DS70209)
- **Section 16. "Analog-to-Digital Converter (ADC)"** (DS70183)
- **Section 17. "UART"** (DS70188)
- **Section 18. "Serial Peripheral Interface (SPI)"** (DS70206)
- **Section 19. "Inter-Integrated Circuit™ (I^2C™)"** (DS70195)
- **Section 23. "CodeGuard™ Security"** (DS70199)
- **Section 25. "Device Configuration"** (DS70194)

NOTES:

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

Note 1: This data sheet summarizes the features of the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 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) 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 dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 family of 16-bit Digital Signal Controllers (DSCs) 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 (even if ADC module is not used) (see **Section 2.2 "Decoupling Capacitors"**)
- VCAP (see **Section 2.3 "CPU Logic Filter Capacitor Connection (VCAP)"**)
- $\overline{\text{MCLR}}$ pin (see **Section 2.4 "Master Clear (MCLR) Pin"**)
- PGECx/PGEDx pins used for In-Circuit Serial Programming™ (ICSP™) and debugging purposes (see **Section 2.5 "ICSP Pins"**)
- OSC1 and OSC2 pins when external oscillator source is used (see **Section 2.6 "External Oscillator Pins"**)

Additionally, the following pins may be required:

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

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

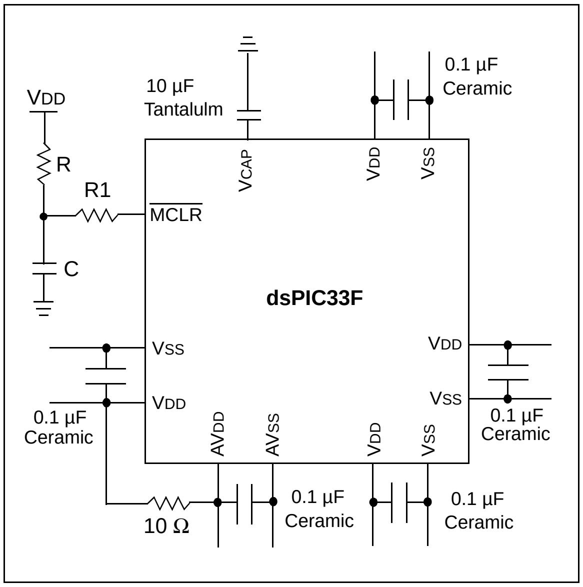
2.2 Decoupling Capacitors

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

Consider the following criteria when using decoupling capacitors:

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

FIGURE 2-1: RECOMMENDED MINIMUM CONNECTION



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 CPU Logic Filter Capacitor Connection (VCAP)

A low-ESR (< 5 Ohms) capacitor is required on the VCAP pin, which is used to stabilize the voltage regulator output voltage. The VCAP pin must not be connected to VDD, and must have a capacitor between 4.7 µF and 10 µF, 16V connected to ground. The type can be ceramic or tantalum. Refer to **Section 22.0 "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 19.2 "On-Chip Voltage Regulator"** for details.

2.4 Master Clear ($\overline{\text{MCLR}}$) Pin

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

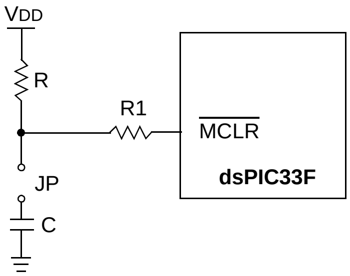
- Device Reset
- Device programming and debugging

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

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

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

FIGURE 2-2: EXAMPLE OF $\overline{\text{MCLR}}$ PIN CONNECTIONS



Note 1: R ≤ 10 kΩ is recommended. A suggested starting value is 10 kΩ Ensure that the $\overline{\text{MCLR}}$ pin VIH and VIL specifications are met.

2: R1 ≤ 470Ω will limit any current flowing into $\overline{\text{MCLR}}$ from the external capacitor C, in the event of $\overline{\text{MCLR}}$ pin breakdown, due to Electrostatic Discharge (ESD) or Electrical Overstress (EOS). Ensure that the $\overline{\text{MCLR}}$ pin VIH and VIL specifications are met.

2.5 ICSP Pins

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

Pull-up resistors, series diodes and capacitors on the PGECx and PGEDx pins are not recommended as they will interfere with the programmer/debugger communications to the device. If such discrete components are an application requirement, they should be removed from the circuit during programming and debugging. Alternatively, refer to the AC/DC characteristics and timing requirements information in the respective device Flash programming specification for information on capacitive loading limits and pin input voltage high (V_{IH}) and input low (V_{IL}) 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 2, MPLAB ICD 3, or MPLAB REAL ICE™ in-circuit emulator.

For more information on MPLAB ICD 2, MPLAB ICD 3, or MPLAB REAL ICE™ in-circuit emulator connection requirements, refer to the following documents that are available on the Microchip website.

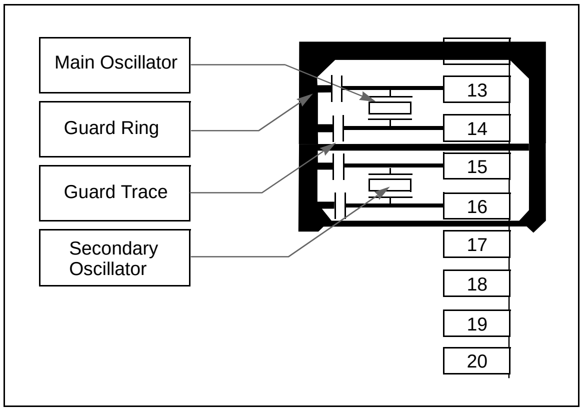
- *"MPLAB® ICD 2 In-Circuit Debugger User's Guide"* DS51331
- *"Using MPLAB® ICD 2"* (poster) DS51265
- *"MPLAB® ICD 2 Design Advisory"* DS51566
- *"Using MPLAB® ICD 3"* (poster) DS51765
- *"MPLAB® ICD 3 Design Advisory"* DS51764
- *"MPLAB® REAL ICE™ In-Circuit Emulator User's Guide"* DS51616
- *"Using MPLAB® REAL ICE™"* (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 8.0 "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 ≤8 MHz for start-up with PLL enabled to comply with device PLL start-up conditions. This means that if the external oscillator frequency is outside this range, the application must start-up in 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 2, MPLAB ICD 3, or MPLAB REAL ICE in-circuit emulator is selected as a debugger, it automatically initializes all of the A/D input pins (ANx) as “digital” pins, by setting all bits in the AD1PCFGL register.

The bits in the registers that correspond to the A/D pins that are initialized by MPLAB ICD 2, MPLAB ICD 3, or MPLAB REAL ICE in-circuit emulator, must not be cleared by the user application firmware; otherwise, communication errors will result between the debugger and the device.

If your application needs to use certain A/D pins as analog input pins during the debug session, the user application must clear the corresponding bits in the AD1PCFGL register during initialization of the ADC module.

When MPLAB ICD 2, MPLAB ICD 3, or MPLAB REAL ICE in-circuit emulator is used as a programmer, the user application firmware must correctly configure the AD1PCFGL register. Automatic initialization of this register is only done during debugger operation. Failure to correctly configure the register(s) will result in all A/D pins being recognized as analog input pins, resulting in the port value being read as a logic ‘`0`’, which may affect user application functionality.

2.9 Unused I/Os

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

Alternatively, connect a 1k to 10k resistor between V_{SS} and the unused pins.

3.0 CPU

Note 1: This data sheet summarizes the features of the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 family 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) of the *"dsPIC33F/PIC24H Family Reference Manual"*, which is available from the Microchip website (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 dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 CPU module has a 16-bit (data) modified Harvard architecture with an enhanced instruction set, including significant support for DSP. The CPU has a 24-bit instruction word with a variable length opcode field. The Program Counter (PC) is 23 bits wide and addresses up to 4M x 24 bits of user program memory space. The actual amount of program memory implemented varies by device. A single-cycle instruction prefetch mechanism is used to help maintain throughput and provides predictable execution. All instructions execute in a single cycle, with the exception of instructions that change the program flow, the double word move (`MOV.D`) instruction and the table instructions. Overhead-free program loop constructs are supported using the `DO` and `REPEAT` instructions, both of which are interruptible at any point.

The dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 devices have sixteen, 16-bit working registers in the programmer's model. Each of the working registers can serve as a data, address or address offset register. The 16th working register (W15) operates as a software Stack Pointer (SP) for interrupts and calls.

The dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 instruction set has two classes of instructions: MCU and DSP. These two instruction classes are seamlessly integrated into a single CPU. The instruction set includes many addressing modes and is designed for optimum C compiler efficiency. For most instructions, the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 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. The programmer's model for the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 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 dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 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 dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 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: dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 CPU CORE BLOCK DIAGRAM

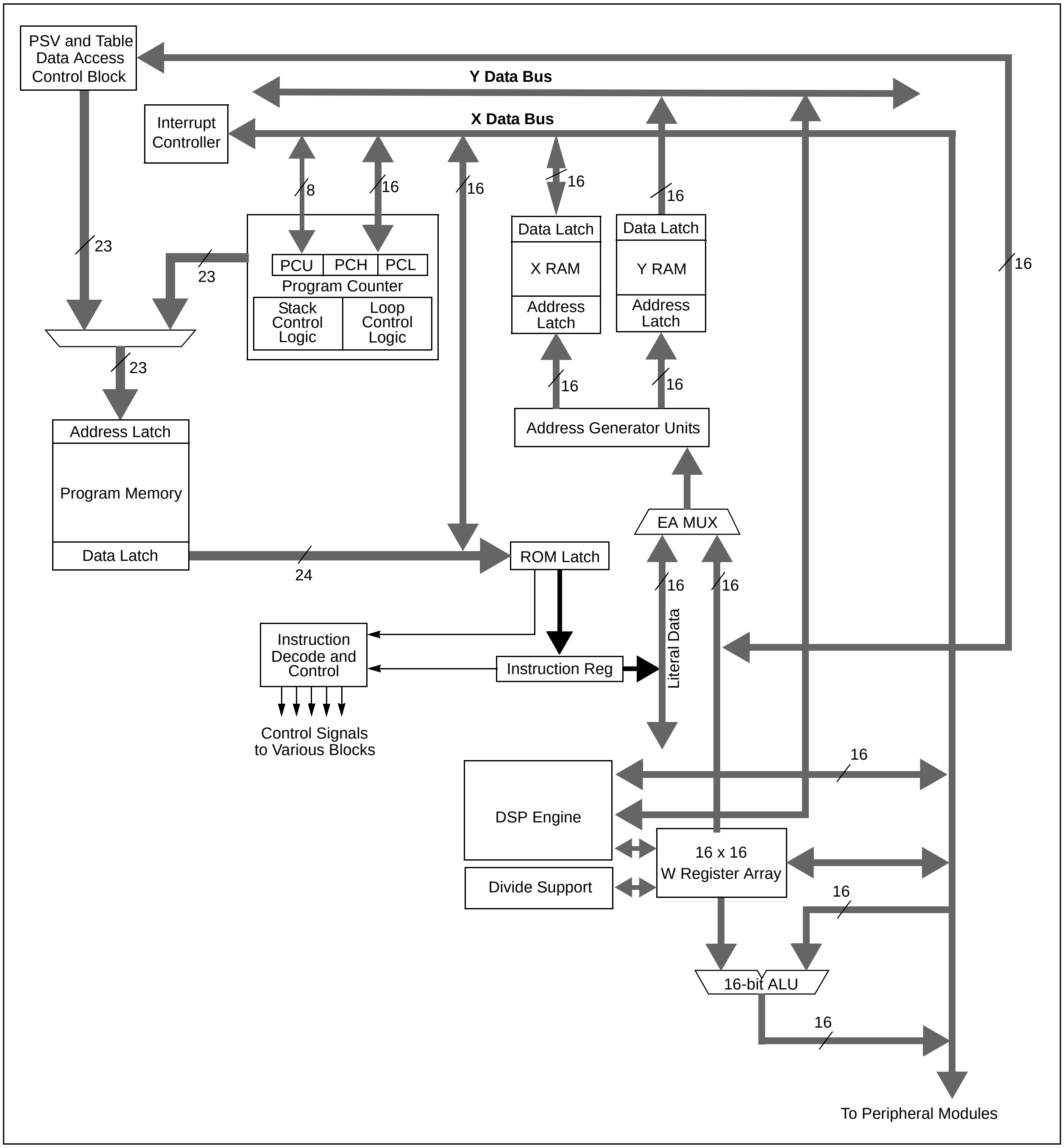
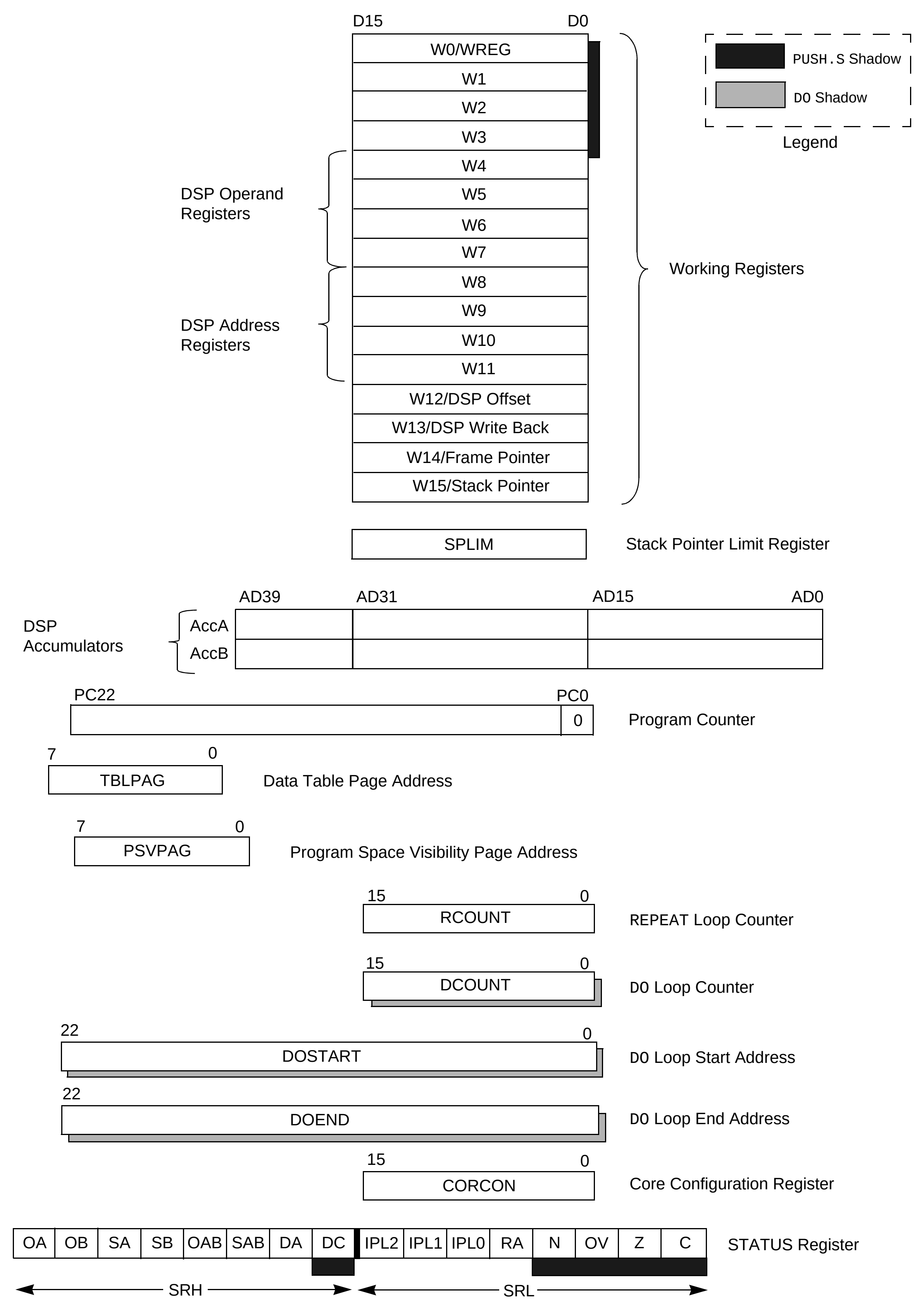


FIGURE 3-2: dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 PROGRAMMER'S MODEL



3.4 CPU Control Registers

CPU control registers include:

- **Register 3-1: “SR: CPU Status Register”**
- **Register 3-2: “CORCON: CORE Control Register”**

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(1)	SB(1)	OAB	SAB	DA	DC
bit 15							bit 8

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

Legend:		
C = Clear only bit	R = Readable bit	U = Unimplemented bit, read as ‘0’
S = Set only bit	W = Writable bit	-n = Value at POR
‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15	**OA:** Accumulator A Overflow Status bit `1` = Accumulator A overflowed `0` = Accumulator A has not overflowed
bit 14	**OB:** Accumulator B Overflow Status bit `1` = Accumulator B overflowed `0` = Accumulator B has not overflowed
bit 13	**SA:** Accumulator A Saturation ‘Sticky’ Status bit(1) `1` = Accumulator A is saturated or has been saturated at some time `0` = Accumulator A is not saturated
bit 12	**SB:** Accumulator B Saturation ‘Sticky’ Status bit(1) `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 `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 **Note:** This bit can be read or cleared (not set). Clearing this bit will clear SA and SB.
bit 9	**DA:** `DO` Loop Active bit `1` = `DO` loop in progress `0` = `DO` loop not in progress

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. 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>).

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

bit 8	**DC:** MCU ALU Half Carry/$\overline{\text{Borrow}}$ 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
bit 7-5	**IPL<2:0>:** CPU Interrupt Priority Level Status bits**(2)** 111 = CPU Interrupt Priority Level is 7 (15), user interrupts disabled 110 = CPU Interrupt Priority Level is 6 (14) 101 = CPU Interrupt Priority Level is 5 (13) 100 = CPU Interrupt Priority Level is 4 (12) 011 = CPU Interrupt Priority Level is 3 (11) 010 = CPU Interrupt Priority Level is 2 (10) 001 = CPU Interrupt Priority Level is 1 (9) 000 = CPU Interrupt Priority Level is 0 (8)
bit 4	**RA:** REPEAT Loop Active bit 1 = REPEAT loop in progress 0 = REPEAT loop not in progress
bit 3	**N:** MCU ALU Negative bit 1 = Result was negative 0 = Result was non-negative (zero or positive)
bit 2	**OV:** MCU ALU Overflow bit This bit is used for signed arithmetic (2’s complement). It indicates an overflow of 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	**Z:** 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	**C:** MCU ALU Carry/$\overline{\text{Borrow}}$ 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. 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>).

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(1)	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(2)	PSV	RND	IF
bit 7							bit 0

Legend:	C = Clear only 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 `DO` Loop Termination Control bit(1)

`1` = Terminate executing `DO` loop at end of current loop iteration
`0` = No effect

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

`111` = 7 `DO` loops active
•
•
•
`001` = 1 `DO` loop active
`000` = 0 `DO` loops active

bit 7 **SATA:** AccA Saturation Enable bit

`1` = Accumulator A saturation 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(2)

`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 dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 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.

The dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 CPU incorporates hardware support for both multiplication and division. This includes a dedicated hardware multiplier and support hardware for 16-bit-divisor division.

Refer to the "*dsPIC30F/33F Programmer's Reference Manual*" (DS70157) for information on the SR bits affected by each instruction.

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/subtracter (with two target accumulators, round and saturation logic).

The dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 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 (e.g., `ED`, `EDAC`).

The DSP engine can also perform 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:

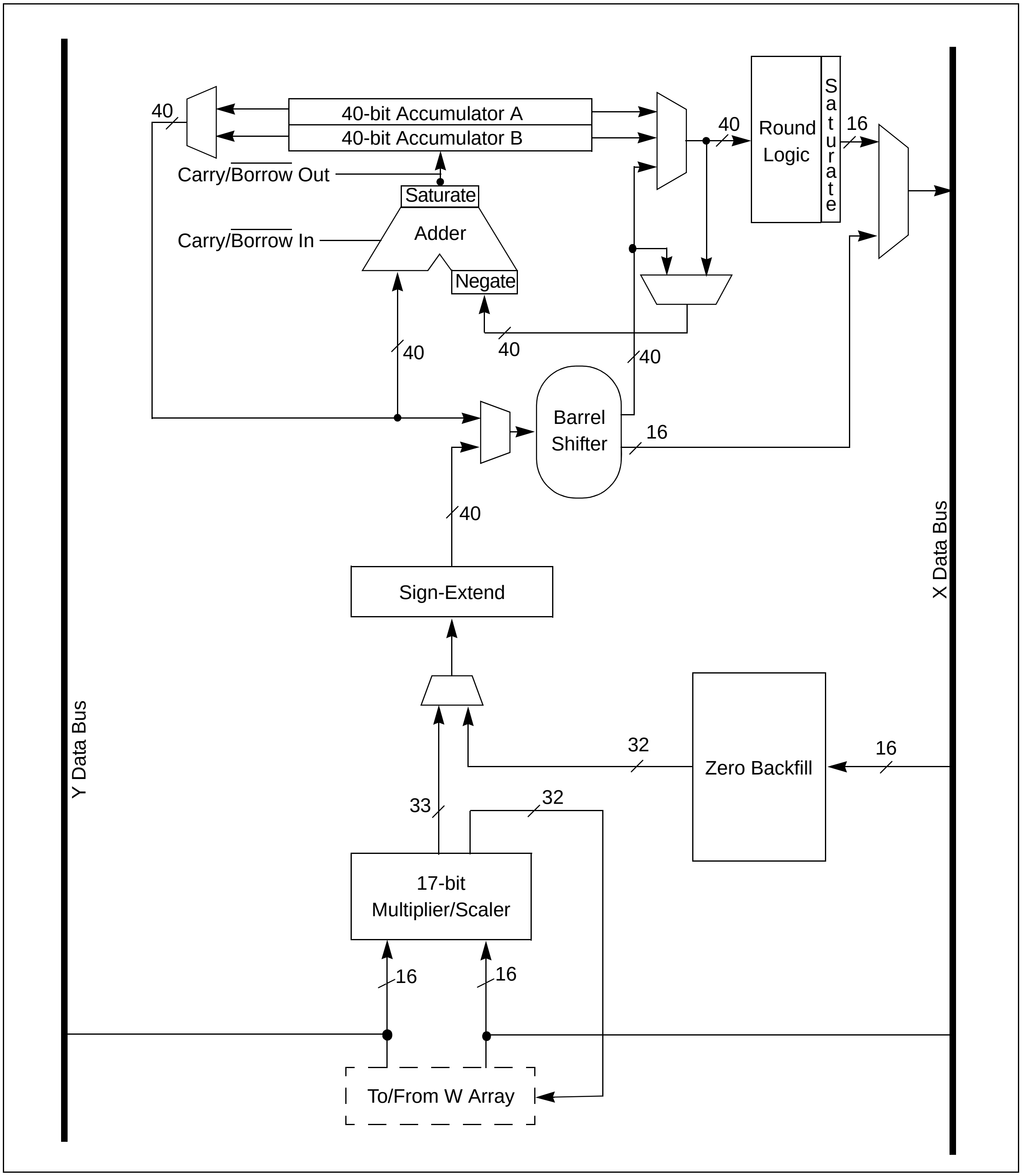
- Fractional or integer DSP multiply (IF)
- Signed or unsigned DSP multiply (US)
- Conventional or convergent rounding (RND)
- Automatic saturation on/off for AccA (SATA), AccB (SATB) and writes to data memory (SATDW)
- Accumulator Saturation mode selection (ACCSAT)

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`	$A = x \bullet y$	No
`MPY`	$A = x^2$	No
`MPY.N`	$A = - x \bullet y$	No
`MSC`	$A = A - x \bullet y$	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 3.01518×10^{-5}. In Fractional mode, the 16 x 16 multiply operation generates a 1.31 product that has a precision of 4.65661×10^{-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/subtracter with automatic sign extension logic. It can select one of two accumulators (A or B) as its pre-accumulation source and post-accumulation destination. For the `ADD` and `LAC` instructions, the data to be accumulated or loaded can be optionally scaled using the barrel shifter prior to accumulation.

3.6.2.1 Adder/Subtracter, Overflow and Saturation

The adder/subtracter is a 40-bit adder with an optional zero input into one side, and either true or complement data into the other input.

- In the case of addition, the Carry/$\overline{\text{Borrow}}$ input is active-high and the other input is true data (not complemented).
- In the case of subtraction, the Carry/$\overline{\text{Borrow}}$ input is active-low and the other input is complemented.

The adder/subtracter generates Overflow Status bits, SA/SB and OA/OB, which are latched and reflected in the STATUS register:

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

The adder has an additional saturation block 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 have been provided to 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/subtracter. When set, they indicate that the most recent operation has overflowed into the accumulator guard bits (bits 32 through 39). The OA and OB bits can also optionally generate an arithmetic warning trap when 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/subtracter, 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). Programs 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 (0x7FFFFFFFFF) or maximally negative 9.31 value (0x8000000000) into the target accumulator. The SA or SB bit is set and remains set until cleared by the user 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.

3.6.2.2 Accumulator ‘Write Back’

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

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

3.6.2.3 Round Logic

The round logic is a combinational block 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 (lsw) is simply discarded.

Conventional rounding zero-extends bit 15 of the accumulator and adds it to the ACCxH word (bits 16 through 31 of the accumulator).

- If the ACCxL word (bits 0 through 15 of the accumulator) is between 0x8000 and 0xFFFF (0x8000 included), ACCxH is incremented.
- If ACCxL is between 0x0000 and 0x7FFF, ACCxH is left unchanged.

A consequence of this algorithm is that over a succession of random rounding operations, the value tends to be biased slightly positive.

Convergent (or unbiased) rounding operates in the same manner as conventional rounding, except when ACCxL equals 0x8000. In this case, the Least Significant bit (bit 16 of the accumulator) of ACCxH is examined.

- If it is ‘`1`’, ACCxH is incremented.
- If it is ‘`0`’, ACCxH is not modified. Assuming that bit 16 is effectively random in nature, this scheme removes any rounding bias that may accumulate.

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

3.6.2.4 Data Space Write Saturation

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

If the SATDW bit in the CORCON register is set, data (after rounding or truncation) is tested for overflow and adjusted accordingly:

- For input data greater than 0x007FFF, data written to memory is forced to the maximum positive 1.15 value, 0x7FFF.
- For input data less than 0xFF8000, data written to memory is forced to the maximum negative 1.15 value, 0x8000.

The Most significant bit of the source (bit 39) is used to determine the sign of the operand being tested.

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

3.6.3 BARREL SHIFTER

The barrel shifter 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 dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 4. “Program Memory”** (DS70202) of the *“dsPIC33F/PIC24H Family Reference Manual”*, which is available from the Microchip website (www.microchip.com).

The dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 architecture features separate program and data memory spaces and buses. This architecture also allows the direct access of program memory from the data space during code execution.

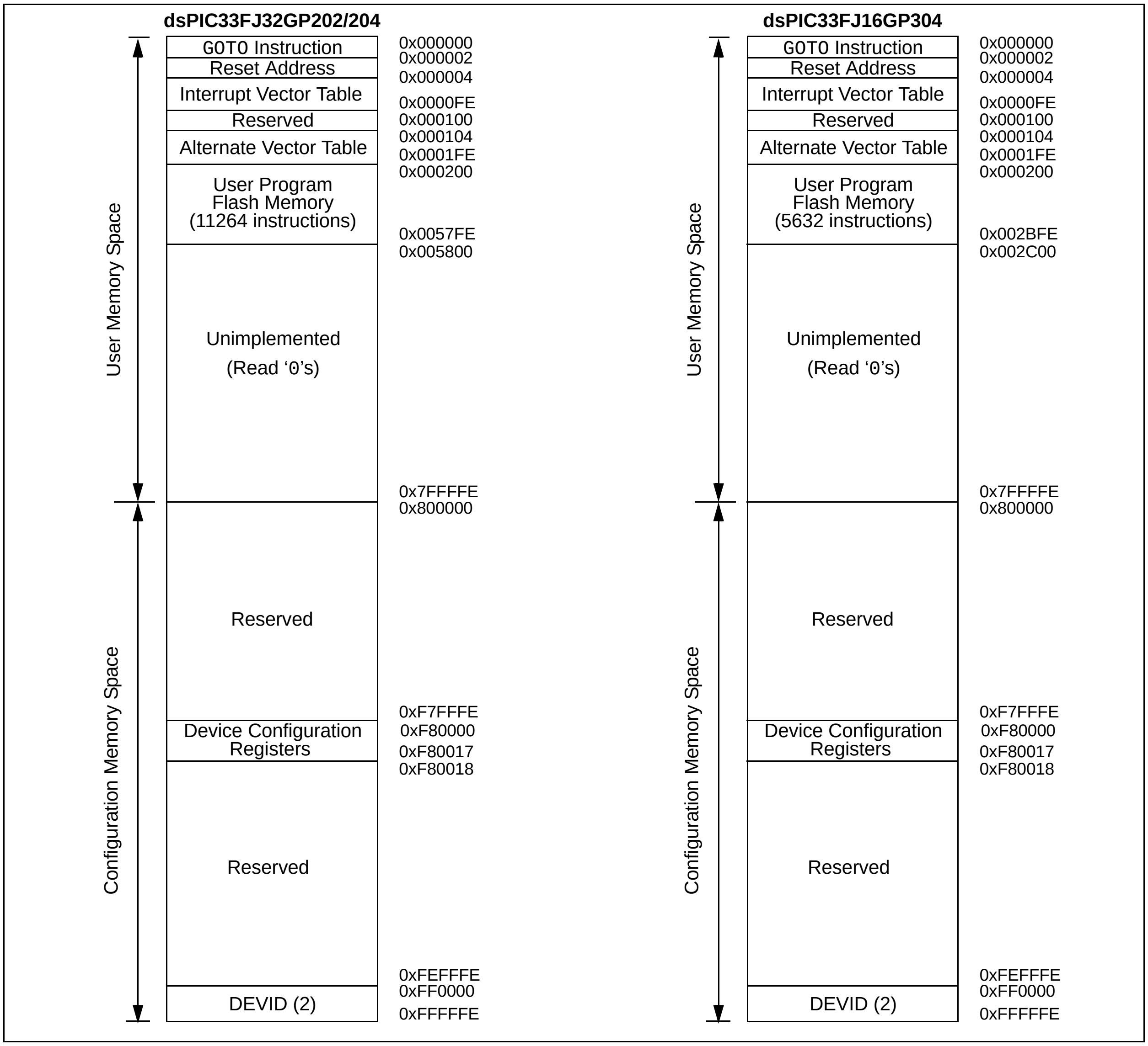
4.1 Program Address Space

The program address memory space of the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 devices 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 0x7FFFFF). The exception is the use of `TBLRD/TBLWT` operations, which use TBLPAG<7> to permit access to the Configuration bits and Device ID sections of the configuration memory space.

The memory maps for the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 devices are shown in Figure 4-1.

FIGURE 4-1: PROGRAM MEMORY FOR dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 DEVICES



4.1.1 PROGRAM MEMORY ORGANIZATION

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

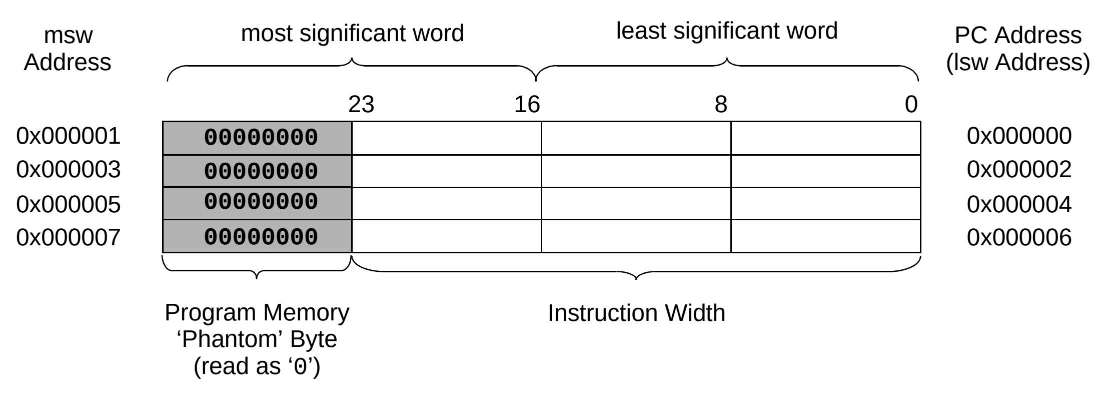
Program memory addresses are always word-aligned on the lower word, and addresses are incremented or decremented by two during code execution. This arrangement provides compatibility with data memory space addressing and makes data in the program memory space accessible.

4.1.2 INTERRUPT AND TRAP VECTORS

All dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 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.

dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 devices also have two interrupt vector tables, located from 0x000004 to 0x0000FF and 0x000100 to 0x0001FF. These vector tables allow each of the many device interrupt sources to be handled by separate Interrupt Service Routines (ISRs). A more detailed discussion of the interrupt vector tables is provided in **Section 7.1 “Interrupt Vector Table”**.

FIGURE 4-2: PROGRAM MEMORY ORGANIZATION



4.2 Data Address Space

The dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 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"**).

dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 devices implement up to 2 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® MCU devices and improve data space memory usage efficiency, the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 instruction set supports both word and byte operations. As a consequence of byte accessibility, all effective address calculations are internally scaled to step through word-aligned memory. For example, the core recognizes that Post-Modified Register Indirect Addressing mode [Ws++] will result in a value of Ws + 1 for byte operations and Ws + 2 for word operations.

Data byte reads will read the complete word that contains the byte, using the LSB of any EA to determine which byte to select. The selected byte is placed onto the LSB of the data path. That is, data memory and registers are organized as two parallel byte-wide entities with shared (word) address decode but separate write lines. Data byte writes only write to the corresponding side of the array or register 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 instruction 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 users 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 dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 core and peripheral modules for controlling the operation of the device.

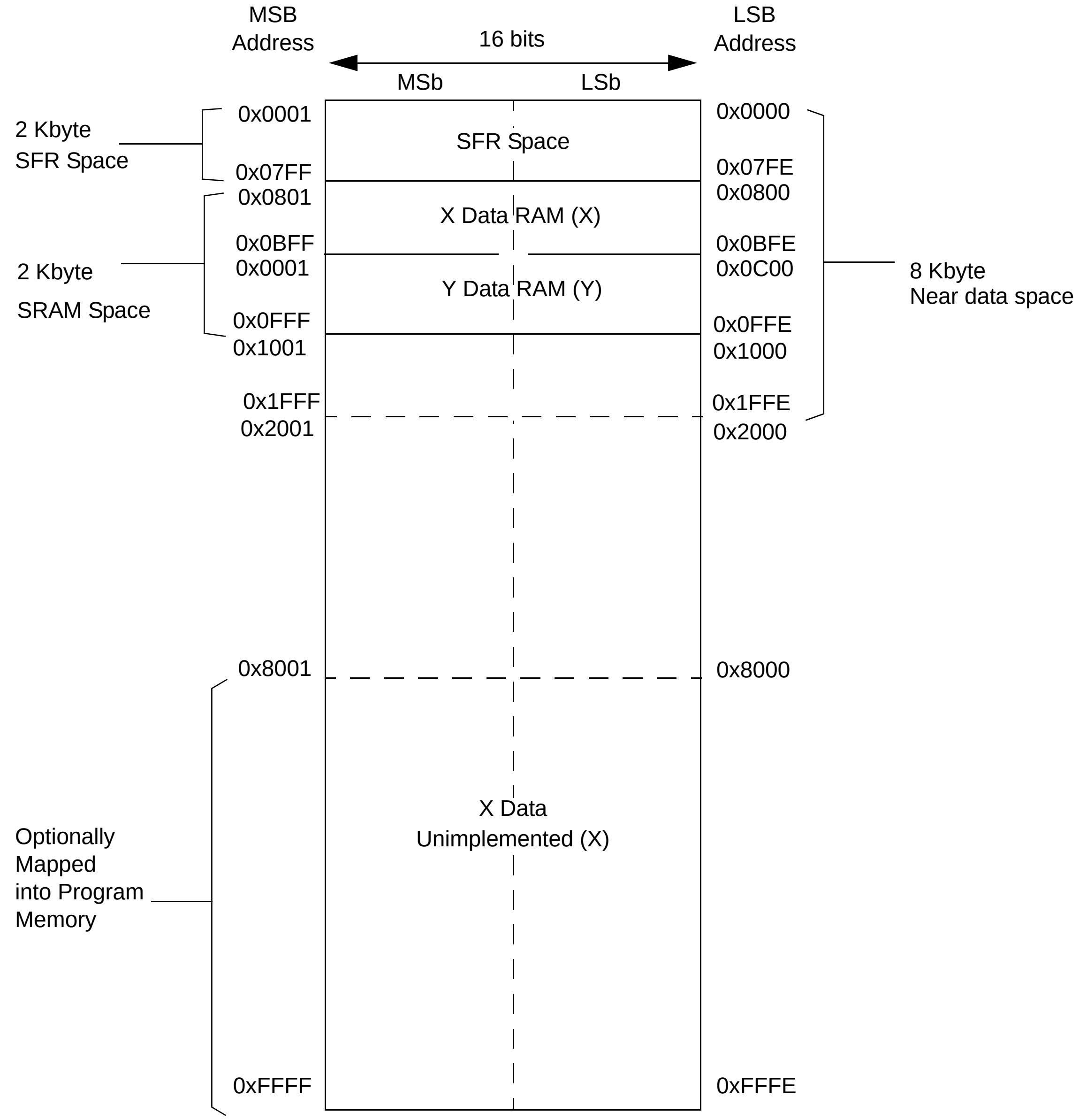
SFRs are distributed among the modules that they control, and are generally grouped together by module. Much of the SFR space contains unused addresses; these are read as '`0`'. A complete listing of implemented SFRs, including their addresses, is shown in Table 4-1 through Table 4-22.

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

4.2.4 NEAR DATA SPACE

The 8 Kbyte area between 0x0000 and 0x1FFF is referred to as the Near Data Space. Locations in this space are directly addressable via a 13-bit absolute address field within all memory direct instructions. Additionally, the whole data space is addressable using `MOV` instructions, which support Memory Direct Addressing mode with a 16-bit address field, or by using Indirect Addressing mode using a working register as an address pointer.

FIGURE 4-3: DATA MEMORY MAP FOR dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 DEVICES WITH 2 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`, `MOVSAC`, `MPY`, `MPY.N` and `MSC`) to provide two concurrent data read paths.

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

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

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

TABLE 4-1: CPU CORE REGISTERS MAP

SFR Name	SFR Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
WREG0	0000	Working Register 0																`0000`
WREG1	0002	Working Register 1																`0000`
WREG2	0004	Working Register 2																`0000`
WREG3	0006	Working Register 3																`0000`
WREG4	0008	Working Register 4																`0000`
WREG5	000A	Working Register 5																`0000`
WREG6	000C	Working Register 6																`0000`
WREG7	000E	Working Register 7																`0000`
WREG8	0010	Working Register 8																`0000`
WREG9	0012	Working Register 9																`0000`
WREG10	0014	Working Register 10																`0000`
WREG11	0016	Working Register 11																`0000`
WREG12	0018	Working Register 12																`0000`
WREG13	001A	Working Register 13																`0000`
WREG14	001C	Working Register 14																`0000`
WREG15	001E	Working Register 15																`0800`
SPLIM	0020	Stack Pointer Limit Register																`xxxx`
ACCAL	0022	Accumulator A Low Word Register																`0000`
ACCAH	0024	Accumulator A High Word Register																`0000`
ACCAU	0026	Accumulator A Upper Word Register																`0000`
ACCBL	0028	Accumulator B Low Word Register																`0000`
ACCBH	002A	Accumulator B High Word Register																`0000`
ACCBU	002C	Accumulator B Upper Word Register																`0000`
PCL	002E	Program Counter Low Word Register																`0000`
PCH	0030	—	—	—	—	—	—	—	—	Program Counter High Byte Register								`0000`
TBLPAG	0032	—	—	—	—	—	—	—	—	Table Page Address Pointer Register								`0000`
PSVPAG	0034	—	—	—	—	—	—	—	—	Program Memory Visibility Page Address Pointer Register								`0000`
RCOUNT	0036	Repeat Loop Counter Register																`xxxx`
DCOUNT	0038	DCOUNT<15:0>																`xxxx`
DOSTARTL	003A	DOSTARTL<15:1>															`0`	`xxxx`
DOSTARTH	003C	—	—	—	—	—	—	—	—	—	—	DOSTARTH<5:0>						`00xx`
DOENDL	003E	DOENDL<15:1>															`0`	`xxxx`
DOENDH	0040	—	—	—	—	—	—	—	—	—	—	DOENDH						`00xx`
SR	0042	OA	OB	SA	SB	OAB	SAB	DA	DC	IPL2	IPL1	IPL0	RA	N	OV	Z	C	`0000`
CORCON	0044	—	—	—	US	EDT	DL<2:0>			SATA	SATB	SATDW	ACCSAT	IPL3	PSV	RND	IF	`0020`
MODCON	0046	XMODEN	YMODEN	—	—	BWM<3:0>				YWM<3:0>				XWM<3:0>				`0000`

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

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

SFR Name	SFR Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
XMODSRT	0048	XS<15:1>															0	xxxx
XMODEND	004A	XE<15:1>															1	xxxx
YMODSRT	004C	YS<15:1>															0	xxxx
YMODEND	004E	YE<15:1>															1	xxxx
XBREV	0050	BREN	XB<14:0>															xxxx
DISICNT	0052	—	—	Disable Interrupts Counter Register														xxxx

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

TABLE 4-2: CHANGE NOTIFICATION REGISTER MAP FOR dsPIC33FJ32GP202

SFR Name	SFR Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
CNEN1	0060	CN15IE	CN14IE	CN13IE	CN12IE	CN11IE	—-	—	—	CN7IE	CN6IE	CN5IE	CN4IE	CN3IE	CN2IE	CN1IE	CN0IE	0000
CNEN2	0062	—	CN30IE	CN29IE	—	CN27IE	—	—	CN24IE	CN23IE	CN22IE	CN21IE	—	—	—	—	CN16IE	0000
CNPU1	0068	CN15PUE	CN14PUE	CN13PUE	CN12PUE	CN11PUE	—	—	—	CN7PUE	CN6PUE	CN5PUE	CN4PUE	CN3PUE	CN2PUE	CN1PUE	CN0PUE	0000
CNPU2	006A	—	CN30PUE	CN29PUE	—	CN27PUE	—	—	CN24PUE	CN23PUE	CN22PUE	CN21PUE	—	—	—	—	CN16PUE	0000

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

TABLE 4-3: CHANGE NOTIFICATION REGISTER MAP FOR dsPIC33FJ32GP204 AND dsPIC33FJ16GP304

SFR Name	SFR Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
CNEN1	0060	CN15IE	CN14IE	CN13IE	CN12IE	CN11IE	CN10IE	CN9IE	CN8IE	CN7IE	CN6IE	CN5IE	CN4IE	CN3IE	CN2IE	CN1IE	CN0IE	0000
CNEN2	0062	—	CN30IE	CN29IE	CN28IE	CN27IE	CN26IE	CN25IE	CN24IE	CN23IE	CN22IE	CN21IE	CN20IE	CN19IE	CN18IE	CN17IE	CN16IE	0000
CNPU1	0068	CN15PUE	CN14PUE	CN13PUE	CN12PUE	CN11PUE	CN10PUE	CN9PUE	CN8PUE	CN7PUE	CN6PUE	CN5PUE	CN4PUE	CN3PUE	CN2PUE	CN1PUE	CN0PUE	0000
CNPU2	006A	—	CN30PUE	CN29PUE	CN28PUE	CN27PUE	CN26PUE	CN25PUE	CN24PUE	CN23PUE	CN22PUE	CN21PUE	CN20PUE	CN19PUE	CN18PUE	CN17PUE	CN16PUE	0000

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

TABLE 4-4: INTERRUPT CONTROLLER REGISTER MAP

SFR Name	SFR Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
INTCON1	0080	NSTDIS	OVAERR	OVBERR	COVAERR	COVBERR	OVATE	OVBTE	COVTE	SFTACERR	DIV0ERR	—	MATHERR	ADDRERR	STKERR	OSCFAIL	—	0000
INTCON2	0082	ALTIVT	DISI	—	—	—	—	—	—	—	—	—	—	—	INT2EP	INT1EP	INT0EP	0000
IFS0	0084	—	—	AD1IF	U1TXIF	U1RXIF	SPI1IF	SPI1EIF	T3IF	T2IF	OC2IF	IC2IF	—	T1IF	OC1IF	IC1IF	INT0IF	0000
IFS1	0086	—	—	INT2IF	—	—	—	—	—	IC8IF	IC7IF	—	INT1IF	CNIF	—	MI2C1IF	SI2C1IF	0000
IFS4	008C	—	—	—	—	—	—	—	—	—	—	—	—	—	—	U1EIF	—	0000
IEC0	0094	—	—	AD1IE	U1TXIE	U1RXIE	SPI1IE	SPI1EIE	T3IE	T2IE	OC2IE	IC2IE	—	T1IE	OC1IE	IC1IE	INT0IE	0000
IEC1	0096	—	—	INT2IE	—	—	—	—	—	IC8IE	IC7IE	—	INT1IE	CNIE	—	MI2C1IE	SI2C1IE	0000
IEC4	009C	—	—	—	—	—	—	—	—	—	—	—	—	—	—	U1EIE	—	0000
IPC0	00A4	—	T1IP<2:0>			—	OC1IP<2:0>			—	IC1IP<2:0>			—	INT0IP<2:0>			4444
IPC1	00A6	—	T2IP<2:0>			—	OC2IP<2:0>			—	IC2IP<2:0>			—	—	—	—	4440
IPC2	00A8	—	U1RXIP<2:0>			—	SPI1IP<2:0>			—	SPI1EIP<2:0>			—	T3IP<2:0>			4444
IPC3	00AA	—	—	—	—	—	—	—	—	—	AD1IP<2:0>			—	U1TXIP<2:0>			0044
IPC4	00AC	—	CNIP<2:0>			—	—	—	—	—	MI2C1IP<2:0>			—	SI2C1IP<2:0>			4044
IPC5	00AE	—	IC8IP<2:0>			—	IC7IP<2:0>			—	—	—	—	—	INT1IP<2:0>			4404
IPC7	00B2	—	—	—	—	—	—	—	—	—	INT2IP<2:0>			—	—	—	—	0040
IPC16	00C4	—	—	—	—	—	—	—	—	—	U1EIP<2:0>			—	—	—	—	0040
INTTREG	00E0	—	—	—	—	ILR<3:0>				—	VECNUM<6:0>							0000

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

TABLE 4-5: TIMER REGISTER MAP

SFR Name	SFR Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
TMR1	0100	Timer1 Register																0000
PR1	0102	Period Register 1																FFFF
T1CON	0104	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>		—	TSYNC	TCS	—	0000
TMR2	0106	Timer2 Register																0000
TMR3HLD	0108	Timer3 Holding Register (for 32-bit timer operations only)																xxxx
TMR3	010A	Timer3 Register																0000
PR2	010C	Period Register 2																FFFF
PR3	010E	Period Register 3																FFFF
T2CON	0110	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>		T32	—	TCS	—	0000
T3CON	0112	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>		—	—	TCS	—	0000

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

TABLE 4-6: INPUT CAPTURE REGISTER MAP

SFR Name	SFR Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
IC1BUF	0140	Input 1 Capture Register																xxxx
IC1CON	0142	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>		ICOV	ICBNE	ICM<2:0>			0000
IC2BUF	0144	Input 2 Capture Register																xxxx
IC2CON	0146	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>		ICOV	ICBNE	ICM<2:0>			0000
IC7BUF	0158	Input 7 Capture Register																xxxx
IC7CON	015A	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>		ICOV	ICBNE	ICM<2:0>			0000
IC8BUF	015C	Input 8 Capture Register																xxxx
IC8CON	015E	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>		ICOV	ICBNE	ICM<2:0>			0000

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

TABLE 4-7: 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	Bit 1	Bit 0	All Resets
OC1RS	0180	Output Compare 1 Secondary Register																xxxx
OC1R	0182	Output Compare 1 Register																xxxx
OC1CON	0184	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	OCM<2:0>			0000
OC2RS	0186	Output Compare 2 Secondary Register																xxxx
OC2R	0188	Output Compare 2 Register																xxxx
OC2CON	018A	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	OCM<2:0>			0000

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

TABLE 4-8: I2C1 REGISTER MAP

SFR Name	SFR Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
I2C1RCV	0200	—	—	—	—	—	—	—	—	Receive Register								0000
I2C1TRN	0202	—	—	—	—	—	—	—	—	Transmit Register								00FF
I2C1BRG	0204	—	—	—	—	—	—	—	Baud Rate Generator Register									0000
I2C1CON	0206	I2CEN	—	I2CSIDL	SCLREL	IPMIEN	A10M	DISSLW	SMEN	GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	1000
I2C1STAT	0208	ACKSTAT	TRSTAT	—	—	—	BCL	GCSTAT	ADD10	IWCOL	I2COV	D_A	P	S	R_W	RBF	TBF	0000
I2C1ADD	020A	—	—	—	—	—	—	Address Register										0000
I2C1MSK	020C	—	—	—	—	—	—	Address Mask Register										0000

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

TABLE 4-9: UART1 REGISTER MAP

SFR Name	SFR Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
U1MODE	0220	UARTEN	—	USIDL	IREN	RTSMD	—	UEN1	UEN0	WAKE	LPBACK	ABAUD	URXINV	BRGH	PDSEL<1:0>		STSEL	0000
U1STA	0222	UTXISEL1	UTXINV	UTXISEL0	—	UTXBRK	UTXEN	UTXBF	TRMT	URXISEL<1:0>		ADDEN	RIDLE	PERR	FERR	OERR	URXDA	0110
U1TXREG	0224	—	—	—	—	—	—	—	UART Transmit Register									xxxx
U1RXREG	0226	—	—	—	—	—	—	—	UART Receive Register									0000
U1BRG	0228	Baud Rate Generator Prescaler																0000

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

TABLE 4-10: SPI1 REGISTER MAP

SFR Name	SFR Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
SPI1STAT	0240	SPIEN	—	SPISIDL	—	—	—	—	—	—	SPIROV	—	—	—	—	SPITBF	SPIRBF	0000
SPI1CON1	0242	—	—	—	DISSCK	DISSDO	MODE16	SMP	CKE	SSEN	CKP	MSTEN	SPRE<2:0>			PPRE<1:0>		0000
SPI1CON2	0244	FRMEN	SPIFSD	FRMPOL	—	—	—	—	—	—	—	—	—	—	—	FRMDLY	—	0000
SPI1BUF	0248	SPI1 Transmit and Receive Buffer Register																0000

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

TABLE 4-11: PERIPHERAL PIN SELECT INPUT REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
RPINR0	0680	—	—	—	INT1R<4:0>					—	—	—	—	—	—	—	—	1F00
RPINR1	0682	—	—	—	—	—	—	—	—	—	—	—	INT2R<4:0>					001F
RPINR3	0686	—	—	—	T3CKR<4:0>					—	—	—	T2CKR<4:0>					1F1F
RPINR7	068E	—	—	—	IC2R<4:0>					—	—	—	IC1R<4:0>					1F1F
RPINR10	0694	—	—	—	IC8R<4:0>					—	—	—	IC7R<4:0>					1F1F
RPINR11	0696	—	—	—	—	—	—	—	—	—	—	—	OCFAR<4:0>					001F
RPINR18	06A4	—	—	—	U1CTSR<4:0>					—	—	—	U1RX<R4:0>					1F1F
RPINR20	06A8	—	—	—	SCK1R<4:0>					—	—	—	SDI1R<4:0>					1F1F
RPINR21	06AA	—	—	—	—	—	—	—	—	—	—	—	SS1R<4:0>					001F

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

TABLE 4-12: PERIPHERAL PIN SELECT OUTPUT REGISTER MAP FOR dsPIC33FJ32GP202

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
RPOR0	06C0	—	—	—	RP1R<4:0>					—	—	—	RP0R<4:0>					0000
RPOR1	06C2	—	—	—	RP3R<4:0>					—	—	—	RP2R<4:0>					0000
RPOR2	06C4	—	—	—	RP5R<4:0>					—	—	—	RP4R<4:0>					0000
RPOR3	06C6	—	—	—	RP7R<4:0>					—	—	—	RP6R<4:0>					0000
RPOR4	06C8	—	—	—	RP9R<4:0>					—	—	—	RP8R<4:0>					0000
RPOR5	06CA	—	—	—	RP11R<4:0>					—	—	—	RP10R<4:0>					0000
RPOR6	06CC	—	—	—	RP13R<4:0>					—	—	—	RP12R<4:0>					0000
RPOR7	06CE	—	—	—	RP15R<4:0>					—	—	—	RP14R<4:0>					0000

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

TABLE 4-13: PERIPHERAL PIN SELECT OUTPUT REGISTER MAP FOR dsPIC33FJ32GP204 AND dsPIC33FJ16GP304

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
RPOR0	06C0	—	—	—	RP1R<4:0>					—	—	—	RP0R<4:0>					0000
RPOR1	06C2	—	—	—	RP3R<4:0>					—	—	—	RP2R<4:0>					0000
RPOR2	06C4	—	—	—	RP5R<4:0>					—	—	—	RP4R<4:0>					0000
RPOR3	06C6	—	—	—	RP7R<4:0>					—	—	—	RP6R<4:0>					0000
RPOR4	06C8	—	—	—	RP9R<4:0>					—	—	—	RP8R<4:0>					0000
RPOR5	06CA	—	—	—	RP11R<4:0>					—	—	—	RP10R<4:0>					0000
RPOR6	06CC	—	—	—	RP13R<4:0>					—	—	—	RP12R<4:0>					0000
RPOR7	06CE	—	—	—	RP15R<4:0>					—	—	—	RP14R<4:0>					0000
RPOR8	06D0	—	—	—	RP17R<4:0>					—	—	—	RP16R<4:0>					0000
RPOR9	06D2	—	—	—	RP19R<4:0>					—	—	—	RP18R<4:0>					0000
RPOR10	06D4	—	—	—	RP21R<4:0>					—	—	—	RP20R<4:0>					0000
RPOR11	06D6	—	—	—	RP23R<4:0>					—	—	—	RP22R<4:0>					0000
RPOR12	06D8	—	—	—	RP25R<4:0>					—	—	—	RP24R<4:0>					0000

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

TABLE 4-14: ADC1 REGISTER MAP FOR dsPIC33FJ32GP204 AND dsPIC33FJ16GP304

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
ADC1BUF0	0300	ADC Data Buffer 0																xxxx
ADC1BUF1	0302	ADC Data Buffer 1																xxxx
ADC1BUF2	0304	ADC Data Buffer 2																xxxx
ADC1BUF3	0306	ADC Data Buffer 3																xxxx
ADC1BUF4	0308	ADC Data Buffer 4																xxxx
ADC1BUF5	030A	ADC Data Buffer 5																xxxx
ADC1BUF6	030C	ADC Data Buffer 6																xxxx
ADC1BUF7	030E	ADC Data Buffer 7																xxxx
ADC1BUF8	0310	ADC Data Buffer 8																xxxx
ADC1BUF9	0312	ADC Data Buffer 9																xxxx
ADC1BUFA	0314	ADC Data Buffer 10																xxxx
ADC1BUFB	0316	ADC Data Buffer 11																xxxx
ADC1BUFC	0318	ADC Data Buffer 12																xxxx
ADC1BUFD	031A	ADC Data Buffer 13																xxxx
ADC1BUFE	031C	ADC Data Buffer 14																xxxx
ADC1BUFE	031E	ADC Data Buffer 15																xxxx
AD1CON1	0320	ADON	—	ADSIDL	—	—	AD12B	FORM<1:0>		SSRC<2:0>			—	SIMSAM	ASAM	SAMP	DONE	0000
AD1CON2	0322	VCFG<2:0>			—	—	CSCNA	CHPS<1:0>		BUFS	—	SMPI<3:0>				BUFM	ALTS	0000
AD1CON3	0324	ADRC	—	—	SAMC<4:0>					ADCS<7:0>								0000
AD1CHS123	0326	—	—	—	—	—	CH123NB<1:0>		CH123SB	—	—	—	—	—	CH123NA<1:0>		CH123SA	0000
AD1CHS0	0328	CH0NB	—	—	CH0SB<4:0>					CH0NA	—	—	CH0SA<4:0>					0000
AD1PCFGL	032C	—	—	—	PCFG12	PCFG11	PCFG10	PCFG9	PCFG8	PCFG7	PCFG6	PCFG5	PCFG4	PCFG3	PCFG2	PCFG1	PCFG0	0000
AD1CSSL	0330	—	—	—	CSS12	CSS11	CSS10	CSS9	CSS8	CSS7	CSS6	CSS5	CSS4	CSS3	CSS2	CSS1	CSS0	0000

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

TABLE 4-15: ADC1 REGISTER MAP FOR dsPIC33FJ32GP202

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
ADC1BUF0	0300	ADC Data Buffer 0																xxxx
ADC1BUF1	0302	ADC Data Buffer 1																xxxx
ADC1BUF2	0304	ADC Data Buffer 2																xxxx
ADC1BUF3	0306	ADC Data Buffer 3																xxxx
ADC1BUF4	0308	ADC Data Buffer 4																xxxx
ADC1BUF5	030A	ADC Data Buffer 5																xxxx
ADC1BUF6	030C	ADC Data Buffer 6																xxxx
ADC1BUF7	030E	ADC Data Buffer 7																xxxx
ADC1BUF8	0310	ADC Data Buffer 8																xxxx
ADC1BUF9	0312	ADC Data Buffer 9																xxxx
ADC1BUFA	0314	ADC Data Buffer 10																xxxx
ADC1BUFB	0316	ADC Data Buffer 11																xxxx
ADC1BUFC	0318	ADC Data Buffer 12																xxxx
ADC1BUFD	031A	ADC Data Buffer 13																xxxx
ADC1BUFE	031C	ADC Data Buffer 14																xxxx
ADC1BUFF	031E	ADC Data Buffer 15																xxxx
AD1CON1	0320	ADON	—	ADSIDL	—	—	AD12B	FORM<1:0>		SSRC<2:0>			—	SIMSAM	ASAM	SAMP	DONE	0000
AD1CON2	0322	VCFG<2:0>			—	—	CSCNA	CHPS<1:0>		BUFS	—	SMPI<3:0>				BUFM	ALTS	0000
AD1CON3	0324	ADRC	—	—	SAMC<4:0>					ADCS<7:0>								0000
AD1CHS123	0326	—	—	—	—	—	CH123NB<1:0>		CH123SB	—	—	—	—	—	CH123NA<1:0>		CH123SA	0000
AD1CHS0	0328	CH0NB	—	—	CH0SB<4:0>					CH0NA	—	—	CH0SA<4:0>					0000
AD1PCFGL	032C	—	—	—	PCFG12	PCFG11	PCFG10	PCFG9	—	—	—	PCFG5	PCFG4	PCFG3	PCFG2	PCFG1	PCFG0	0000
AD1CSSL	0330	—	—	—	CSS12	CSS11	CSS10	CSS9	—	—	—	CSS5	CSS4	CSS3	CSS2	CSS1	CSS0	0000

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

TABLE 4-16: PORTA REGISTER MAP FOR dsPIC33FJ32GP202

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
TRISA	02C0	—	—	—	—	—	—	—	—	—	—	—	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	001F
PORTA	02C2	—	—	—	—	—	—	—	—	—	—	—	RA4	RA3	RA2	RA1	RA0	xxxx
LATA	02C4	—	—	—	—	—	—	—	—	—	—	—	LATA4	LATA3	LATA2	LATA1	LATA0	xxxx
ODCA	02C6	—	—	—	—	—	—	—	—	—	—	—	ODCA4	ODCA3	ODCA2	ODCA1	ODCA0	0000

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

TABLE 4-17: PORTA REGISTER MAP FOR dsPIC33FJ32GP204 AND dsPIC33FJ16GP304

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
TRISA	02C0	—	—	—	—	—	TRISA10	TRISA9	TRISA8	TRISA7	—	—	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	001F
PORTA	02C2	—	—	—	—	—	RA10	RA9	RA8	RA7	—	—	RA4	RA3	RA2	RA1	RA0	xxxx
LATA	02C4	—	—	—	—	—	LATA10	LATA9	LATA8	LATA7	—	—	LATA4	LATA3	LATA2	LATA1	LATA0	xxxx
ODCA	02C6	—	—	—	—	—	ODCA10	ODCA9	ODCA8	ODCA7	—	—	ODCA4	ODCA3	ODCA2	ODCA1	ODCA0	0000

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

TABLE 4-18: PORTB REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
TRISB	02C8	TRISB15	TRISB14	TRISB13	TRISB12	TRISB11	TRISB10	TRISB9	TRISB8	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	FFFF
PORTB	02CA	RB15	RB14	RB13	RB12	RB11	RB10	RB9	RB8	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx
LATB	02CC	LATB15	LATB14	LATB13	LATB12	LATB11	LATB10	LATB9	LATB8	LATB7	LATB6	LATB5	LATB4	LATB3	LATB2	LATB1	LATB0	xxxx
ODCB	02CE	ODCB15	ODCB14	ODCB13	ODCB12	ODCB11	ODCB10	ODCB9	ODCB8	ODCB7	ODCB6	ODCB5	ODCB4	ODCB3	ODCB2	ODCB1	ODCB0	0000

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

TABLE 4-19: PORTC REGISTER MAP FOR dsPIC33FJ32GP204 AND dsPIC33FJ16GP304

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
TRISC	02D0	—	—	—	—	—	—	TRISC9	TRISC8	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	03FF
PORTC	02D2	—	—	—	—	—	—	RC9	RC8	RC7	RC6	RC5	RC4	RC4	RC2	RC1	RC0	xxxx
LATC	02D4	—	—	—	—	—	—	LATC9	LATC8	LATC7	LATC6	LATC5	LATC4	LATC4	LATC2	LATC1	LATC0	xxxx
ODCC	02D6	—	—	—	—	—	—	ODCC9	ODCC8	ODCC7	ODCC6	ODCC5	ODCC4	ODCC4	ODCC2	ODCC1	ODCC0	0000

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

TABLE 4-20: SYSTEM CONTROL REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
RCON	0740	TRAPR	IOPUWR	—	—	—	—	CM	VREGS	EXTR	SWR	SWDTEN	WDTO	SLEEP	IDLE	BOR	POR	xxxx(1)
OSCCON	0742	—	COSC<2:0>			—	NOSC<2:0>			CLKLOCK	IOLOCK	LOCK	—	CF	—	LPOSCEN	OSWEN	0300(2)
CLKDIV	0744	ROI	DOZE<2:0>			DOZEN	FRCDIV<2:0>			PLLPOST<1:0>		—	PLLPRE<4:0>					3040
PLLFBD	0746	—	—	—	—	—	—	—	PLLDIV<8:0>									0030
OSCTUN	0748	—	—	—	—	—	—	—	—	—	—	TUN<5:0>						0000

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

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

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

TABLE 4-21: NVM REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
NVMCON	0760	WR	WREN	WRERR	—	—	—	—	—	—	ERASE	—	—	NVMOP<3:0>				0000(1)
NVMKEY	0766	—	—	—	—	—	—	—	—	NVMKEY<7:0>								0000

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

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

TABLE 4-22: PMD REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
PMD1	0770	—	—	T3MD	T2MD	T1MD	—	—	—	I2C1MD	—	U1MD	—	SPI1MD	—	—	AD1MD	0000
PMD2	0772	IC8MD	IC7MD	—	—	—	—	IC2MD	IC1MD	—	—	—	—	—	—	OC2MD	OC1MD	0000

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

4.2.6 SOFTWARE STACK

In addition to its use as a working register, the W15 register in the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 devices is also used as a software Stack Pointer. The Stack Pointer always points to the first available free word and grows from lower to higher addresses. It pre-decrements for stack pops and post-increments for stack pushes, as shown in Figure 4-4. 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.

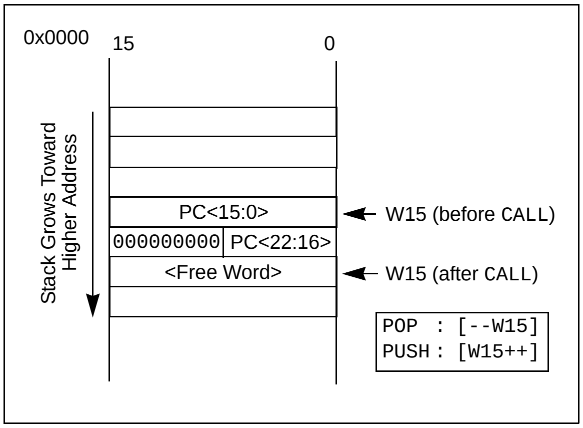
The Stack Pointer Limit register (SPLIM) associated with the Stack Pointer sets an upper address boundary for the stack. SPLIM is uninitialized at Reset. As is the case for the Stack Pointer, SPLIM<0> is forced to ‘`0`’ because all stack operations must be word-aligned.

When 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 0x1000 in RAM, initialize the SPLIM with the value 0x0FFE.

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 4-4: CALL STACK FRAME



4.2.7 DATA RAM PROTECTION FEATURE

The dsPIC33F product family supports Data RAM protection features that enable segments of RAM to be protected when used in conjunction with Boot and Secure Code Segment Security. BSRAM (Secure RAM segment for Boot Segment) is accessible only from the Boot Segment Flash code when enabled. SSRAM (Secure RAM segment for RAM) is accessible only from the Secure Segment Flash code when enabled. See Table 4-1 for an overview of the BSRAM and SSRAM SFRs.

4.3 Instruction Addressing Modes

The addressing modes shown in Table 4-23 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:

`Operand 3 = Operand 1 <function> 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 4-23: 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`, `MOVSAC` 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 `NOP`, do not have any operands.

4.4 Modulo Addressing

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

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

In general, any particular circular buffer can 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

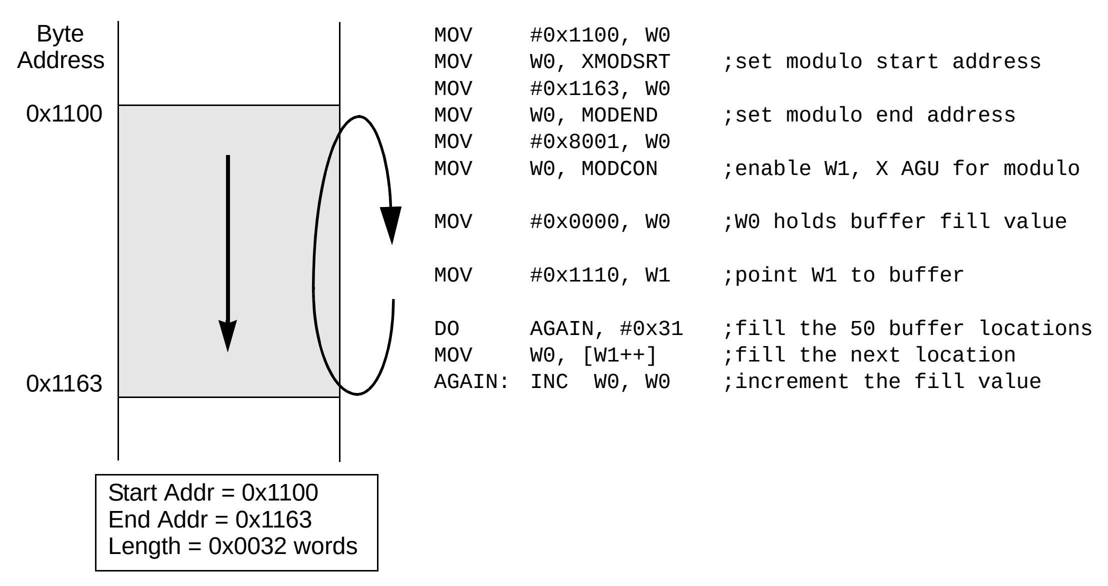
The Modulo and Bit-Reversed Addressing Control register, MODCON<15:0>, contains enable flags as well as a W register field to specify the W Address registers. The XWM and YWM fields select the registers that will operate with Modulo Addressing:

- 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 4-5: MODULO ADDRESSING OPERATION EXAMPLE



```
MOV     #0x1100, W0
MOV     W0, XMODSRT     ;set modulo start address
MOV     #0x1163, W0
MOV     W0, MODEND      ;set modulo end address
MOV     #0x8001, W0
MOV     W0, MODCON      ;enable W1, X AGU for modulo

MOV     #0x0000, W0     ;W0 holds buffer fill value

MOV     #0x1110, W1     ;point W1 to buffer

DO      AGAIN, #0x31    ;fill the 50 buffer locations
MOV     W0, [W1++]      ;fill the next location
AGAIN:  INC  W0, W0     ;increment the fill value
```

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:

- The upper boundary addresses for incrementing buffers
- The lower boundary addresses for decrementing buffers

It is important to realize that the address boundaries also check for addresses less than or greater than these addresses. 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 WAGU Modulo Addressing will be disabled. However, Modulo Addressing will continue to function in the X RAGU.

If Bit-Reversed Addressing has already been enabled by setting the BREN bit (XBREV<15>), 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 4-6: BIT-REVERSED ADDRESS EXAMPLE

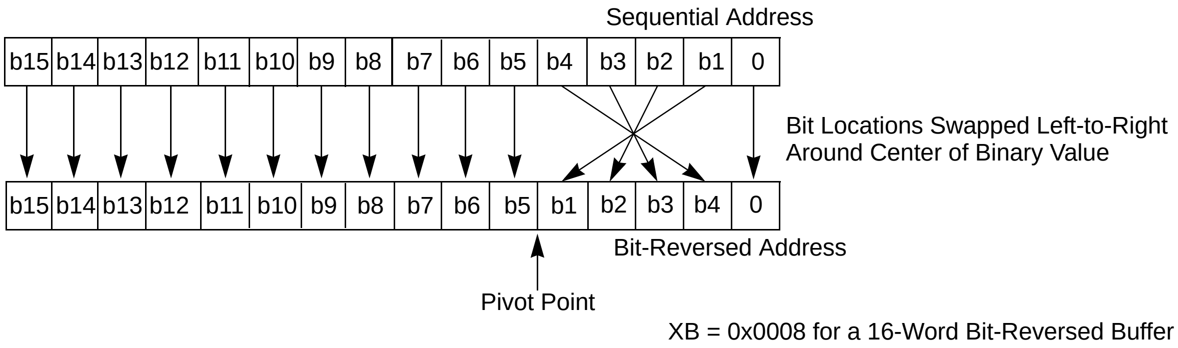


TABLE 4-24: BIT-REVERSED ADDRESS SEQUENCE (16-ENTRY)

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 dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 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 dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 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.

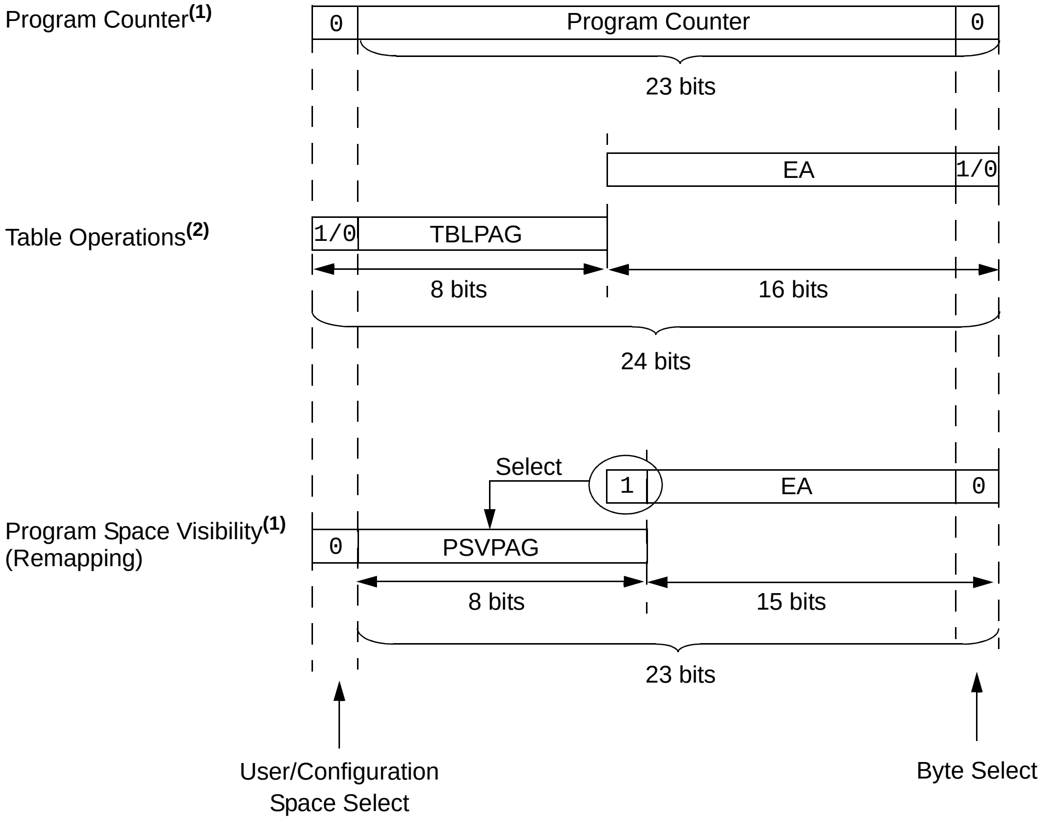
Table 4-25 and Figure 4-7 show how the program EA is created for table operations and remapping accesses from the data EA. Here, P<23:0> refers to a program space word, and D<15:0> refers to a data space word.

TABLE 4-25: 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>(1)	
		`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 4-7: DATA ACCESS FROM PROGRAM SPACE ADDRESS GENERATION



Note 1: The Least Significant bit (LSb) of program space addresses is always fixed as ‘`0`’ to maintain word alignment of data in the program and data spaces.

2: Table operations are not required to be word-aligned. Table read operations are permitted in the configuration memory space.

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>).

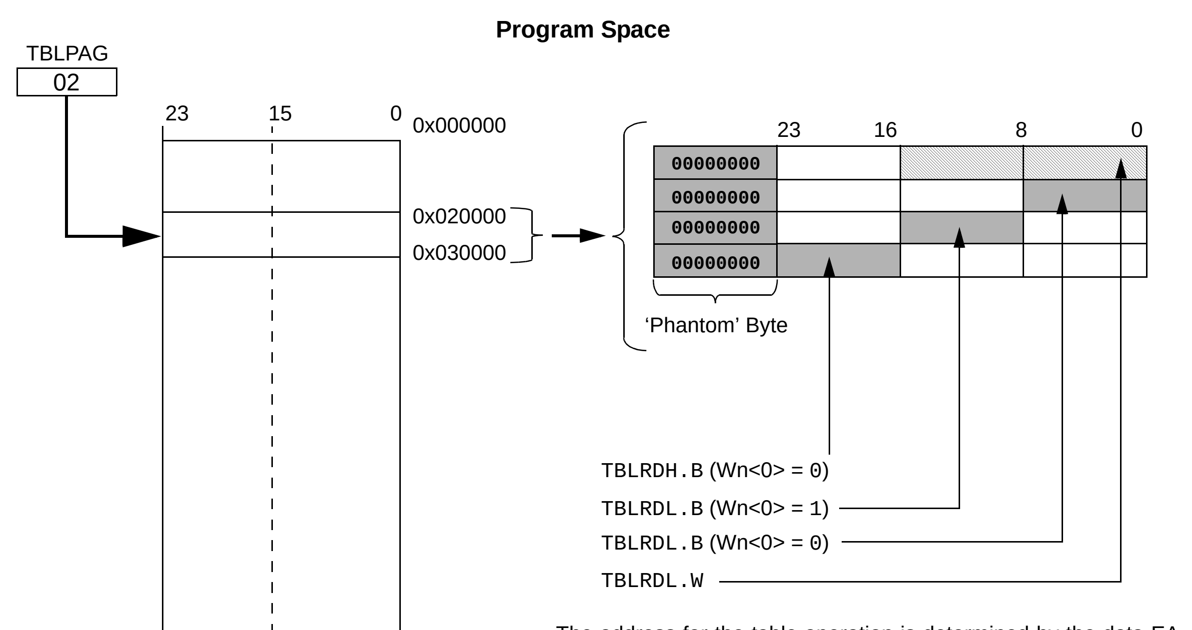
  In Byte mode, either the upper or lower byte of the lower program word is mapped to the lower byte of a data address. The upper byte is selected when Byte Select is ‘`1`’; the lower byte is selected when it is ‘`0`’.
- `TBLRDH` (Table Read High): In Word mode, this instruction maps the entire upper word 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, as in the `TBLRDL` instruction. Note that the data will always be ‘`0`’ when the upper ‘phantom’ byte is selected (Byte Select = `1`).

In a similar fashion, two table instructions, `TBLWTH` and `TBLWTL`, are used to write individual bytes or words to a program space address. The details of their operation are explained in **Section 5.0 “Flash Program Memory”**.

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

FIGURE 4-8: ACCESSING PROGRAM MEMORY WITH TABLE INSTRUCTIONS



The address for the table operation is determined by the data EA within the page defined by the TBLPAG register.
Only read operations are shown; write operations are also valid in the user memory area.

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 `TBLRDH`).

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 4-9), 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.

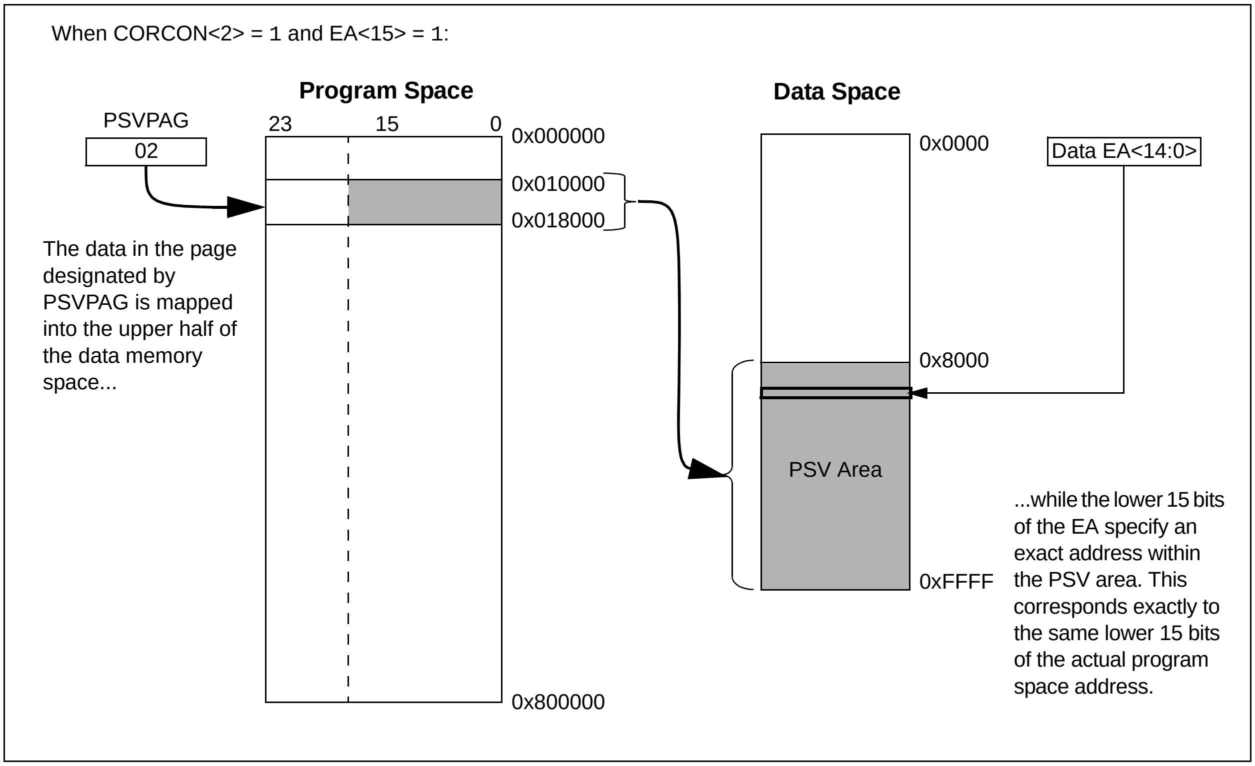
For operations that use PSV and are executed outside a `REPEAT` loop, the `MOV` and `MOV.D` instructions require one instruction cycle in addition to the specified execution time. All other instructions require two instruction cycles in addition to the specified execution time.

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 4-9: PROGRAM SPACE VISIBILITY OPERATION



The data in the page designated by PSVPAG is mapped into the upper half of the data memory space...

...while the lower 15 bits of the EA specify an exact address within the PSV area. This corresponds exactly to the same lower 15 bits of the actual program space address.

NOTES:

5.0 FLASH PROGRAM MEMORY

> **Note 1:** This data sheet summarizes the features of the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 family 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) of the *"dsPIC33F/PIC24H Family Reference Manual"*, which is available from the Microchip website (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 dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 devices contain internal Flash program memory for storing and executing application code. The memory is readable, writable and erasable during normal operation over the entire VDD range.

Flash memory can be programmed in two ways:

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

ICSP allows a dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 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: PGECx/PGEDx), and three other lines for power (VDD), ground (VSS) and Master Clear ($\overline{\text{MCLR}}$). 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.

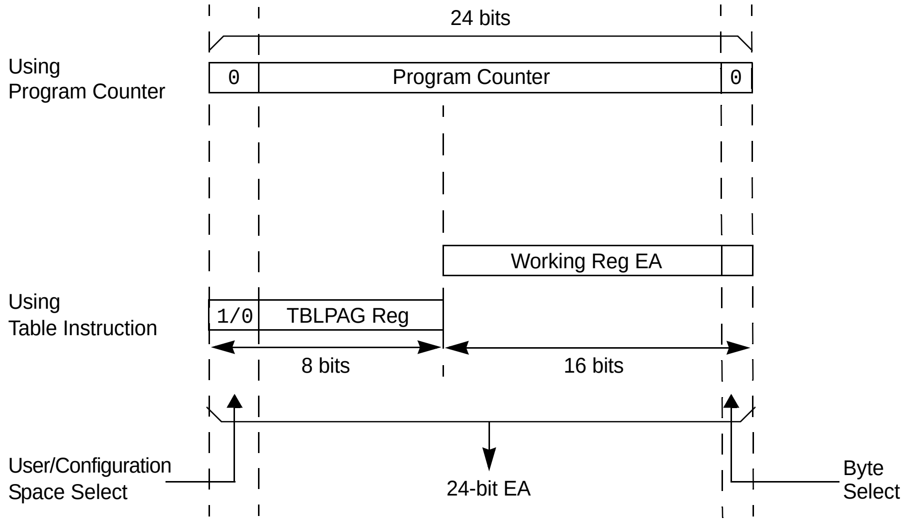
5.1 Table Instructions and Flash Programming

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 dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 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. 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 22-18) and the value of the FRC Oscillator Tuning register (see Register 8-4). 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 22-12).

EQUATION 5-1: PROGRAMMING TIME

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

For example, if the device is operating at +125° C, the FRC accuracy will be ±5%. If the TUN<5:0> bits (see Register 8-4) are set to `'b111111`, the minimum row write time is equal to Equation 5-2.

EQUATION 5-2: MINIMUM ROW WRITE TIME

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

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

EQUATION 5-3: MAXIMUM ROW WRITE TIME

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

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

5.4 Control Registers

The two SFRs that are used to read and write the program Flash memory are:

- **NVMCON: Flash Memory Control Register**
- **NVMKEY: Nonvolatile Memory Key Register**

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 (Register 5-2) 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[1]	R/W-0[1]	R/W-0[1]	U-0	U-0	U-0	U-0	U-0
WR	WREN	WRERR	—	—	—	—	—
bit 15							bit 8

U-0	R/W-0[1]	U-0	U-0	R/W-0[1]	R/W-0[1]	R/W-0[1]	R/W-0[1]
—	ERASE	—	—	NVMOP<3:0>[2]			
bit 7							bit 0

Legend:	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[2]

If ERASE = `1`:
`1111` = Memory bulk erase operation
`1101` = Erase General Segment
`1100` = Erase Secure 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
`1100` = 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> bits are unimplemented.

REGISTER 5-2: NVMKEY: NONVOLATILE 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:	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 15-8	**Unimplemented:** Read as ‘0’
bit 7-0	**NVMKEY<7:0>:** Key Register (Write Only) bits

5.4.1 PROGRAMMING ALGORITHM FOR FLASH PROGRAM MEMORY

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

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 bit (NVMCON<6>) and the WREN bit (NVMCON<14>).
    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 `NOP`s, 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                                 ; Insert two NOPs after the erase
       NOP                                 ; 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 dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 family 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) of the *"dsPIC33F/PIC24H Family Reference Manual"*, which is available from the Microchip website (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: RESET Instruction
- WDTO: Watchdog Timer Reset
- CM: Configuration Mismatch 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 manual for register Reset states.

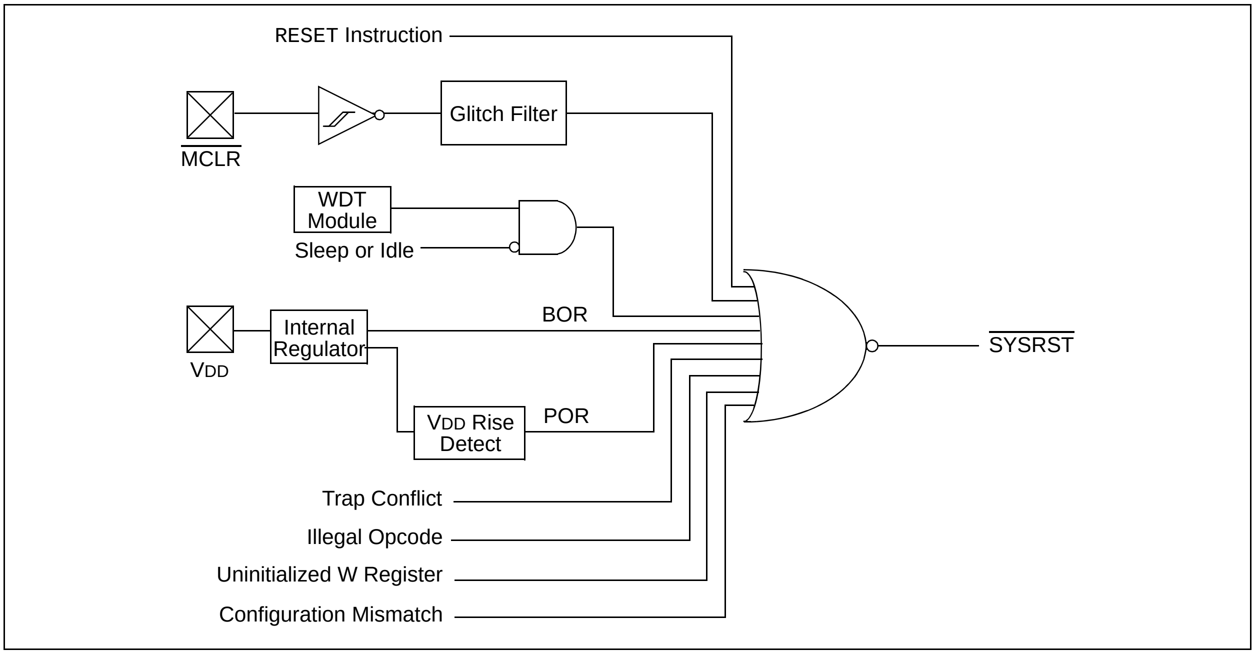
All types of device Reset sets a corresponding status bit in the RCON register to indicate the type of Reset (see Register 6-1).

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 MODULE BLOCK DIAGRAM



REGISTER 6-1: RCON: RESET CONTROL REGISTER(1)

R/W-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
TRAPR	IOPUWR	—	—	—	—	CM	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(2)	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 13-10 **Unimplemented:** Read as ‘0’

bit 9 **CM:** Configuration Mismatch Flag bit
1 = A configuration mismatch Reset has occurred
0 = A configuration mismatch Reset has NOT occurred

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 ($\overline{\text{MCLR}}$) Pin bit
1 = A Master Clear (pin) Reset has occurred
0 = A Master Clear (pin) Reset has not occurred

bit 6 **SWR:** Software Reset (Instruction) Flag bit
1 = A RESET instruction has been executed
0 = A RESET instruction has not been executed

bit 5 **SWDTEN:** Software Enable/Disable of WDT bit(2)
1 = WDT is enabled
0 = WDT is disabled

bit 4 **WDTO:** Watchdog Timer Time-out Flag bit
1 = WDT time-out has occurred
0 = WDT time-out has not occurred

bit 3 **SLEEP:** Wake-up from Sleep Flag bit
1 = Device has been in Sleep mode
0 = Device has not been in Sleep mode

bit 2 **IDLE:** Wake-up from Idle Flag bit
1 = Device was in Idle mode
0 = Device was not in Idle mode

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

2: If the FWDTEN Configuration bit is ‘1’ (unprogrammed), the WDT is always enabled, regardless of the SWDTEN bit setting.

REGISTER 6-1: RCON: RESET CONTROL REGISTER[1] (CONTINUED)

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 FWDTEN Configuration bit is ‘`1`’ (unprogrammed), the WDT is always enabled, regardless of the SWDTEN bit setting.

6.1 System Reset

The dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 family 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 device configuration register selects the device clock source.

A warm Reset is the result of all 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 bits (COSC<2:0>) in the Oscillator Control register (OSCCON<14:12>).

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 shown in Figure 6-2.

TABLE 6-1: OSCILLATOR DELAY

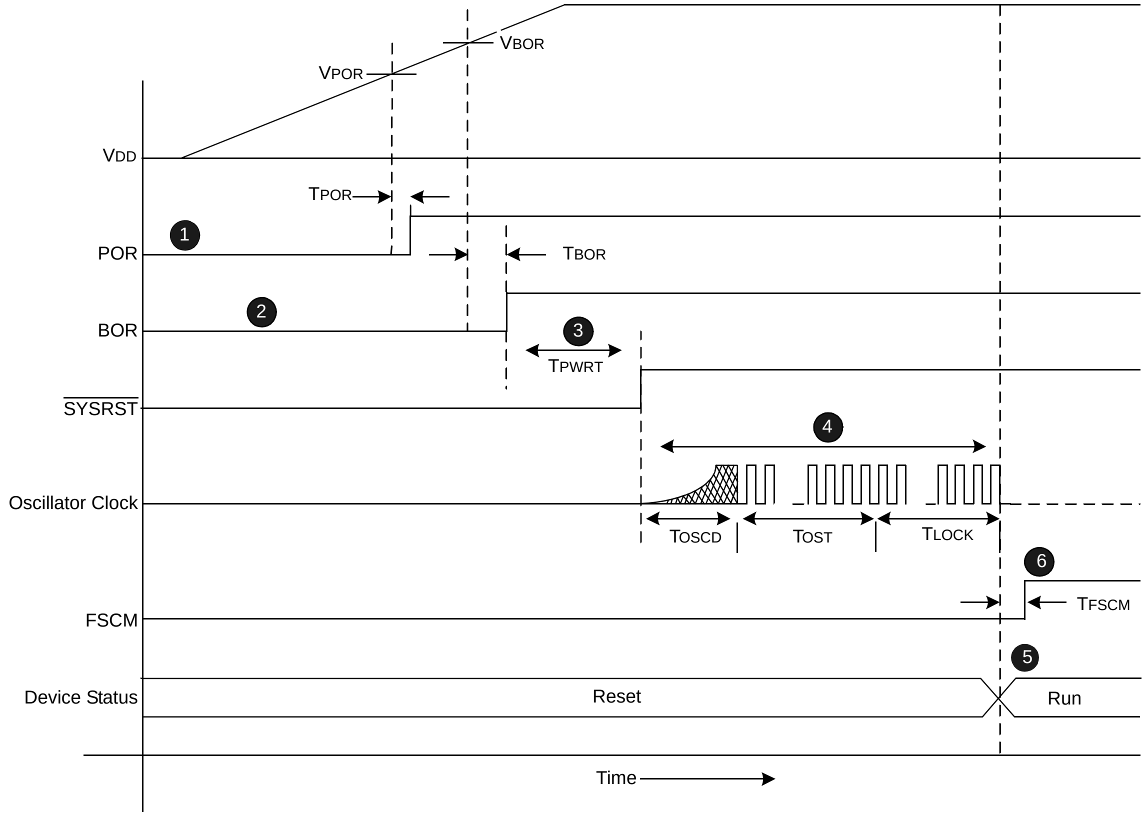
Oscillator Mode	Oscillator Start-up Delay	Oscillator Start-up Timer	PLL Lock Time	Total Delay
FRC, FRCDIV16, FRCDIVN	TOSCD	—	—	TOSCD
FRCPLL	TOSCD	—	TLOCK	TOSCD + TLOCK
XT	TOSCD	TOST	—	TOSCD + TOST
HS	TOSCD	TOST	—	TOSCD + TOST
EC	—	—	—	—
XTPLL	TOSCD	TOST	TLOCK	TOSCD + TOST + TLOCK
HSPLL	TOSCD	TOST	TLOCK	TOSCD + TOST + TLOCK
ECPLL	—	—	TLOCK	TLOCK
SOSC	TOSCD	TOST	—	TOSCD + TOST
LPRC	TOSCD	—	—	TOSCD

Note 1: TOSCD = 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: TOST = Oscillator Start-up Timer Delay (1024 oscillator clock period). For example, TOST = 102.4 μs for a 10 MHz crystal and TOST = 32 ms for a 32 kHz crystal.

3: TLOCK = PLL lock time (1.5 ms nominal), if PLL is enabled.

FIGURE 6-2: SYSTEM RESET TIMING



Note 1: **POR:** A POR circuit holds the device in Reset when the power supply is turned on. The POR circuit is active until VDD crosses the VPOR threshold and the delay TPOR has elapsed.

2: **BOR:** The on-chip voltage regulator has a BOR circuit that keeps the device in Reset until VDD crosses the VBOR threshold and the delay TBOR has elapsed. The delay TBOR ensures the voltage regulator output becomes stable.

3: **PWRT Timer:** The programmable power-up timer continues to hold the processor in Reset for a specific period of time (TPWRT) after a BOR. The delay TPWRT ensures that the system power supplies have stabilized at the appropriate level for full-speed operation. After the delay TPWRT has elapsed, the SYSRST becomes inactive, which in turn enables the selected oscillator to start generating clock cycles.

4: **Oscillator Delay:** The total delay for the clock to be ready for various clock source selections are given in Table 6-1. Refer to **Section 8.0 "Oscillator Configuration"** for more information.

5: When the oscillator clock is ready, the processor begins execution from location 0x000000. The user application programs a `GOTO` instruction at the reset address, which redirects program execution to the appropriate start-up routine.

6: The Fail-Safe Clock Monitor (FSCM), if enabled, begins to monitor the system clock when the system clock is ready and the delay TFSCM elapsed.

TABLE 6-2: OSCILLATOR PARAMETERS

Symbol	Parameter	Value
VPOR	POR threshold	1.8V nominal
TPOR	POR extension time	30 μs maximum
VBOR	BOR threshold	2.5V nominal
TBOR	BOR extension time	100 μs maximum
TPWRT	Programmable power-up time delay	0-128 ms nominal
TFSCM	Fail-Safe Clock Monitor Delay	900 μs maximum

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 22.0 “Electrical Characteristics”** for details.

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

6.2.1 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 (VDD < VBOR) for proper device operation. The BOR circuit keeps the device in Reset until VDD crosses VBOR threshold and the delay TBOR has elapsed. The delay TBOR ensures the voltage regulator output becomes stable.

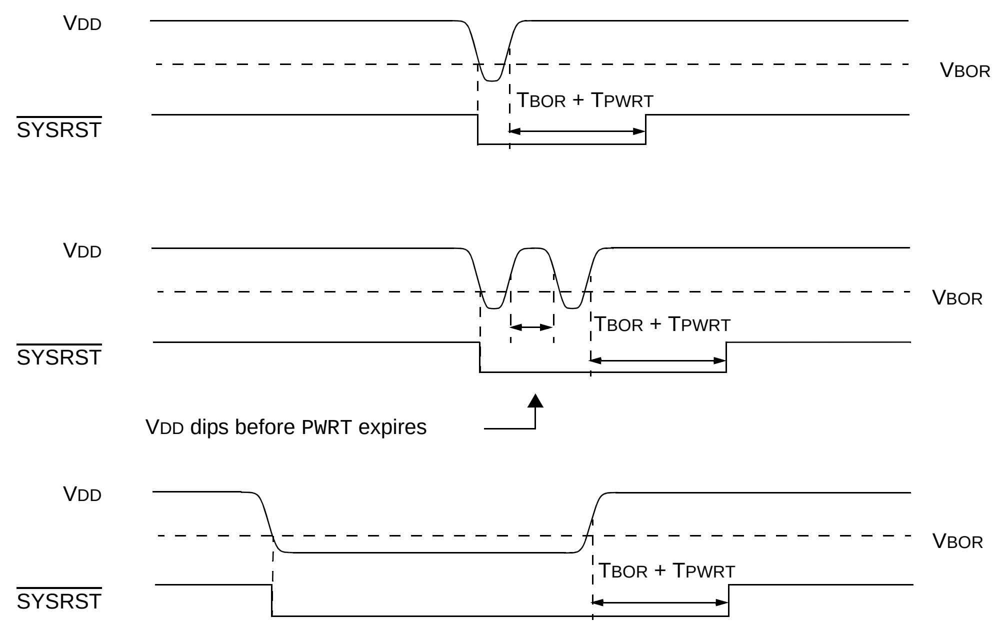
The BOR status bit in the Reset Control register (RCON<1>) is set to indicate the Brown-out Reset.

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 bits (FPWRT<2:0>) in the POR Configuration register (FPOR<2:0>), which provide eight settings (from 0 ms to 128 ms). Refer to **Section 19.0 “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 VBOR trip point

FIGURE 6-3: BROWN-OUT SITUATIONS



6.3 External Reset (EXTR)

The external Reset is generated by driving the MCLR pin low. The 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 22.0 "Electrical Characteristics"** for minimum pulse-width specifications. The External Reset (MCLR) Pin (EXTR) bit in the Reset Control (RCON) register is set to indicate the MCLR Reset.

6.3.1 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 MCLR pin to Reset the device when the rest of system is Reset.

6.3.2 INTERNAL SUPERVISORY CIRCUIT

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

6.4 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 (Instruction) Flag bit (SWR) in the Reset Control register (RCON<6>) is set to indicate the software Reset.

6.5 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 Flag (WDTO) bit in the Reset Control register (RCON<4>) is set to indicate the Watchdog Reset. Refer to **Section 19.4 "Watchdog Timer (WDT)"** for more information on Watchdog Reset.

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

The Trap Reset Flag bit (TRAPR) in the Reset Control register (RCON<15>) is set to indicate the Trap Conflict Reset. Refer to **Section 7.0 "Interrupt Controller"** for more information on trap conflict Resets.

6.7 Configuration Mismatch Reset

To maintain the integrity of the peripheral pin select control registers, they are constantly monitored with shadow registers in hardware. If an unexpected change in any of the registers occur (such as cell disturbances caused by ESD or other external events), a configuration mismatch Reset occurs.

The Configuration Mismatch Flag bit (CM) in the Reset Control register (RCON<9>) is set to indicate the configuration mismatch Reset. Refer to **Section 10.0 “I/O Ports”** for more information on the configuration mismatch Reset.

Note: The configuration mismatch feature and associated reset flag is not available on all devices.

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 Flag bit (IOPUWR) in the Reset Control register (RCON<14>) 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 attempts 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 19.6 “Code Protection and CodeGuard™ Security”** for more information on Security Reset.

6.9 Using the RCON Status Bits

The user application can read the Reset Control register (RCON) 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 6-3 provides a summary of the reset flag bit operation.

TABLE 6-3: RESET FLAG BIT OPERATION

Flag Bit	Set by:	Cleared by:
TRAPR (RCON<15>)	Trap conflict event	POR,BOR
IOPWR (RCON<14>)	Illegal opcode or uninitialized W register access or Security Reset	POR,BOR
CM (RCON<9>)	Configuration Mismatch	POR,BOR
EXTR (RCON<7>)	$\overline{\text{MCLR}}$ Reset	POR
SWR (RCON<6>)	`RESET` instruction	POR,BOR
WDTO (RCON<4>)	WDT time-out	`PWRSAV` instruction, `CLRWDT` instruction, POR,BOR
SLEEP (RCON<3>)	`PWRSAV #SLEEP` instruction	POR,BOR
IDLE (RCON<2>)	`PWRSAV #IDLE` instruction	POR,BOR
BOR (RCON<1>)	POR, BOR	—
POR (RCON<0>)	POR	—

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

7.0 INTERRUPT CONTROLLER

> **Note 1:** This data sheet summarizes the features of the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 6. "Interrupts"** (DS70184) of the *"dsPIC33F/PIC24H Family Reference Manual"*, which is available from the Microchip website (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 dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 interrupt controller reduces the numerous peripheral interrupt request signals to a single interrupt request signal to the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 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 is shown in Figure 7-1. The IVT resides in program memory, starting at location 000004h. The IVT contains 126 vectors consisting of 8 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.

dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 devices implement up to 21 unique interrupts and four nonmaskable traps. These are summarized in Table 7-1 and Table 7-2.

7.1.1 ALTERNATE INTERRUPT VECTOR TABLE

The Alternate Interrupt Vector Table (AIVT) is located after the IVT, as shown in Figure 7-1. Access to the AIVT is provided by the ALTIVT control bit (INTCON2<15>). If the ALTIVT bit is set, all interrupt and exception processes use the alternate vectors instead of the default vectors. The alternate vectors are organized in the same manner as the default vectors.

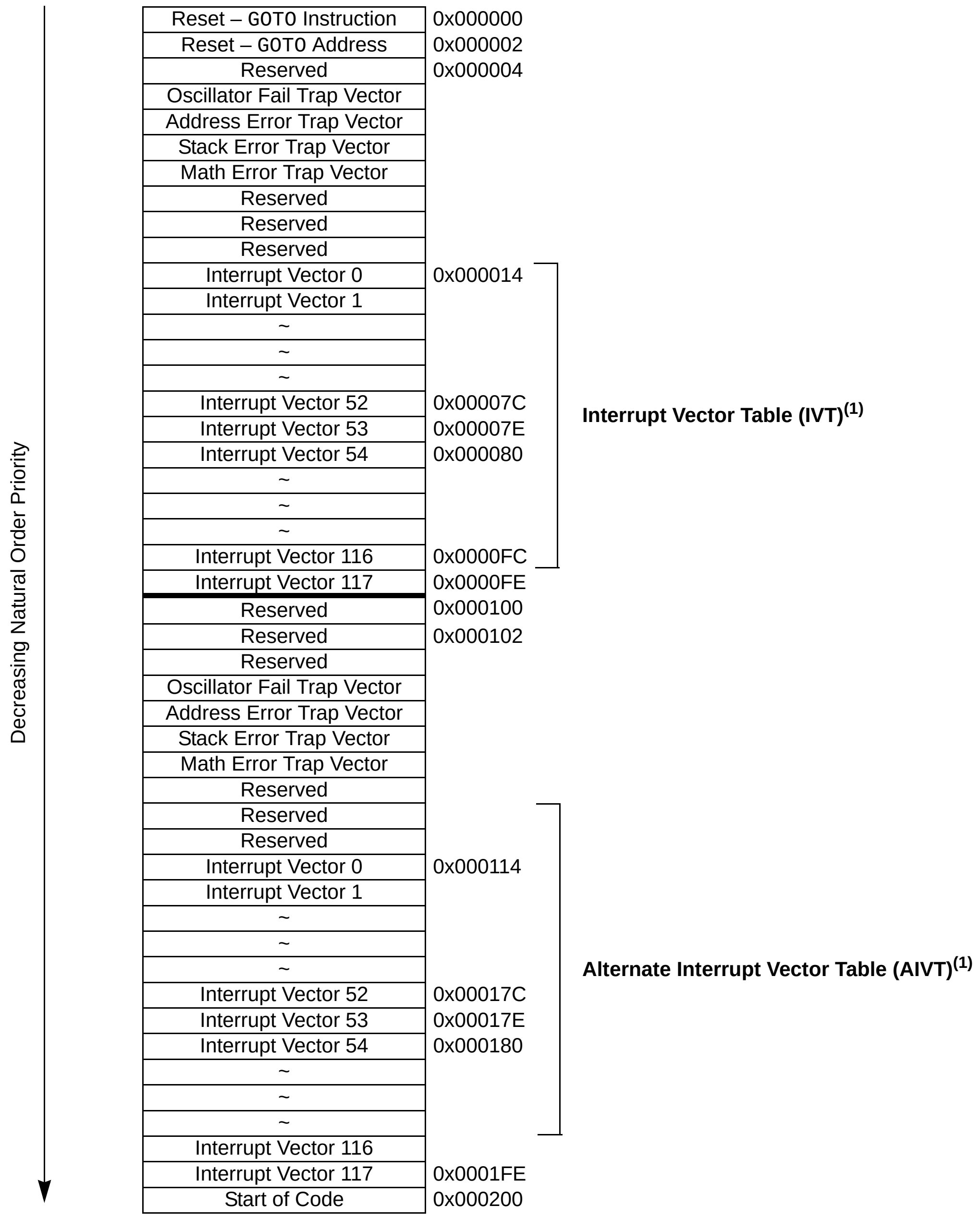
The AIVT supports debugging by providing a means to switch between an application and a support environment without requiring the interrupt vectors to be reprogrammed. This feature also enables switching between applications for evaluation of different software algorithms at run time. If the AIVT is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

7.2 Reset Sequence

A device Reset is not a true exception because the interrupt controller is not involved in the Reset process. The dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 device clears its registers in response to a Reset, which forces the PC to zero. The digital signal controller then begins program execution at location 0x000000. The user application can use a `GOTO` instruction at the Reset address which redirects program execution to the appropriate start-up routine.

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

FIGURE 7-1: dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 INTERRUPT VECTOR TABLE



Reset – `GOTO` Instruction	0x000000
Reset – `GOTO` Address	0x000002
Reserved	0x000004
Oscillator Fail Trap Vector	
Address Error Trap Vector	
Stack Error Trap Vector	
Math Error Trap Vector	
Reserved	
Reserved	
Reserved	
Interrupt Vector 0	0x000014
Interrupt Vector 1	
~	
~	
~	
Interrupt Vector 52	0x00007C
Interrupt Vector 53	0x00007E
Interrupt Vector 54	0x000080
~	
~	
~	
Interrupt Vector 116	0x0000FC
Interrupt Vector 117	0x0000FE
Reserved	0x000100
Reserved	0x000102
Reserved	
Oscillator Fail Trap Vector	
Address Error Trap Vector	
Stack Error Trap Vector	
Math Error Trap Vector	
Reserved	
Reserved	
Reserved	
Interrupt Vector 0	0x000114
Interrupt Vector 1	
~	
~	
~	
Interrupt Vector 52	0x00017C
Interrupt Vector 53	0x00017E
Interrupt Vector 54	0x000180
~	
~	
~	
Interrupt Vector 116	
Interrupt Vector 117	0x0001FE
Start of Code	0x000200

Note 1: See Table 7-1 for the list of implemented interrupt vectors.

TABLE 7-1: INTERRUPT VECTORS

Vector Number	Interrupt Request (IRQ) Number	IVT Address	AIVT Address	Interrupt Source
8	0	0x000014	0x000114	INT0 – External Interrupt 0
9	1	0x000016	0x000116	IC1 – Input Capture 1
10	2	0x000018	0x000118	OC1 – Output Compare 1
11	3	0x00001A	0x00011A	T1 – Timer1
12	4	0x00001C	0x00011C	Reserved
13	5	0x00001E	0x00011E	IC2 – Input Capture 2
14	6	0x000020	0x000120	OC2 – Output Compare 2
15	7	0x000022	0x000122	T2 – Timer2
16	8	0x000024	0x000124	T3 – Timer3
17	9	0x000026	0x000126	SPI1E – SPI1 Error
18	10	0x000028	0x000128	SPI1 – SPI1 Transfer Done
19	11	0x00002A	0x00012A	U1RX – UART1 Receiver
20	12	0x00002C	0x00012C	U1TX – UART1 Transmitter
21	13	0x00002E	0x00012E	ADC1 – ADC1
22	14	0x000030	0x000130	Reserved
23	15	0x000032	0x000132	Reserved
24	16	0x000034	0x000134	SI2C1 – I2C1 Slave Events
25	17	0x000036	0x000136	MI2C1 – I2C1 Master Events
26	18	0x000038	0x000138	Reserved
27	19	0x00003A	0x00013A	Change Notification Interrupt
28	20	0x00003C	0x00013C	INT1 – External Interrupt 1
29	21	0x00003E	0x00013E	Reserved
30	22	0x000040	0x000140	IC7 – Input Capture 7
31	23	0x000042	0x000142	IC8 – Input Capture 8
32	24	0x000044	0x000144	Reserved
33	25	0x000046	0x000146	Reserved
34	26	0x000048	0x000148	Reserved
35	27	0x00004A	0x00014A	Reserved
36	28	0x00004C	0x00014C	Reserved
37	29	0x00004E	0x00014E	INT2 – External Interrupt 2
38	30	0x000050	0x000150	Reserved
39	31	0x000052	0x000152	Reserved
40	32	0x000054	0x000154	Reserved
41	33	0x000056	0x000156	Reserved
42	34	0x000058	0x000158	Reserved
43	35	0x00005A	0x00015A	Reserved
44	36	0x00005C	0x00015C	Reserved
45	37	0x00005E	0x00015E	Reserved
46	38	0x000060	0x000160	Reserved
47	39	0x000062	0x000162	Reserved
48	40	0x000064	0x000164	Reserved
49	41	0x000066	0x000166	Reserved
50	42	0x000068	0x000168	Reserved
51	43	0x00006A	0x00016A	Reserved
52	44	0x00006C	0x00016C	Reserved
53	45	0x00006E	0x00016E	Reserved

TABLE 7-1: INTERRUPT VECTORS (CONTINUED)

Vector Number	Interrupt Request (IRQ) Number	IVT Address	AIVT Address	Interrupt Source
54	46	0x000070	0x000170	Reserved
55	47	0x000072	0x000172	Reserved
56	48	0x000074	0x000174	Reserved
57	49	0x000076	0x000176	Reserved
58	50	0x000078	0x000178	Reserved
59	51	0x00007A	0x00017A	Reserved
60	52	0x00007C	0x00017C	Reserved
61	53	0x00007E	0x00017E	Reserved
62	54	0x000080	0x000180	Reserved
63	55	0x000082	0x000182	Reserved
64	56	0x000084	0x000184	Reserved
65	57	0x000086	0x000186	Reserved
66	58	0x000088	0x000188	Reserved
67	59	0x00008A	0x00018A	Reserved
68	60	0x00008C	0x00018C	Reserved
69	61	0x00008E	0x00018E	Reserved
70	62	0x000090	0x000190	Reserved
71	63	0x000092	0x000192	Reserved
72	64	0x000094	0x000194	Reserved
73	65	0x000096	0x000196	U1E – UART1 Error
74	66	0x000098	0x000198	Reserved
75	67	0x00009A	0x00019A	Reserved
76	68	0x00009C	0x00019C	Reserved
77	69	0x00009E	0x00019E	Reserved
78	70	0x0000A0	0x0001A0	Reserved
79	71	0x0000A2	0x0001A2	Reserved
80-125	72-117	0x0000A4-0x0000FE	0x0001A4-0x0001FE	Reserved

TABLE 7-2: TRAP VECTORS

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

7.3 Interrupt Control and Status Registers

dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 devices implement a total of 17 registers for the interrupt controller:

- Interrupt Control Register 1 (INTCON1)
- Interrupt Control Register 2 (INTCON2)
- Interrupt Flag Status Registers (IFSx)
- Interrupt Enable Control Registers (IECx)
- Interrupt Priority Control Registers (IPCx)
- Interrupt Control and Status Register (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 IFS registers maintain all of the interrupt request flags. Each source of interrupt has a status bit, which is set by the respective peripherals or external signal and is cleared via software.

7.3.3 IECx

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

7.3.4 IPCx

The IPC registers are used to set the interrupt priority level for each source of interrupt. Each user interrupt source can be assigned to one of eight priority levels.

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 bits (ILR<3:0>) in the INTTREG register. The new interrupt priority level is the priority of the pending interrupt.

The interrupt sources are assigned to the IFSx, IECx and IPCx registers in the same sequence that they are listed in Table 7-1. For example, the INT0 (External Interrupt 0) is shown as having vector number 8 and a natural order priority of 0. Thus, the INT0IF bit is found in IFS0<0>, the INT0IE bit in IEC0<0>, and the INT0IP bits in the first position of IPC0 (IPC0<2:0>).

7.3.6 STATUS 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>, also indicates the current CPU priority level. IPL3 is a read-only bit, so that trap events cannot be masked by the user software.

All Interrupt bits and registers are described in Register 7-1 through Register 7-19.

REGISTER 7-1: SR: CPU STATUS REGISTER(1)

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

Legend:		
C = Clear only bit	R = Readable bit	U = Unimplemented bit, read as ‘0’
S = Set only 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(2)

`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 bit (INTCON1<15>) = `1`.

REGISTER 7-2: CORCON: CORE CONTROL REGISTER(1)

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(2)**	PSV	RND	IF
bit 7							bit 0

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

bit 3 **IPL3:** CPU Interrupt Priority Level Status bit 3(2)

`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.

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	U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0
SFTACERR	DIV0ERR	—	MATHERR	ADDRERR	STKERR	OSCFAIL	—
bit 7							bit 0

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

bit 15	**NSTDIS:** Interrupt Nesting Disable bit 1 = Interrupt nesting is disabled 0 = Interrupt nesting is enabled
bit 14	**OVAERR:** Accumulator A Overflow Trap Flag bit 1 = Trap was caused by overflow of Accumulator A 0 = Trap was not caused by overflow of Accumulator A
bit 13	**OVBERR:** Accumulator B Overflow Trap Flag bit 1 = Trap was caused by overflow of Accumulator B 0 = Trap was not caused by overflow of Accumulator B
bit 12	**COVAERR:** Accumulator A Catastrophic Overflow Trap Flag bit 1 = Trap was caused by catastrophic overflow of Accumulator A 0 = Trap was not caused by catastrophic overflow of Accumulator A
bit 11	**COVBERR:** Accumulator B Catastrophic Overflow Trap Flag bit 1 = Trap was caused by catastrophic overflow of Accumulator B 0 = Trap was not caused by catastrophic overflow of Accumulator B
bit 10	**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	**Unimplemented:** Read as ‘0’
bit 4	**MATHERR:** Arithmetic Error Status bit 1 = Math error trap has occurred 0 = Math error trap has not occurred
bit 3	**ADDRERR:** Address Error Trap Status bit 1 = Address error trap has occurred 0 = Address error trap has not occurred

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

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	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	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 13-3	**Unimplemented:** Read as ‘`0`’
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

REGISTER 7-5: IFS0: INTERRUPT FLAG STATUS REGISTER 0

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	AD1IF	U1TXIF	U1RXIF	SPI1IF	SPI1EIF	T3IF
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
T2IF	OC2IF	IC2IF	—	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 15-14	**Unimplemented:** Read as ‘0’
bit 13	**AD1IF:** ADC1 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	**Unimplemented:** Read as ‘0’
bit 3	**T1IF:** Timer1 Interrupt Flag Status bit 1 = Interrupt request has occurred 0 = Interrupt request has not occurred
bit 2	**OC1IF:** Output Compare Channel 1 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 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

REGISTER 7-6: IFS1: INTERRUPT FLAG STATUS REGISTER 1

U-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
—	—	INT2IF	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	U-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
IC8IF	IC7IF	—	INT1IF	CNIF	—	MI2C1IF	SI2C1IF
bit 7							bit 0

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

bit 15-14	**Unimplemented:** Read as ‘0’
bit 13	**INT2IF:** External Interrupt 2 Flag Status bit 1 = Interrupt request has occurred 0 = Interrupt request has not occurred
bit 12-8	**Unimplemented:** Read as ‘0’
bit 7	**IC8IF:** Input Capture Channel 8 Interrupt Flag Status bit 1 = Interrupt request has occurred 0 = Interrupt request has not occurred
bit 6	**IC7IF:** Input Capture Channel 7 Interrupt Flag Status bit 1 = Interrupt request has occurred 0 = Interrupt request has not occurred
bit 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	**Unimplemented:** Read as ‘0’
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: IFS4: INTERRUPT FLAG STATUS REGISTER 4

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

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	U-0
—	—	—	—	—	—	U1EIF	—
bit 7							bit 0

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

bit 15-2	**Unimplemented:** Read as ‘`0`’
bit 1	**U1EIF:** UART1 Error Interrupt Flag Status bit `1` = Interrupt request has occurred `0` = Interrupt request has not occurred
bit 0	**Unimplemented:** Read as ‘`0`’

REGISTER 7-8: IEC0: INTERRUPT ENABLE CONTROL REGISTER 0

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	AD1IE	U1TXIE	U1RXIE	SPI1IE	SPI1EIE	T3IE
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
T2IE	OC2IE	IC2IE	—	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 15-14 **Unimplemented:** Read as ‘0’

bit 13 **AD1IE:** 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 Error 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 **Unimplemented:** Read as ‘0’

bit 3 **T1IE:** Timer1 Interrupt Enable bit
1 = Interrupt request enabled
0 = Interrupt request not enabled

bit 2 **OC1IE:** Output Compare Channel 1 Interrupt Enable bit
1 = Interrupt request enabled
0 = Interrupt request not enabled

REGISTER 7-8: IEC0: INTERRUPT ENABLE CONTROL REGISTER 0 (CONTINUED)

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

REGISTER 7-9: IEC1: INTERRUPT ENABLE CONTROL REGISTER 1

U-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
—	—	INT2IE	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	U-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
IC8IE	IC7IE	—	INT1IE	CNIE	—	MI2C1IE	SI2C1IE
bit 7							bit 0

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

bit 15-14	**Unimplemented:** Read as ‘0’
bit 13	**INT2IE:** External Interrupt 2 Enable bit 1 = Interrupt request enabled 0 = Interrupt request not enabled
bit 12-8	**Unimplemented:** Read as ‘0’
bit 7	**IC8IE:** Input Capture Channel 8 Interrupt Enable bit 1 = Interrupt request enabled 0 = Interrupt request not enabled
bit 6	**IC7IE:** Input Capture Channel 7 Interrupt Enable bit 1 = Interrupt request enabled 0 = Interrupt request not enabled
bit 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	**Unimplemented:** Read as ‘0’
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

REGISTER 7-10: IEC4: INTERRUPT ENABLE CONTROL REGISTER 4

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

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	U-0
—	—	—	—	—	—	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 15-2	**Unimplemented:** Read as ‘`0`’
bit 1	**U1EIE:** UART1 Error Interrupt Enable bit `1` = Interrupt request enabled `0` = Interrupt request not enabled
bit 0	**Unimplemented:** Read as ‘`0`’

REGISTER 7-11: 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 7-12: 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	U-0	U-0	U-0
—	IC2IP<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 **Unimplemented:** Read as ‘`0`’

REGISTER 7-13: 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

REGISTER 7-14: IPC3: INTERRUPT PRIORITY CONTROL REGISTER 3

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
—	AD1IP<2:0>			—	U1TXIP<2:0>		
bit 7							bit 0

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

bit 15-7	**Unimplemented:** Read as ‘`0`’
bit 6-4	**AD1IP<2:0>:** ADC1 Conversion Complete Interrupt Priority bits `111` = Interrupt is priority 7 (highest priority interrupt) • • • `001` = Interrupt is priority 1 `000` = Interrupt source is disabled
bit 3	**Unimplemented:** Read as ‘`0`’
bit 2-0	**U1TXIP<2:0>:** UART1 Transmitter Interrupt Priority bits `111` = Interrupt is priority 7 (highest priority interrupt) • • • `001` = Interrupt is priority 1 `000` = Interrupt source is disabled

REGISTER 7-15: IPC4: INTERRUPT PRIORITY CONTROL REGISTER 4

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

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	MI2C1IP<2:0>			—	SI2C1IP<2:0>		
bit 7							bit 0

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

bit 15	**Unimplemented:** Read as ‘`0`’
bit 14-12	**CNIP<2:0>:** Change Notification Interrupt Priority bits `111` = Interrupt is priority 7 (highest priority interrupt) • • • `001` = Interrupt is priority 1 `000` = Interrupt source is disabled
bit 11-7	**Unimplemented:** Read as ‘`0`’
bit 6-4	**MI2C1IP<2:0>:** I2C1 Master Events Interrupt Priority bits `111` = Interrupt is priority 7 (highest priority interrupt) • • • `001` = Interrupt is priority 1 `000` = Interrupt source is disabled
bit 3	**Unimplemented:** Read as ‘`0`’
bit 2-0	**SI2C1IP<2:0>:** I2C1 Slave Events Interrupt Priority bits `111` = Interrupt is priority 7 (highest priority interrupt) • • • `001` = Interrupt is priority 1 `000` = Interrupt source is disabled

REGISTER 7-16: IPC5: INTERRUPT PRIORITY CONTROL REGISTER 5

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	IC8IP<2:0>			—	IC7IP<2:0>		
bit 15							bit 8

U-0	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 **Unimplemented:** Read as ‘`0`’

bit 14-12 **IC8IP<2:0>:** Input Capture Channel 8 Interrupt Priority bits

`111` = Interrupt is priority 7 (highest priority interrupt)
•
•
•
`001` = Interrupt is priority 1
`000` = Interrupt source is disabled

bit 11 **Unimplemented:** Read as ‘`0`’

bit 10-8 **IC7IP<2:0>:** Input Capture Channel 7 Interrupt Priority bits

`111` = Interrupt is priority 7 (highest priority interrupt)
•
•
•
`001` = Interrupt is priority 1
`000` = Interrupt source is disabled

bit 7-3 **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

REGISTER 7-17: IPC7: INTERRUPT PRIORITY 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	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	INT2IP<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 **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 3-0 **Unimplemented:** Read as ‘`0`’

REGISTER 7-18: IPC16: INTERRUPT PRIORITY CONTROL REGISTER 16

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

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	U1EIP<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	**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`’

REGISTER 7-19: INTTREG: INTERRUPT CONTROL AND STATUS REGISTER

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

U-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
—	VECNUM<6:0>						
bit 7							bit 0

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

bit 15-12	**Unimplemented:** Read as ‘`0`’
bit 11-8	**ILR<3:0>:** New CPU Interrupt Priority Level bits `1111` = CPU Interrupt Priority Level is 15 • • • `0001` = CPU Interrupt Priority Level is 1 `0000` = CPU Interrupt Priority Level is 0
bit 7	**Unimplemented:** Read as ‘`0`’
bit 6-0	**VECNUM<6:0>:** Vector Number of Pending Interrupt bits `0111111` = Interrupt Vector pending is number 135 • • • `0000001` = Interrupt Vector pending is number 9 `0000000` = Interrupt Vector pending is number 8

7.4 Interrupt Setup Procedures

7.4.1 INITIALIZATION

To configure an interrupt source 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 Interrupt Service Routine (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, the 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

All user interrupts can be disabled using this procedure:

1. Push the current SR value onto the software stack using the `PUSH` instruction.
2. Force the CPU to priority level 7 by inclusive ORing the value OEh with SRL.

To enable user interrupts, the `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.

NOTES:

8.0 OSCILLATOR CONFIGURATION

Note 1: This data sheet summarizes the features of the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 7. "Oscillator"** (DS70186) of the *"dsPIC33F/PIC24H Family Reference Manual"*, which is available from the Microchip website (www.microchip.com).

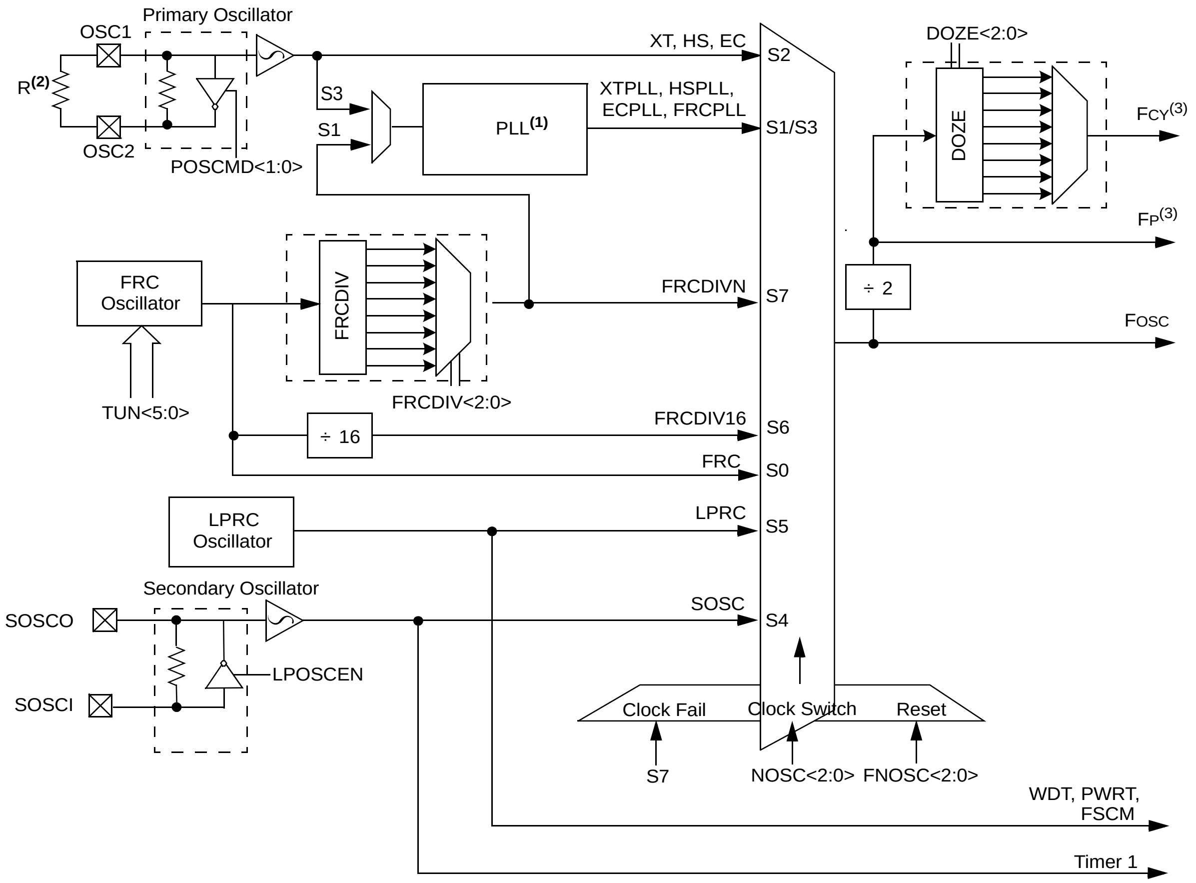
2: Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

The dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 oscillator system provides:

- External and internal oscillator options as clock sources.
- An on-chip 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.
- An Oscillator Control register (OSCCON).
- Nonvolatile Configuration bits for main oscillator selection.

A simplified diagram of the oscillator system is shown in Figure 8-1.

FIGURE 8-1: dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 OSCILLATOR SYSTEM DIAGRAM



Note 1: See Figure 8-2 for PLL details.

2: If the Oscillator is used with XT or HS modes, an external parallel resistor with the value of 1 MΩ must be connected.

3: The term FP refers to the clock source for all the peripherals, while FCY refers to the clock source for the CPU. Throughout this document FCY and FP are used interchangeably, except in the case of Doze mode. FP and FCY will be different when Doze mode is used in any ratio other than 1:1, which is the default.

8.1 CPU Clocking System

The dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 device provides seven system clock options:

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

8.1.1 SYSTEM CLOCK SOURCES

8.1.1.1 Fast RC

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> bits (CLKDIV<10:8>).

8.1.1.2 Primary

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

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

8.1.1.3 Secondary

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.

8.1.1.4 Low-Power RC

The Low-Power RC (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).

8.1.1.5 FRC

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 8.1.3 "PLL Configuration"**.

The FRC frequency depends on the FRC accuracy (see Table 22-18) and the value of the FRC Oscillator Tuning register (see Register 8-4).

8.1.2 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 19.1 "Configuration Bits"** for further details.) The Initial Oscillator Selection Configuration bits, FNOSC<2:0> (FOSCSEL<2:0>), and the Primary Oscillator Mode Select Configuration bits, POSCMD<1:0> (FOSC<1:0>), select the oscillator source that is used at a Power-on Reset. The FRC primary oscillator is the default (unprogrammed) selection.

The Configuration bits allow users to choose among 12 different clock modes, shown in Table 8-1.

The output of the oscillator (or the output of the PLL if a PLL mode has been selected) FOSC is divided by 2 to generate the device instruction clock (FCY) and the peripheral clock time base (FP). FCY defines the operating speed of the device, and speeds up to 40 MHz are supported by the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 architecture.

Instruction execution speed or device operating frequency, FCY, is given by:

EQUATION 8-1: DEVICE OPERATING FREQUENCY

$$F_{CY} = \frac{F_{OSC}}{2}$$

8.1.3 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 8-2.

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 80 MHz, which generates device operating speeds of 6.25-40 MIPS.

For a primary oscillator or FRC oscillator, output ‘FIN’, The PLL output ‘FOSC’ is given by:

EQUATION 8-2: FOSC CALCULATION

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

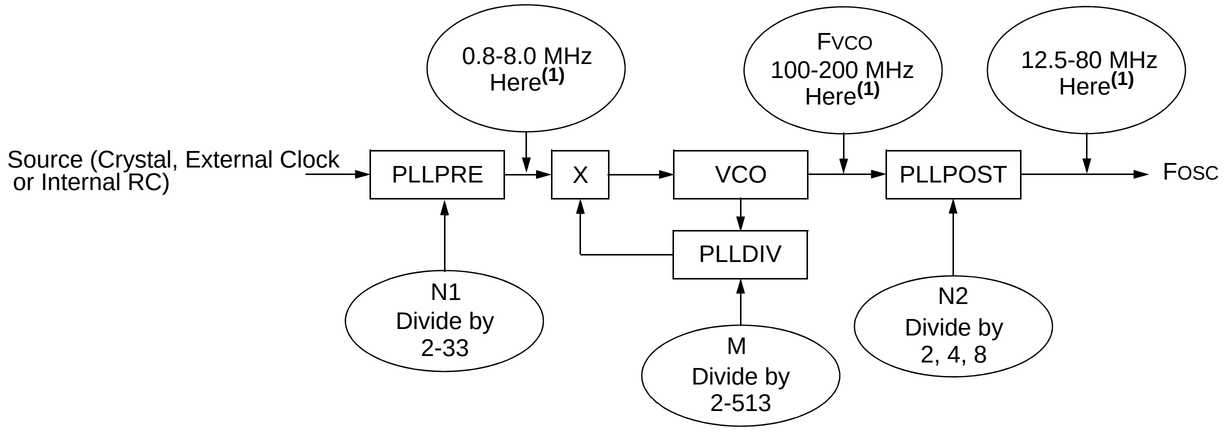
For example, suppose a 10 MHz crystal is being used, with “XT with PLL” being the selected oscillator mode.

- If PLLPRE<4:0> = `0`, then N1 = 2. This yields a VCO input of 10/2 = 5 MHz, which is within the acceptable range of 0.8-8 MHz.
- If PLLDIV<8:0> = 0x1E, then M = 32. This yields a VCO output of 5 x 32 = 160 MHz, which is within the 100-200 MHz ranged needed.
- If PLLPOST<1:0> = `0`, then N2 = 2. This provides a Fosc of 160/2 = 80 MHz. The resultant device operating speed is 80/2 = 40 MIPS.

EQUATION 8-3: XT WITH PLL MODE EXAMPLE

$$F_{CY} = \frac{F_{OSC}}{2} = \frac{1}{2} \cdot \left(\frac{10000000 \cdot 32}{2 \cdot 2}\right) = 40 \text{ MIPS}$$

FIGURE 8-2: dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 PLL BLOCK DIAGRAM



Note 1: This frequency range must be satisfied at all times.

TABLE 8-1: CONFIGURATION BIT VALUES FOR CLOCK SELECTION

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

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

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

REGISTER 8-1: OSCCON: OSCILLATOR CONTROL REGISTER(1,3)

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

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

Legend:	y = Value set from Configuration bits on POR		C = Clear only bit
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as ‘0’

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

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 **IOLOCK:** Peripheral Pin Select Lock bit
1 = Peripherial Pin Select is locked, write to peripheral pin select register is not allowed
0 = Peripherial Pin Select is unlocked, write to peripheral pin select register is allowed

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’

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

2: Direct clock switches between any primary oscillator mode with PLL and FRCPLL 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: This register is reset only on a Power-on Reset (POR).

REGISTER 8-1: OSCCON: OSCILLATOR CONTROL REGISTER[1,3] (CONTINUED)

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	**Unimplemented:** Read as '`0`'
bit 1	**LPOSCEN:** Secondary (LP) Oscillator Enable bit `1` = Enable secondary oscillator `0` = Disable secondary oscillator
bit 0	**OSWEN:** Oscillator Switch Enable bit `1` = Request oscillator switch to selection specified by NOSC<2:0> bits `0` = Oscillator switch is complete

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

2: Direct clock switches between any primary oscillator mode with PLL and FRCPLL 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: This register is reset only on a Power-on Reset (POR).

REGISTER 8-2: CLKDIV: CLOCK DIVISOR REGISTER[2]

R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0
ROI	DOZE<2:0>			DOZEN[1]	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:	y = Value set from Configuration bits on POR		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **ROI:** Recover on Interrupt bit

1 = Interrupts will clear the DOZEN bit and the processor clock/peripheral clock ratio is set to 1:1
0 = Interrupts have no effect on the DOZEN bit

bit 14-12 **DOZE<2:0>:** Processor Clock Reduction Select bits

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[1]

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’

Note 1: This bit is cleared when the ROI bit is set and an interrupt occurs.

2: This register is reset only on a Power-on Reset (POR).

REGISTER 8-2: CLKDIV: CLOCK DIVISOR REGISTER[2] (CONTINUED)

bit 4-0 **PLLPRE<4:0>:** PLL Phase Detector Input Divider bits (also denoted as ‘N1’, PLL prescaler)

`11111` = Input/33

•

•

•

`00001` = Input/3
`00000` = Input/2 (default)

Note 1: This bit is cleared when the ROI bit is set and an interrupt occurs.

2: This register is reset only on a Power-on Reset (POR).

REGISTER 8-3: PLLFBD: PLL FEEDBACK DIVISOR REGISTER[1]

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)

`111111111` = 513
•
•
•
`000110000` = 50 (default)
•
•
•
`000000010` = 4
`000000001` = 3
`000000000` = 2

Note 1: This register is reset only on a Power-on Reset (POR).

REGISTER 8-4: OSCTUN: FRC OSCILLATOR TUNING 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	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	TUN<5:0>[1]					
bit 7							bit 0

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

bit 15-6 **Unimplemented:** Read as ‘`0`’

bit 5-0 **TUN<5:0>:** FRC Oscillator Tuning bits[1]

`111111` = Center frequency - 0.375% (7.345 MHz)
•
•
•
`100001` = Center frequency - 11.625% (6.52 MHz)
`100000` = Center frequency - 12% (6.49 MHz)
`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)

Note 1: The OSCTUN functionality has been provided to help customers compensate for temperature effects on the FRC frequency over a wide range of temperatures. The tuning step size is an approximation and is neither characterized nor tested.

2: This register is reset only on a Power-on Reset (POR).

8.2 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, dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 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.

8.2.1 ENABLING CLOCK SWITCHING

To enable clock switching, the FCKSM1 Configuration bit in the Configuration register must be programmed to '`0`'. (Refer to **Section 19.1 "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.

8.2.2 OSCILLATOR SWITCHING SEQUENCE

Performing a clock switch requires this basic sequence:

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 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 status bits, LOCK (OSCCON<5>) and CF (OSCCON<3>) 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 FRC-PLL mode are not permitted. This applies to clock switches in either direction. In these instances, the application must switch to FRC mode as a transition clock source between the two PLL modes.

3: Refer to **Section 7. "Oscillator"** (DS70186) in the *"dsPIC33F/PIC24H Family Reference Manual"* for details.

8.3 Fail-Safe Clock Monitor (FSCM)

The Fail-Safe Clock Monitor (FSCM) allows the device to continue to operate even in the event of an oscillator failure. The FSCM function is enabled by programming. If the FSCM function is enabled, the LPRC internal oscillator runs at all times (except during Sleep mode) and is not subject to control by the Watchdog Timer.

In the event of an oscillator failure, the FSCM generates a clock failure trap event and switches the system clock over to the FRC oscillator. Then the application program can either attempt to restart the oscillator or execute a controlled shutdown. The trap can be treated as a warm Reset by simply loading the Reset address into the oscillator fail trap vector.

If the PLL multiplier is used to scale the system clock, the internal FRC is also multiplied by the same factor on clock failure. Essentially, the device switches to FRC with PLL on a clock failure.

9.0 POWER-SAVING FEATURES

> **Note 1:** This data sheet summarizes the features of the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 family 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) of the *"dsPIC33F/PIC24H Family Reference Manual"*, which is available from the Microchip website (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 dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 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. dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 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.

9.1 Clock Frequency and Clock Switching

dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 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 8.0 "Oscillator Configuration"**.

9.2 Instruction-Based Power-Saving Modes

dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 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 9-1.

> **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.

9.2.1 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 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 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 the 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.

EXAMPLE 9-1: PWRSAV INSTRUCTION SYNTAX

```
PWRSAV  #SLEEP_MODE    ; Put the device into SLEEP mode
PWRSAV  #IDLE_MODE     ; Put the device into IDLE mode
```

9.2.2 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 9.4 "Peripheral Module Disable"**).
- If the WDT or FSCM is enabled, the LPRC also remains active.

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

- Any interrupt that is individually enabled.
- Any form of 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 cycles later), starting with the instruction following the `PWRSAV` instruction, or the first instruction in the ISR.

9.2.3 INTERRUPTS COINCIDENT WITH POWER SAVE INSTRUCTIONS

Any interrupt that coincides with the execution of a `PWRSAV` instruction is held off until entry into Sleep or Idle mode has completed. The device then wakes up from Sleep or Idle mode.

9.3 Doze Mode

The preferred strategies for reducing power consumption are changing clock speed and invoking one of the power-saving modes. In some circumstances, however, these are not practical. For example, it may be necessary for an application to maintain uninterrupted synchronous communication, even while it is doing nothing else. Reducing system clock speed 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.

9.4 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.

Note: 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).

REGISTER 9-1: PMD1: PERIPHERAL MODULE DISABLE CONTROL REGISTER 1

U-0	U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
—	—	T3MD	T2MD	T1MD	—	—	—
bit 15							bit 8

R/W-0	U-0	R/W-0	U-0	R/W-0	U-0	U-0	R/W-0
I2C1MD	—	U1MD	—	SPI1MD	—	—	AD1MD[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 15-14	**Unimplemented:** Read as ‘0’
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 10-8	**Unimplemented:** Read as ‘0’
bit 7	**I2C1MD:** I2C1 Module Disable bit 1 = I2C1 module is disabled 0 = I2C1 module is enabled
bit 6	**Unimplemented:** Read as ‘0’
bit 5	**U1MD:** UART1 Module Disable bit 1 = UART1 module is disabled 0 = UART1 module is enabled
bit 4	**Unimplemented:** Read as ‘0’
bit 3	**SPI1MD:** SPI1 Module Disable bit 1 = SPI1 module is disabled 0 = SPI1 module is enabled
bit 2-1	**Unimplemented:** Read as ‘0’
bit 0	**AD1MD:** ADC1 Module Disable bit[1] 1 = ADC1 module is disabled 0 = ADC1 module is enabled

Note 1: PCFGx bits have no effect if the ADC module is disabled by setting this bit. In this case, all port pins multiplexed with ANx will be in Digital mode.

REGISTER 9-2: PMD2: PERIPHERAL MODULE DISABLE CONTROL REGISTER 2

R/W-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
IC8MD	IC7MD	—	—	—	—	IC2MD	IC1MD
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	OC2MD	OC1MD
bit 7							bit 0

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

bit 15 **IC8MD:** Input Capture 8 Module Disable bit
1 = Input Capture 8 module is disabled
0 = Input Capture 8 module is enabled

bit 14 **IC7MD:** Input Capture 2 Module Disable bit
1 = Input Capture 7 module is disabled
0 = Input Capture 7 module is enabled

bit 13-10 **Unimplemented:** Read as ‘0’

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 7-2 **Unimplemented:** Read as ‘0’

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

10.0 I/O PORTS

Note 1: This data sheet summarizes the features of the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 family 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) of the *"dsPIC33F/PIC24H Family Reference Manual"*, which is available from the Microchip website (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 V_{DD}, V_{SS}, $\overline{\text{MCLR}}$ 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.

10.1 Parallel I/O (PIO) Ports

A parallel I/O port that shares a pin with a peripheral is generally 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 10-1 shows how ports are shared with other peripherals and the associated I/O pin to which they are connected.

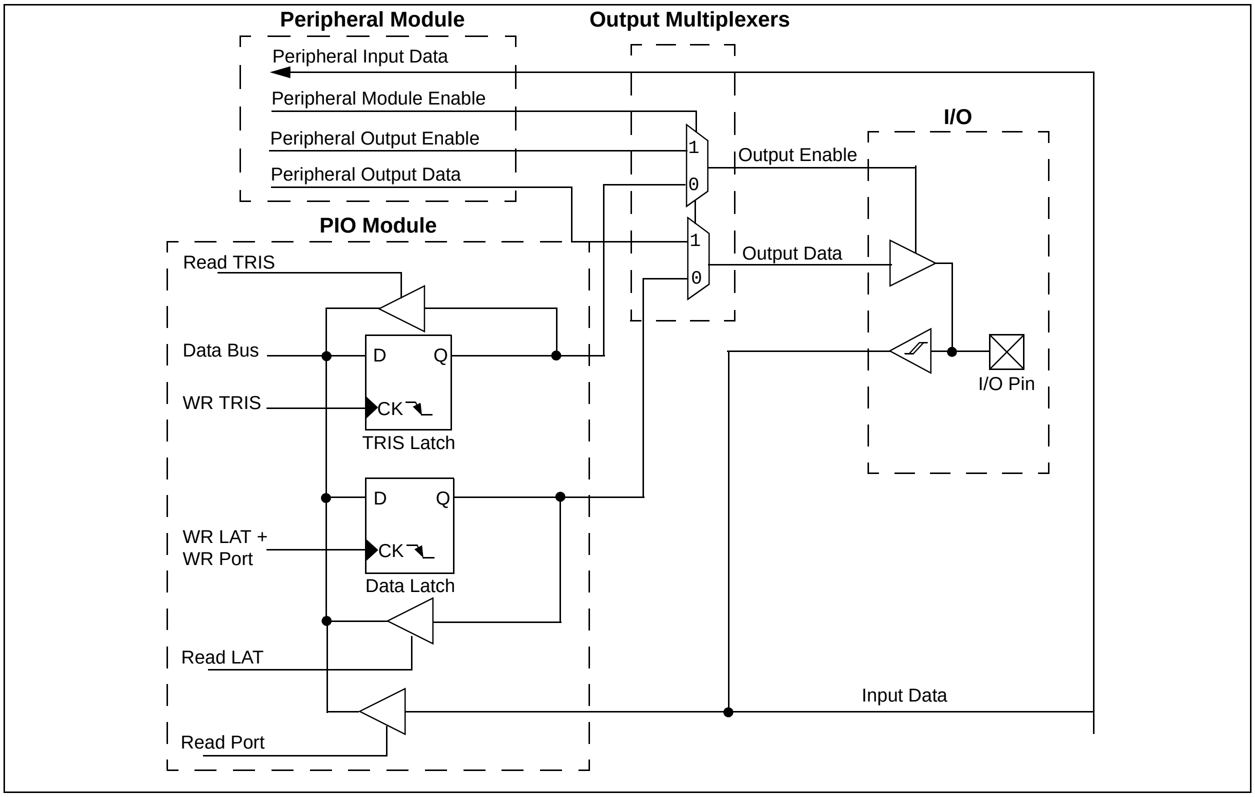
When a peripheral is enabled and the peripheral is actively driving an associated pin, the use of the pin as a general purpose output pin is disabled. The I/O pin can be read, but the output driver for the parallel port bit is disabled. If a peripheral is enabled, but the peripheral is not actively driving 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 a '1', then the pin is an input. All port pins are defined as inputs after a Reset. Reads from the latch (LATx) read the latch. Writes to the latch, write the latch. Reads from the port (PORTx) read the port pins, while writes to the port pins write the latch.

Any bit and its associated data and control registers that are not valid for a particular device will be disabled. That means the corresponding LATx and TRISx registers and the port 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 10-1: BLOCK DIAGRAM OF A TYPICAL SHARED PORT STRUCTURE



10.2 Open-Drain Configuration

In addition to the PORT, LAT and TRIS registers for data control, some port pins can also be individually configured for either digital or open-drain output. This is controlled by the Open-Drain Control register, ODCx, associated with each port. Setting any of the bits configures the corresponding pin to act as an open-drain output.

The open-drain feature allows the generation of outputs higher than V_{DD} (e.g., 5V) on any desired 5V tolerant pins by using external pull-up resistors. The maximum open-drain voltage allowed is the same as the maximum V_{IH} specification.

See **"Pin Diagrams"** for the available pins and their functionality.

10.3 Configuring Analog Port Pins

The AD1PCFG and TRIS registers control the operation of the analog-to-digital (A/D) port pins. The port pins that are desired as analog inputs must have their corresponding TRIS bit set (input). If the TRIS bit is cleared (output), the digital output level (V_{OH} or V_{OL}) will be converted.

The AD1PCFGL register has a default value of 0x0000; 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 that is defined as a digital input (including the ANx pins) can cause the input buffer to consume current that exceeds the device specifications.

10.4 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`. Examples are shown in Example 10-1 and Example 10-2. This also applies to PORT bit operations, such as `BSET PORTB, # RB0`, which are single cycle read-modify-write. All PORT bit operations, such as `MOV PORTB, W0` or `BSET PORTB, # RBx`, read the pin and *not* the latch.

10.5 Input Change Notification

The input change notification function of the I/O ports allows the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 devices to generate interrupt requests to the processor in response to a change-of-state 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 31 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 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.

EXAMPLE 10-1: PORT WRITE/READ

```
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
```

EXAMPLE 10-2: PORT BIT OPERATIONS

```
Incorrect:
BSET    PORTB, #RB1         ;Set PORTB<RB1> high
BSET    PORTB, #RB6         ;Set PORTB<RB6> high

Correct:
BSET    PORTB, #RB1         ;Set PORTB<RB1> high
NOP
BSET    PORTB, #RB6         ;Set PORTB<RB6> high
NOP

Preferred:
BSET    LATB, LATB1         ;Set PORTB<RB1> high
BSET    LATB, LATB6         ;Set PORTB<RB6> high
```

10.6 Peripheral Pin Select

A major challenge in general-purpose devices is providing the largest possible set of peripheral features while minimizing the conflict of features on I/O pins. The challenge is even greater on low-pin count devices. In an application where more than one peripheral must be assigned to a single pin, inconvenient workarounds in application code or a complete redesign may be the only option.

Peripheral pin select configuration enables peripheral set selection and placement on a wide range of I/O pins. By increasing the pinout options available on a particular device, programmers can better tailor the microcontroller to their entire application, rather than trimming the application to fit the device.

The peripheral pin select configuration feature operates over a fixed subset of digital I/O pins. Programmers can independently map the input and/or output of most digital peripherals to any one of these I/O pins. Peripheral pin select is performed in software, and generally does not require the device to be reprogrammed. Hardware safeguards are included that prevent accidental or spurious changes to the peripheral mapping, once it has been established.

10.6.1 AVAILABLE PINS

The peripheral pin select feature is used with a range of up to 26 pins. The number of available pins depends on the particular device and its pin count. Pins that support the peripheral pin select feature include the designation “RPn” in their full pin designation, where “RP” designates a remappable peripheral and “n” is the remappable pin number.

10.6.2 CONTROLLING PERIPHERAL PIN SELECT

Peripheral pin select features are controlled through two sets of special function registers: one to map peripheral inputs, and one to map outputs. Because they are separately controlled, a particular peripheral’s input and output (if the peripheral has both) can be placed on any selectable function pin without constraint.

The association of a peripheral to a peripheral selectable pin is handled in two different ways, depending on whether an input or output is being mapped.

10.6.2.1 Input Mapping

The inputs of the peripheral pin select options are mapped on the basis of the peripheral. A control register associated with a peripheral dictates the pin it will be mapped to. The RPINRx registers are used to configure peripheral input mapping (see Register 10-1 through Register 10-9). Each register contains sets of 5-bit fields, with each set associated with one of the remappable peripherals. Programming a given peripheral’s bit field with an appropriate 5-bit value maps the RPn pin with that value to that peripheral. For any given device, the valid range of values for any bit field corresponds to the maximum number of peripheral pin selections supported by the device.

Figure 10-2 Illustrates remappable pin selection for U1RX input.

Note:	For input mapping only, the Peripheral Pin Select (PPS) functionality does not have priority over the TRISx settings. Therefore, when configuring the RPn pin for input, the corresponding bit in the TRISx register must also be configured for input (i.e., set to ‘1’).

FIGURE 10-2: REMAPPABLE MUX INPUT FOR U1RX

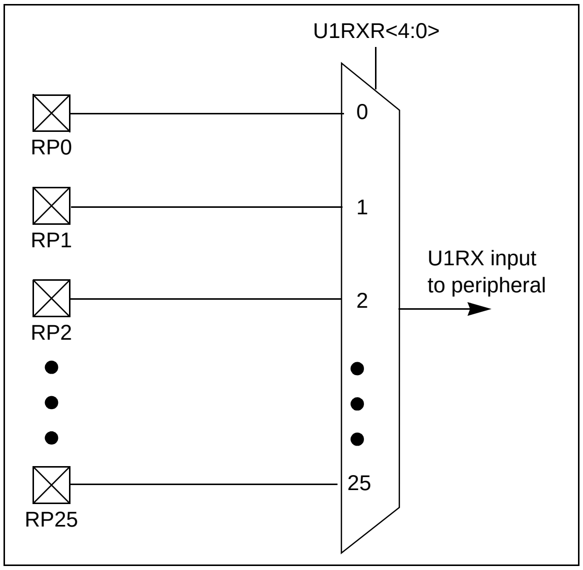


TABLE 10-1: REMAPPABLE PERIPHERAL INPUTS(1)

Input Name	Function Name	Register	Configuration Bits
External Interrupt 1	INT1	RPINR0	INT1R<4:0>
External Interrupt 2	INT2	RPINR1	INT2R<4:0>
Timer2 External Clock	T2CK	RPINR3	T2CKR<4:0>
Timer3 External Clock	T3CK	RPINR3	T3CKR<4:0>
Input Capture 1	IC1	RPINR7	IC1R<4:0>
Input Capture 2	IC2	RPINR7	IC2R<4:0>
Input Capture 7	IC7	RPINR10	IC7R<4:0>
Input Capture 8	IC8	RPINR10	IC8R<4:0>
Output Compare Fault A	OCFA	RPINR11	OCFAR<4:0>
UART1 Receive	U1RX	RPINR18	U1RXR<4:0>
UART1 Clear To Send	$\overline{\text{U1CTS}}$	RPINR18	U1CTSR<4:0>
SPI1 Data Input	SDI1	RPINR20	SDI1R<4:0>
SPI1 Clock Input	SCK1IN	RPINR20	SCK1R<4:0>
SPI1 Slave Select Input	SS1IN	RPINR21	SS1R<4:0>

Note 1: Unless otherwise noted, all inputs use the Schmitt input buffers.

10.6.2.2 Output Mapping

In contrast to inputs, the outputs of the peripheral pin select options are mapped on the basis of the pin. In this case, a control register associated with a particular pin dictates the peripheral output to be mapped. The RPORx registers are used to control output mapping. Like the RPINRx registers, each register contains sets of 5-bit fields, with each set associated with one RPn pin (see Register 10-10 through Register 10-22). The value of the bit field corresponds to one of the peripherals, and that peripheral's output is mapped to the pin (see Table 10-2 and Figure 10-3).

The list of peripherals for output mapping also includes a null value of '`00000`' because of the mapping technique. This permits any given pin to remain unconnected from the output of any of the pin selectable peripherals.

FIGURE 10-3: MULTIPLEXING OF REMAPPABLE OUTPUT FOR RPn

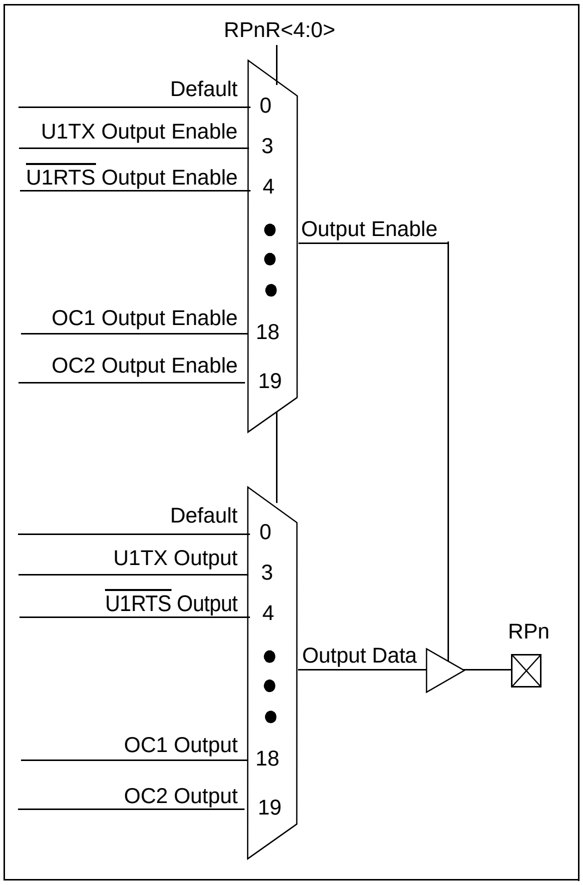


TABLE 10-2: OUTPUT SELECTION FOR REMAPPABLE PIN (RPn)

Function	RPnR<4:0>	Output Name
NULL	`00000`	RPn tied to default port pin
U1TX	`00011`	RPn tied to UART1 Transmit
$\overline{\text{U1RTS}}$	`00100`	RPn tied to UART1 Ready To Send
SDO1	`00111`	RPn tied to SPI1 Data Output
SCK1OUT	`01000`	RPn tied to SPI1 Clock Output
SS1OUT	`01001`	RPn tied to SPI1 Slave Select Output
OC1	`10010`	RPn tied to Output Compare 1
OC2	`10011`	RPn tied to Output Compare 2

10.6.3 CONTROLLING CONFIGURATION CHANGES

Because peripheral remapping can be changed during run time, some restrictions on peripheral remapping are needed to prevent accidental configuration changes. dsPIC33F devices include three features to prevent alterations to the peripheral map:

- Control register lock sequence
- Continuous state monitoring
- Configuration bit pin select lock

10.6.3.1 Control Register Lock

Under normal operation, writes to the RPINRx and RPORx registers are not allowed. Attempted writes appear to execute normally, but the contents of the registers remain unchanged. To change these registers, they must be unlocked in hardware. The register lock is controlled by the IOLOCK bit (OSCCON<6>). Setting IOLOCK prevents writes to the control registers; clearing IOLOCK allows writes.

To set or clear IOLOCK, a specific command sequence must be executed:

1. Write 0x46 to OSCCON<7:0>.
2. Write 0x57 to OSCCON<7:0>.
3. Clear (or set) IOLOCK as a single operation.

Note: MPLAB® C30 provides built-in C language functions for unlocking the OSCCON register:

```
__builtin_write_OSCCONL(value)
__builtin_write_OSCCONH(value)
```

See MPLAB® IDE Help for more information.

Unlike the similar sequence with the oscillator’s LOCK bit, IOLOCK remains in one state until changed. This allows all of the peripheral pin selects to be configured with a single unlock sequence followed by an update to all control registers, then locked with a second lock sequence.

10.6.3.2 Continuous State Monitoring

In addition to being protected from direct writes, the contents of the RPINRx and RPORx registers are constantly monitored in hardware by shadow registers. If an unexpected change in any of the registers occurs (such as cell disturbances caused by ESD or other external events), a configuration mismatch Reset will be triggered.

10.6.3.3 Configuration Bit Pin Select Lock

As an additional level of safety, the device can be configured to prevent more than one write session to the RPINRx and RPORx registers. The IOL1WAY configuration bit (FOSC<5>) blocks the IOLOCK bit from being cleared after it has been set once.

In the default (unprogrammed) state, IOL1WAY is set, restricting users to one write session. Programming IOL1WAY allows user applications unlimited access (with the proper use of the unlock sequence) to the peripheral pin select registers.

10.7 Peripheral Pin Select Registers

The dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 devices implement 17 registers for remappable peripheral configuration:

- Input Remappable Peripheral Registers (9)
- Output Remappable Peripheral Registers (8)

Note: Input and Output Register values can only be changed if OSCCON[IOLOCK] = `0`. See **Section 10.6.3.1 "Control Register Lock"** for a specific command sequence.

REGISTER 10-1: RPINR0: PERIPHERAL PIN SELECT INPUT REGISTER 0

U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	—	INT1R<4:0>				
bit 15							bit 8

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

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

bit 15-13 **Unimplemented:** Read as '`0`'

bit 12-8 **INT1R<4:0>:** Assign External Interrupt 1 (INTR1) to the corresponding RPn pin

`11111` = Input tied to Vss
`11001` = Input tied to RP25
•
•
•
`00001` = Input tied to RP1
`00000` = Input tied to RP0

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

REGISTER 10-2: RPINR1: PERIPHERAL PIN SELECT INPUT REGISTER 1

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

U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	—	INT2R<4:0>				
bit 7							bit 0

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

bit 15-5 **Unimplemented:** Read as ‘`0`’

bit 4-0 **INT2R<4:0>:** Assign External Interrupt 2 (INTR2) to the corresponding RPn pin

`11111` = Input tied to Vss
`11001` = Input tied to RP25

•

•

•

`00001` = Input tied to RP1
`00000` = Input tied to RP0

REGISTER 10-3: RPINR3: PERIPHERAL PIN SELECT INPUT REGISTER 3

U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	—	T3CKR<4:0>				
bit 15							bit 8

U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	—	T2CKR<4:0>				
bit 7							bit 0

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

bit 15-13 **Unimplemented:** Read as ‘`0`’

bit 12-8 **T3CKR<4:0>:** Assign Timer3 External Clock (T3CK) to the corresponding RPn pin

`11111` = Input tied to Vss
`11001` = Input tied to RP25
•
•
•
`00001` = Input tied to RP1
`00000` = Input tied to RP0

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

bit 4-0 **T2CKR<4:0>:** Assign Timer2 External Clock (T2CK) to the corresponding RPn pin

`11111` = Input tied to Vss
`11001` = Input tied to RP25
•
•
•
`00001` = Input tied to RP1
`00000` = Input tied to RP0

REGISTER 10-4: RPINR7: PERIPHERAL PIN SELECT INPUT REGISTER 7

U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	—	IC2R<4:0>				
bit 15							bit 8

U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	—	IC1R<4:0>				
bit 7							bit 0

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

bit 15-13 **Unimplemented:** Read as ‘`0`’

bit 12-8 **IC2R<4:0>:** Assign Input Capture 2 (IC2) to the corresponding RPn pin

`11111` = Input tied to Vss
`11001` = Input tied to RP25
•
•
•
`00001` = Input tied to RP1
`00000` = Input tied to RP0

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

bit 4-0 **IC1R<4:0>:** Assign Input Capture 1 (IC1) to the corresponding RPn pin

`11111` = Input tied to Vss
`11001` = Input tied to RP25
•
•
•
`00001` = Input tied to RP1
`00000` = Input tied to RP0

REGISTER 10-5: RPINR10: PERIPHERAL PIN SELECT INPUT REGISTER 10

U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	—	IC8R<4:0>				
bit 15							bit 8

U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	—	IC7R<4:0>				
bit 7							bit 0

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

bit 15-13 **Unimplemented:** Read as ‘`0`’

bit 12-8 **IC8R<4:0>:** Assign Input Capture 8 (IC8) to the corresponding RPn pin

`11111` = Input tied to Vss
`11001` = Input tied to RP25
•
•
•
`00001` = Input tied to RP1
`00000` = Input tied to RP0

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

bit 4-0 **IC7R<4:0>:** Assign Input Capture 7 (IC7) to the corresponding RPn pin

`11111` = Input tied to Vss
`11001` = Input tied to RP25
•
•
•
`00001` = Input tied to RP1
`00000` = Input tied to RP0

REGISTER 10-6: RPINR11: PERIPHERAL PIN SELECT INPUT REGISTER 11

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

U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	—	OCFAR<4:0>				
bit 7							bit 0

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

bit 15-5	**Unimplemented:** Read as ‘`0`’
bit 4-0	**OCFAR<4:0>:** Assign Output Capture A (OCFA) to the corresponding RPn pin `11111` = Input tied to Vss `11001` = Input tied to RP25 • • • `00001` = Input tied to RP1 `00000` = Input tied to RP0

REGISTER 10-7: RPINR18: PERIPHERAL PIN SELECT INPUT REGISTER 18

U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	—	U1CTSR<4:0>				
bit 15							bit 8

U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	—	U1RXR<4:0>				
bit 7							bit 0

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

bit 15-13 **Unimplemented:** Read as ‘`0`’

bit 12-8 **U1CTSR<4:0>:** Assign UART 1 Clear to Send ($\overline{\text{U1CTS}}$) to the corresponding RPn pin

`11111` = Input tied to Vss
`11001` = Input tied to RP25
•
•
•
`00001` = Input tied to RP1
`00000` = Input tied to RP0

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

bit 4-0 **U1RXR<4:0>:** Assign UART 1 Receive (U1RX) to the corresponding RPn pin

`11111` = Input tied to Vss
`11001` = Input tied to RP25
•
•
•
`00001` = Input tied to RP1
`00000` = Input tied to RP0

REGISTER 10-8: RPINR20: PERIPHERAL PIN SELECT INPUT REGISTER 20

U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	—	SCK1R<4:0>				
bit 15							bit 8

U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	—	SDI1R<4:0>				
bit 7							bit 0

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

bit 15-13	**Unimplemented:** Read as ‘0’
bit 12-8	**SCK1R<4:0>:** Assign SPI 1 Clock Input (SCK1IN) to the corresponding RPn pin `11111` = Input tied to Vss `11001` = Input tied to RP25 • • • `00001` = Input tied to RP1 `00000` = Input tied to RP0
bit 7-5	**Unimplemented:** Read as ‘0’
bit 4-0	**SDI1R<4:0>:** Assign SPI 1 Data Input (SDI1) to the corresponding RPn pin `11111` = Input tied to Vss `11001` = Input tied to RP25 • • • `00001` = Input tied to RP1 `00000` = Input tied to RP0

REGISTER 10-9: RPINR21: PERIPHERAL PIN SELECT INPUT REGISTER 21

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

U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	—	SS1R<4:0>				
bit 7							bit 0

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

bit 15-5 **Unimplemented:** Read as ‘`0`’

bit 4-0 **SS1R<4:0>:** Assign SPI1 Slave Select Input (SS1IN) to the corresponding RPn pin

`11111` = Input tied to Vss
`11001` = Input tied to RP25
•
•
•
`00001` = Input tied to RP1
`00000` = Input tied to RP0

REGISTER 10-10: RPOR0: PERIPHERAL PIN SELECT OUTPUT REGISTER 0

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	RP1R<4:0>				
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
—	—	—	RP0R<4:0>				
bit 7							bit 0

Legend:

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

bit 15-13	**Unimplemented:** Read as ‘0’
bit 12-8	**RP1R<4:0>:** Peripheral Output Function is Assigned to RP1 Output Pin (see Table 10-2 for peripheral function numbers)
bit 7-5	**Unimplemented:** Read as ‘0’
bit 4-0	**RP0R<4:0>:** Peripheral Output Function is Assigned to RP0 Output Pin (see Table 10-2 for peripheral function numbers)

REGISTER 10-11: RPOR1: PERIPHERAL PIN SELECT OUTPUT REGISTER 1

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	RP3R<4:0>				
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
—	—	—	RP2R<4:0>				
bit 7							bit 0

Legend:

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

bit 15-13	**Unimplemented:** Read as ‘0’
bit 12-8	**RP3R<4:0>:** Peripheral Output Function is Assigned to RP3 Output Pin (see Table 10-2 for peripheral function numbers)
bit 7-5	**Unimplemented:** Read as ‘0’
bit 4-0	**RP2R<4:0>:** Peripheral Output Function is Assigned to RP2 Output Pin (see Table 10-2 for peripheral function numbers)

REGISTER 10-12: RPOR2: PERIPHERAL PIN SELECT OUTPUT REGISTER 2

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	RP5R<4:0>				
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
—	—	—	RP4R<4:0>				
bit 7							bit 0

Legend:

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

bit 15-13	**Unimplemented:** Read as ‘0’
bit 12-8	**RP5R<4:0>:** Peripheral Output Function is Assigned to RP5 Output Pin (see Table 10-2 for peripheral function numbers)
bit 7-5	**Unimplemented:** Read as ‘0’
bit 4-0	**RP4R<4:0>:** Peripheral Output Function is Assigned to RP4 Output Pin (see Table 10-2 for peripheral function numbers)

REGISTER 10-13: RPOR3: PERIPHERAL PIN SELECT OUTPUT REGISTER 3

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	RP7R<4:0>				
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
—	—	—	RP6R<4:0>				
bit 7							bit 0

Legend:

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

bit 15-13	**Unimplemented:** Read as ‘0’
bit 12-8	**RP7R<4:0>:** Peripheral Output Function is Assigned to RP7 Output Pin (see Table 10-2 for peripheral function numbers)
bit 7-5	**Unimplemented:** Read as ‘0’
bit 4-0	**RP6R<4:0>:** Peripheral Output Function is Assigned to RP6 Output Pin (see Table 10-2 for peripheral function numbers)

REGISTER 10-14: RPOR4: PERIPHERAL PIN SELECT OUTPUT REGISTER 4

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	RP9R<4:0>				
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
—	—	—	RP8R<4:0>				
bit 7							bit 0

Legend:

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

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

bit 15-13	**Unimplemented:** Read as ‘`0`’
bit 12-8	**RP9R<4:0>:** Peripheral Output Function is Assigned to RP9 Output Pin (see Table 10-2 for peripheral function numbers)
bit 7-5	**Unimplemented:** Read as ‘`0`’
bit 4-0	**RP8R<4:0>:** Peripheral Output Function is Assigned to RP8 Output Pin (see Table 10-2 for peripheral function numbers)

REGISTER 10-15: RPOR5: PERIPHERAL PIN SELECT OUTPUT REGISTER 5

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	RP11R<4:0>				
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
—	—	—	RP10R<4:0>				
bit 7							bit 0

Legend:

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

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

bit 15-13	**Unimplemented:** Read as ‘`0`’
bit 12-8	**RP11R<4:0>:** Peripheral Output Function is Assigned to RP11 Output Pin (see Table 10-2 for peripheral function numbers)
bit 7-5	**Unimplemented:** Read as ‘`0`’
bit 4-0	**RP10R<4:0>:** Peripheral Output Function is Assigned to RP10 Output Pin (see Table 10-2 for peripheral function numbers)

REGISTER 10-16: RPOR6: PERIPHERAL PIN SELECT OUTPUT REGISTER 6

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	RP13R<4:0>				
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
—	—	—	RP12R<4:0>				
bit 7							bit 0

Legend:

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

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

bit 15-13	**Unimplemented:** Read as ‘0’
bit 12-8	**RP13R<4:0>:** Peripheral Output Function is Assigned to RP13 Output Pin (see Table 10-2 for peripheral function numbers)
bit 7-5	**Unimplemented:** Read as ‘0’
bit 4-0	**RP12R<4:0>:** Peripheral Output Function is Assigned to RP12 Output Pin (see Table 10-2 for peripheral function numbers)

REGISTER 10-17: RPOR7: PERIPHERAL PIN SELECT OUTPUT REGISTER 7

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	RP15R<4:0>				
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
—	—	—	RP14R<4:0>				
bit 7							bit 0

Legend:

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

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

bit 15-13	**Unimplemented:** Read as ‘0’
bit 12-8	**RP15R<4:0>:** Peripheral Output Function is Assigned to RP15 Output Pin (see Table 10-2 for peripheral function numbers)
bit 7-5	**Unimplemented:** Read as ‘0’
bit 4-0	**RP14R<4:0>:** Peripheral Output Function is Assigned to RP14 Output Pin (see Table 10-2 for peripheral function numbers)

REGISTER 10-18: RPOR8: PERIPHERAL PIN SELECT OUTPUT REGISTER 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	RP17R<4:0>				
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
—	—	—	RP16R<4:0>				
bit 7							bit 0

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

bit 15-13	**Unimplemented:** Read as ‘0’
bit 12-8	**RP17R<4:0>:** Peripheral Output Function is Assigned to RP17 Output Pin (see Table 10-2 for peripheral function numbers)
bit 7-5	**Unimplemented:** Read as ‘0’
bit 4-0	**RP16R<4:0>:** Peripheral Output Function is Assigned to RP16 Output Pin (see Table 10-2 for peripheral function numbers)

REGISTER 10-19: RPOR9: PERIPHERAL PIN SELECT OUTPUT REGISTER 9

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	RP19R<4:0>				
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
—	—	—	RP18R<4:0>				
bit 7							bit 0

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

bit 15-13	**Unimplemented:** Read as ‘0’
bit 12-8	**RP19R<4:0>:** Peripheral Output Function is Assigned to RP19 Output Pin (see Table 10-2 for peripheral function numbers)
bit 7-5	**Unimplemented:** Read as ‘0’
bit 4-0	**RP18R<4:0>:** Peripheral Output Function is Assigned to RP18 Output Pin (see Table 10-2 for peripheral function numbers)

REGISTER 10-20: RPOR10: PERIPHERAL PIN SELECT OUTPUT REGISTER 10

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	RP21R<4:0>				
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
—	—	—	RP20R<4:0>				
bit 7							bit 0

Legend:

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

bit 15-13	**Unimplemented:** Read as ‘`0`’
bit 12-8	**RP21R<4:0>:** Peripheral Output Function is Assigned to RP21 Output Pin (see Table 10-2 for peripheral function numbers)
bit 7-5	**Unimplemented:** Read as ‘`0`’
bit 4-0	**RP20R<4:0>:** Peripheral Output Function is Assigned to RP20 Output Pin (see Table 10-2 for peripheral function numbers)

REGISTER 10-21: RPOR11: PERIPHERAL PIN SELECT OUTPUT REGISTER 11

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	RP23R<4:0>				
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
—	—	—	RP22R<4:0>				
bit 7							bit 0

Legend:

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

bit 15-13	**Unimplemented:** Read as ‘`0`’
bit 12-8	**RP23R<4:0>:** Peripheral Output Function is Assigned to RP23 Output Pin (see Table 10-2 for peripheral function numbers)
bit 7-5	**Unimplemented:** Read as ‘`0`’
bit 4-0	**RP22R<4:0>:** Peripheral Output Function is Assigned to RP22 Output Pin (see Table 10-2 for peripheral function numbers)

REGISTER 10-22: RPOR12: PERIPHERAL PIN SELECT OUTPUT REGISTER 12

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	RP25R<4:0>				
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
—	—	—	RP24R<4:0>				
bit 7							bit 0

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

bit 15-13	**Unimplemented:** Read as ‘0’
bit 12-8	**RP25R<4:0>:** Peripheral Output Function is Assigned to RP25 Output Pin (see Table 10-2 for peripheral function numbers)
bit 7-5	**Unimplemented:** Read as ‘0’
bit 4-0	**RP24R<4:0>:** Peripheral Output Function is Assigned to RP24 Output Pin (see Table 10-2 for peripheral function numbers)

NOTES:

11.0 TIMER1

> **Note 1:** This data sheet summarizes the features of the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 family 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) of the *"dsPIC33F/PIC24H Family Reference Manual"*, which is available from the Microchip website (www.microchip.com).
>
> **2:** Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

The Timer1 module is a 16-bit timer, which can serve as the time counter for the real-time clock, or operate as a free-running interval timer/counter. Timer1 can operate in three modes:

- 16-bit Timer
- 16-bit Synchronous Counter
- 16-bit Asynchronous Counter

Timer1 also supports these features:

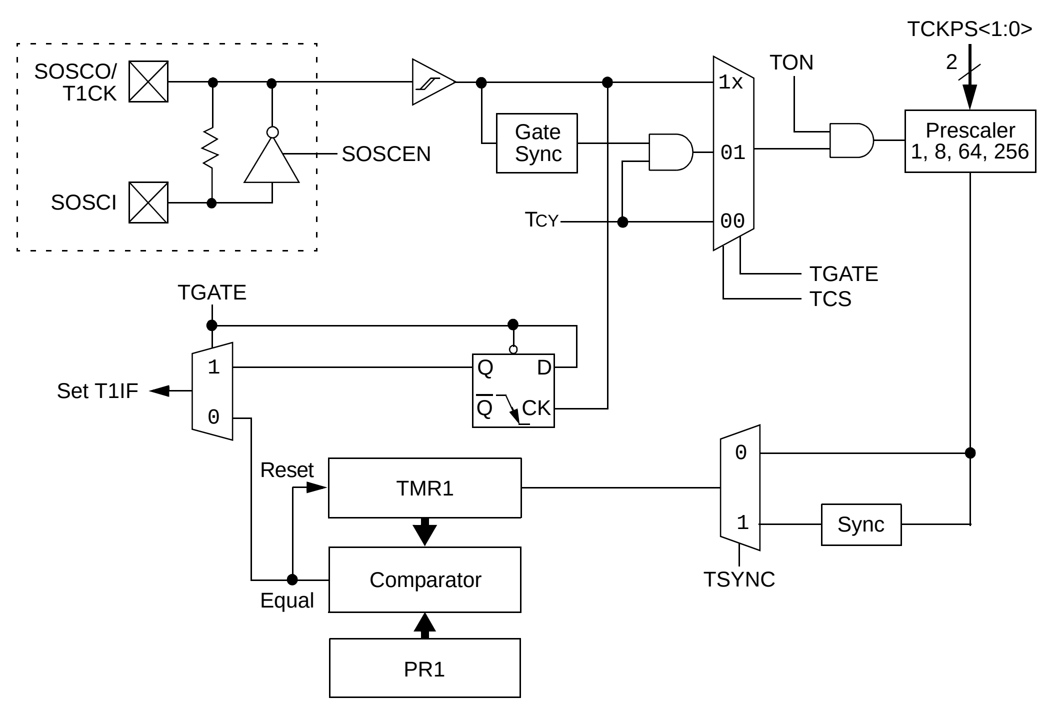
- Timer gate operation
- Selectable prescaler settings
- Timer operation during CPU Idle and Sleep modes
- Interrupt on 16-bit Period register match or falling edge of external gate signal

Figure 11-1 illustrates a block diagram of the 16-bit timer module.

To configure Timer1 for operation:

1. Set the TON bit (= `1`) in the T1CON register.
2. Select the timer prescaler ratio using the TCKPS<1:0> bits in the T1CON register.
3. Set the Clock and Gating modes using the TCS and TGATE bits in the T1CON register.
4. Set or clear the TSYNC bit in the T1CON register to select synchronous or asynchronous operation.
5. Load the timer period value into the PR1 register.
6. If interrupts are required, set the interrupt enable bit, T1IE. Use the priority bits, T1IP<2:0>, to set the interrupt priority.

FIGURE 11-1: 16-BIT TIMER1 MODULE BLOCK DIAGRAM



REGISTER 11-1: 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 pin T1CK (on the rising edge) 0 = Internal clock (FCY)
bit 0	**Unimplemented:** Read as ‘0’

12.0 TIMER2/3 FEATURE

Note 1: This data sheet summarizes the features of the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 family 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) of the *"dsPIC33F/PIC24H Family Reference Manual"*, which is available from the Microchip website (www.microchip.com).

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

The Timer2/3 feature has 32-bit timers that can also be configured as two independent 16-bit timers with selectable operating modes.

As a 32-bit timer, the Timer2/3 feature permits operation in three modes:

- Two Independent 16-bit timers (Timer2 and Timer3) with all 16-bit operating modes (except Asynchronous Counter mode)
- Single 32-bit timer (Timer2/3)
- Single 32-bit synchronous counter (Timer2/3)

The Timer2/3 feature also supports:

- Timer gate operation
- Selectable Prescaler Settings
- Timer operation during Idle and Sleep modes
- Interrupt on a 32-bit Period Register Match
- Time Base for Input Capture and Output Compare Modules (Timer2 and Timer3 only)
- ADC1 Event Trigger (Timer2/3 only)

Individually, all eight of the 16-bit timers can function as synchronous timers or counters. They also offer the features listed above, except for the event trigger. The operating modes and enabled features are determined by setting the appropriate bit(s) in the T2CON and T3CON registers. T2CON registers are shown in generic form in Register 12-1. T3CON registers are shown in Register 12-2.

For 32-bit timer/counter operation, Timer2 is the least significant word (lsw), and Timer3 is the most significant word (msw) of the 32-bit timers.

Note: For 32-bit operation, T3CON control bits are ignored. Only T2CON control bit is used for setup and control. Timer2 clock and gate inputs are used for the 32-bit timer modules, but an interrupt is generated with the Timer3 interrupt flags.

12.1 32-Bit Operation

To configure the Timer2/3 feature for 32-bit operation:

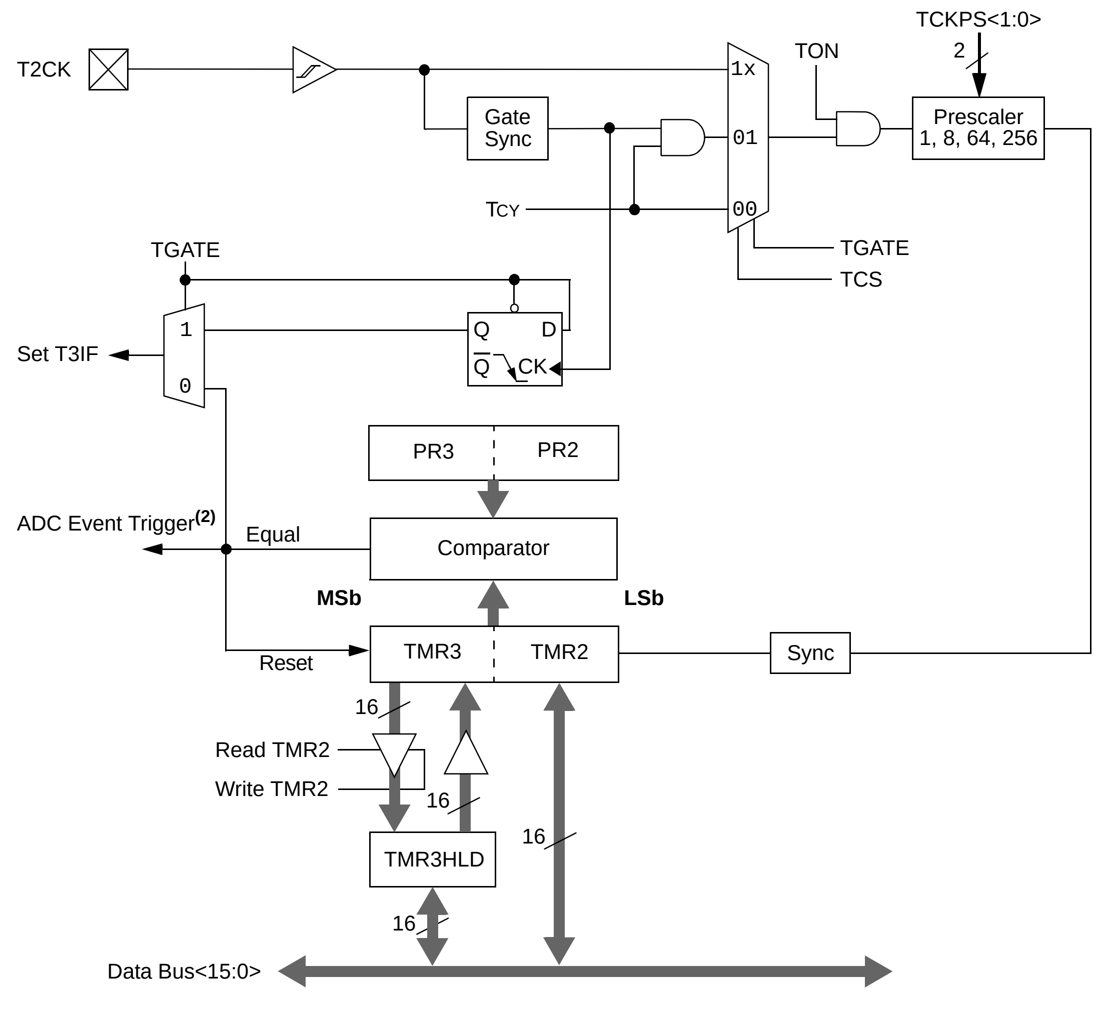
1. Set the corresponding T32 control bit.
2. Select the prescaler ratio for Timer2 using the TCKPS<1:0> bits.
3. Set the Clock and Gating modes using the corresponding TCS and TGATE bits.
4. Load the timer period value. PR3 contains the most significant word of the value, while PR2 contains the least 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.

The timer value at any point is stored in the register pair TMR3:TMR2. TMR3 always contains the most significant word of the count, while TMR2 contains the least significant word.

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.

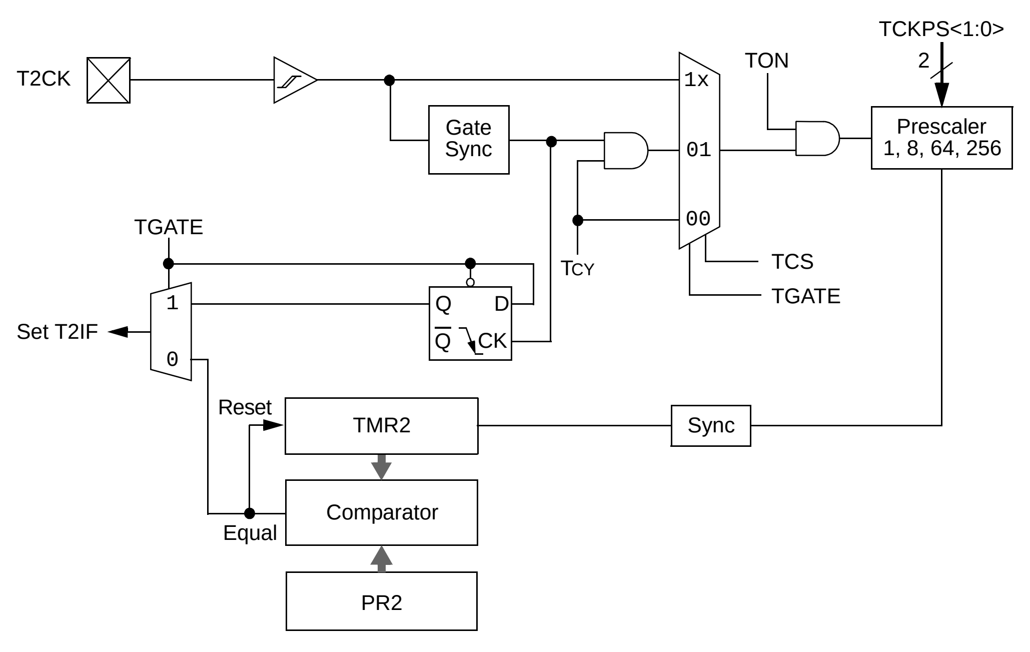
FIGURE 12-1: TIMER2/3 (32-BIT) BLOCK DIAGRAM(1)



Note 1: The 32-bit timer control bit, T32, must be set for 32-bit timer/counter operation. All control bits are respective to the T2CON register.

2: The ADC event trigger is available only on Timer2/3.

FIGURE 12-2: TIMER2 (16-BIT) BLOCK DIAGRAM



REGISTER 12-1: T2CON CONTROL REGISTER

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
TON	—	TSIDL	—	—	—	—	—
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	U-0
—	TGATE	TCKPS<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:** Timer2 On bit

When T32 = 1:
1 = Starts 32-bit Timer2/3
0 = Stops 32-bit Timer2/3

When T32 = 0:
1 = Starts 16-bit Timer2
0 = Stops 16-bit Timer2

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:** Timer2 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>:** Timer2 Input Clock Prescale Select bits

11 = 1:256
10 = 1:64
01 = 1:8
00 = 1:1

bit 3 **T32:** 32-bit Timer Mode Select bit

1 = Timer2 and Timer3 form a single 32-bit timer
0 = Timer2 and Timer3 act as two 16-bit timers

bit 2 **Unimplemented:** Read as ‘0’

bit 1 **TCS:** Timer2 Clock Source Select bit

1 = External clock from pin T2CK (on the rising edge)
0 = Internal clock (FCY)

bit 0 **Unimplemented:** Read as ‘0’

REGISTER 12-2: T3CON CONTROL REGISTER

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
TON[2]	—	TSIDL[1]	—	—	—	—	—
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	U-0
—	TGATE[2]	TCKPS<1:0>[2]		—	—	TCS[2]	—
bit 7							bit 0

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

bit 15 **TON:** Timer3 On bit[2]

`1` = Starts 16-bit Timer3
`0` = Stops 16-bit Timer3

bit 14 **Unimplemented:** Read as ‘`0`’

bit 13 **TSIDL:** Stop in Idle Mode bit[1]

`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:** Timer3 Gated Time Accumulation Enable bit[2]

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>:** Timer3 Input Clock Prescale Select bits[2]

`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:** Timer3 Clock Source Select bit[2]

`1` = External clock from T3CK 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 (T2CON<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 register (T2CON<3>), these bits have no effect.

NOTES:

13.0 INPUT CAPTURE

Note 1: This data sheet summarizes the features of the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 family 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) of the *"dsPIC33F/PIC24H Family Reference Manual"*, which is available from the Microchip website (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 dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 devices support up to eight input capture channels.

The input capture module captures the 16-bit value of the selected Time Base register when an event occurs at the ICx pin. The events that cause a capture event are listed below in three categories:

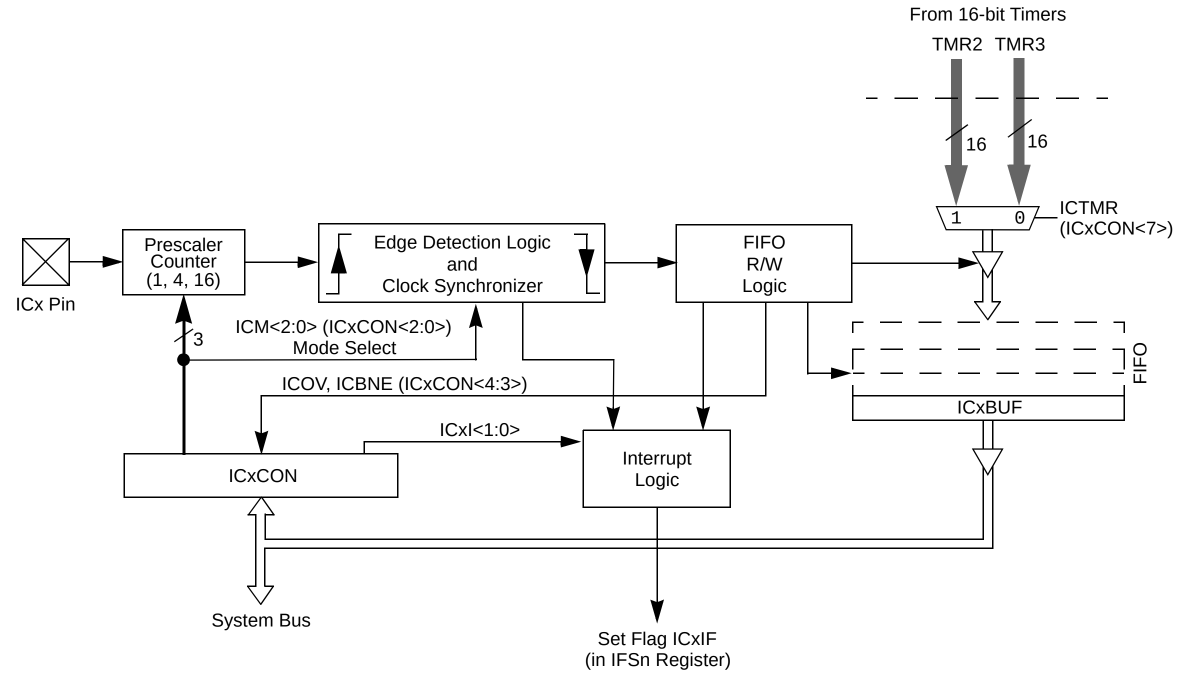
- Simple Capture Event modes:
  - Capture timer value on every falling edge of input at ICx pin
  - Capture timer value on every rising edge of input at ICx pin
- Capture timer value on every edge (rising and falling).
- Prescaler Capture Event modes:
  - Capture timer value on every 4th rising edge 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 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 13-1: INPUT CAPTURE BLOCK DIAGRAM



Note: An 'x' in a signal, register or bit name denotes the number of the capture channel.

13.1 Input Capture Registers

REGISTER 13-1: ICxCON: INPUT CAPTURE x CONTROL REGISTER

U-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
—	—	ICSIDL	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R-0, HC	R-0, HC	R/W-0	R/W-0	R/W-0
ICTMR	ICI<1:0>		ICOV	ICBNE	ICM<2:0>		
bit 7							bit 0

Legend:		HC = Cleared 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 15-14	**Unimplemented:** Read as ‘`0`’
bit 13	**ICSIDL:** Input Capture Module Stop in Idle Control bit `1` = Input capture module will halt in CPU Idle mode `0` = Input capture module will continue to operate in CPU Idle mode
bit 12-8	**Unimplemented:** Read as ‘`0`’
bit 7	**ICTMR:** Input Capture Timer Select bits `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

14.0 OUTPUT COMPARE

Note 1: This data sheet summarizes the features of the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 family 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) of the *"dsPIC33F/PIC24H Family Reference Manual"*, which is available from the Microchip website (www.microchip.com).

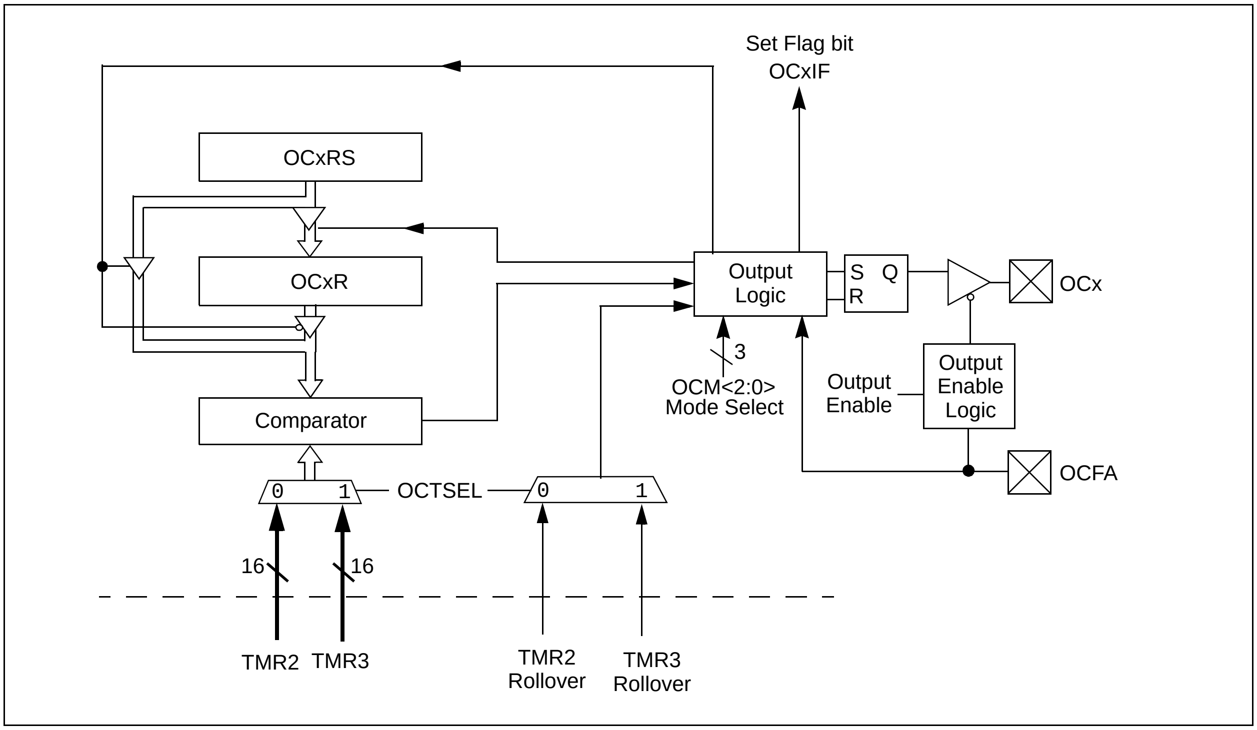
2: Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

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 14-1: OUTPUT COMPARE MODULE BLOCK DIAGRAM



14.1 Output Compare Modes

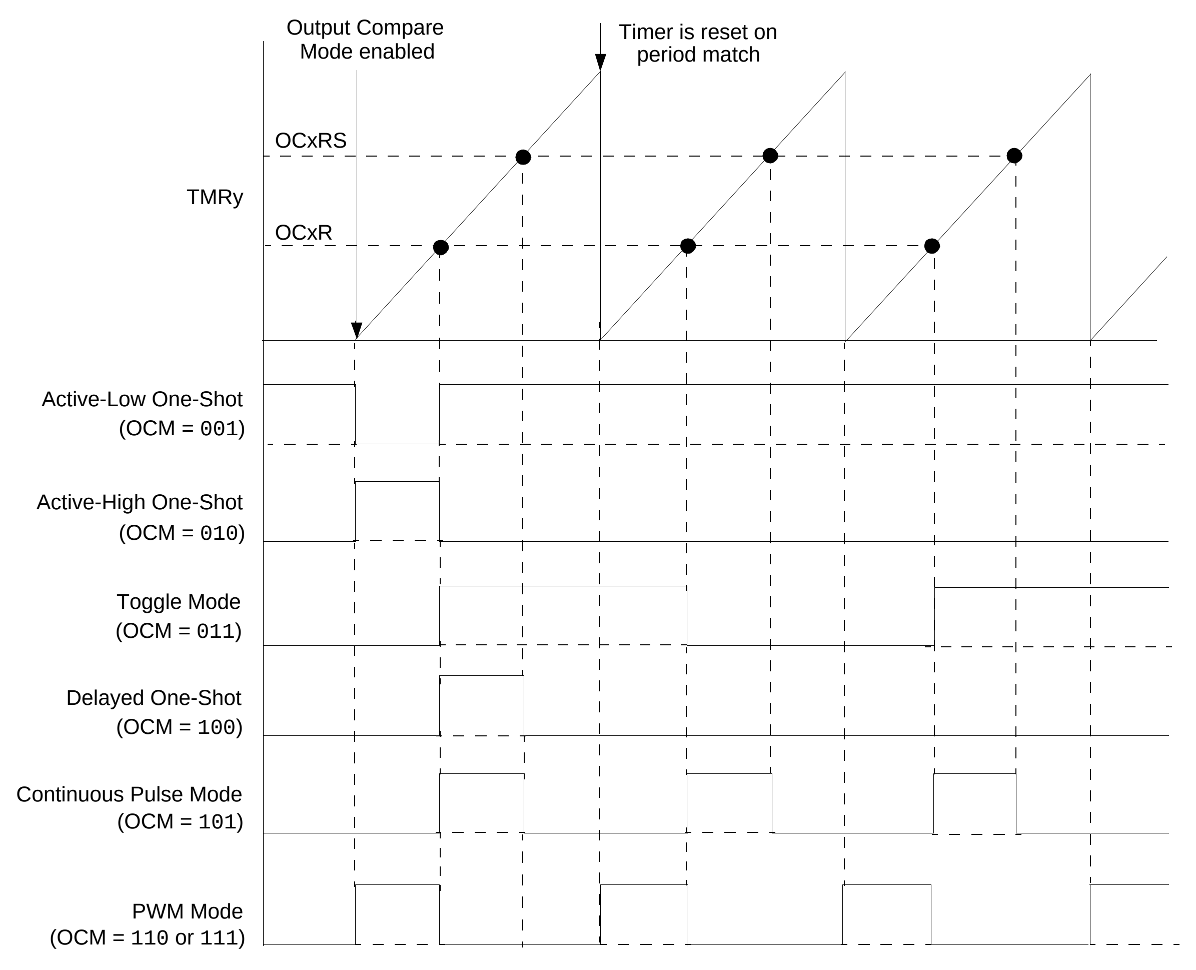
Configure the Output Compare modes by setting the appropriate Output Compare Mode bits (OCM<2:0>) in the Output Compare Control register (OCxCON<2:0>). Table 14-1 lists the different bit settings for the Output Compare modes. Figure 14-2 illustrates the output compare operation for various modes. The 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 14-1: 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 Mode	Current output is maintained	OCx Rising and Falling edge
`100`	Delayed One-Shot	`0`	OCx Falling edge
`101`	Continuous Pulse mode	`0`	OCx Falling edge
`110`	PWM mode without fault protection	`0`, if OCxR is zero `1`, if OCxR is non-zero	No interrupt
`111`	PWM mode with fault protection	`0`, if OCxR is zero `1`, if OCxR is non-zero	OCFA Falling edge for OC1 to OC4

FIGURE 14-2: OUTPUT COMPARE OPERATION



14.2 Output Compare Register

REGISTER 14-1: OCxCON: OUTPUT COMPARE x CONTROL REGISTER

U-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
—	—	OCSIDL	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	R-0 HC	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	OCFLT	OCTSEL	OCM<2:0>		
bit 7							bit 0

Legend:	HC = 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 15-14	**Unimplemented:** Read as ‘0’
bit 13	**OCSIDL:** Stop Output Compare in Idle Mode Control bit 1 = Output Compare x will halt in CPU Idle mode 0 = Output Compare x will continue 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:

15.0 SERIAL PERIPHERAL INTERFACE (SPI)

Note 1: This data sheet summarizes the features of the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 family 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) of the *"dsPIC33F/PIC24H Family Reference Manual"*, which is available from the Microchip website (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 (ADC), etc. The SPI module is compatible with Motorola® SPI and SIOP.

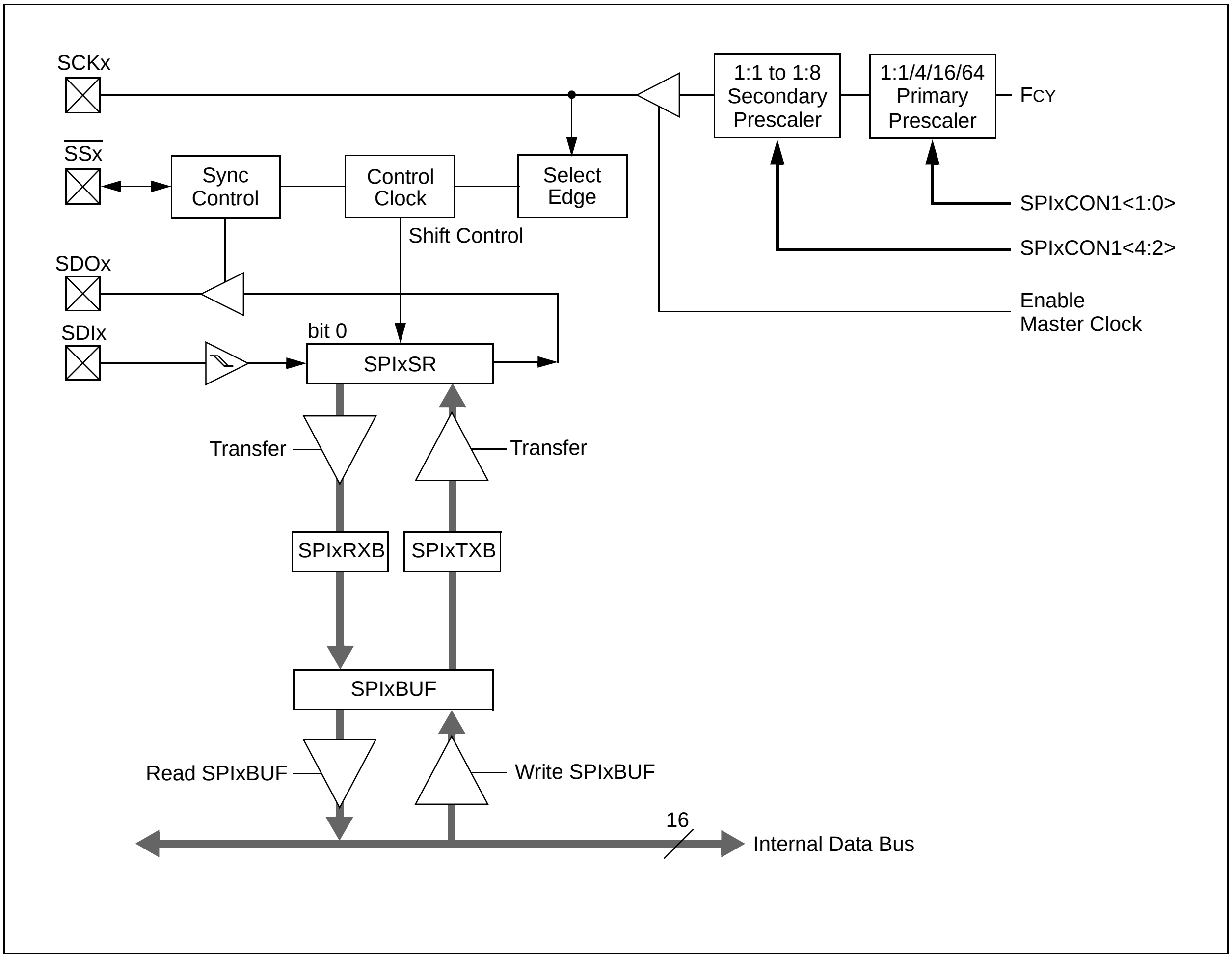
Each SPI module consists of a 16-bit shift register, SPIxSR (where x = 1 or 2), used for shifting data in and out, and a buffer register, SPIxBUF. A control register, SPIxCON, configures the module. Additionally, a status register, SPIxSTAT, indicates status conditions.

The serial interface consists of 4 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 15-1: SPI MODULE BLOCK DIAGRAM



REGISTER 15-1: SPIxSTAT: SPIx STATUS AND CONTROL REGISTER

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
SPIEN	—	SPISIDL	—	—	—	—	—
bit 15							bit 8

U-0	R/C-0	U-0	U-0	U-0	U-0	R-0	R-0
—	SPIROV	—	—	—	—	SPITBF	SPIRBF
bit 7							bit 0

Legend:	C = Clearable 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	**SPIEN:** SPIx Enable bit `1` = Enables module and configures SCKx, SDOx, SDIx and $\overline{\text{SSx}}$ 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 15-2: 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[1]
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SSEN[2]	CKP	MSTEN	SPRE<2:0>[3]			PPRE<1:0>[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 15-13 **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[1]
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)[2]
1 = $\overline{\text{SSx}}$ pin used for Slave mode
0 = $\overline{\text{SSx}}$ 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: This bit must be cleared when FRMEN = 1.

3: Do not set both Primary and Secondary prescalers to a value of 1:1.

REGISTER 15-2: SPIxCON1: SPIx CONTROL REGISTER 1 (CONTINUED)

bit 4-2 **SPRE<2:0>:** Secondary Prescale bits (Master mode)(3)
`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)(3)
`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: This bit must be cleared when FRMEN = `1`.

3: Do not set both Primary and Secondary prescalers to a value of 1:1.

REGISTER 15-3: 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 ($\overline{\text{SSx}}$ 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:

16.0 INTER-INTEGRATED CIRCUIT™ (I^2C™)

Note 1: This data sheet summarizes the features of the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 family 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^2C™)"** (DS70195) of the *"dsPIC33F/PIC24H Family Reference Manual"*, which is available on the Microchip website (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^2C) module provides complete hardware support for both Slave and Multi-Master modes of the I^2C serial communication standard, with a 16-bit interface.

The I^2C module has a 2-pin interface:

- The SCLx pin is clock
- The SDAx pin is data

The I^2C module offers the following key features:

- I^2C interface supporting both Master and Slave modes of operation
- I^2C Slave mode supports 7-bit and 10-bit addressing
- I^2C Master mode supports 7-bit and 10-bit addressing
- I^2C port allows bidirectional transfers between master and slaves
- Serial clock synchronization for I^2C port can be used as a handshake mechanism to suspend and resume serial transfer (SCLREL control)
- I^2C supports multi-master operation, detects bus collision and arbitrates accordingly

16.1 Operating Modes

The hardware fully implements all the master and slave functions of the I^2C Standard and Fast mode specifications, as well as 7-bit and 10-bit addressing.

The I^2C module can operate either as a slave or a master on an I^2C bus.

The following types of I^2C operation are supported:

- I^2C slave operation with 7-bit addressing
- I^2C slave operation with 10-bit addressing
- I^2C 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*".

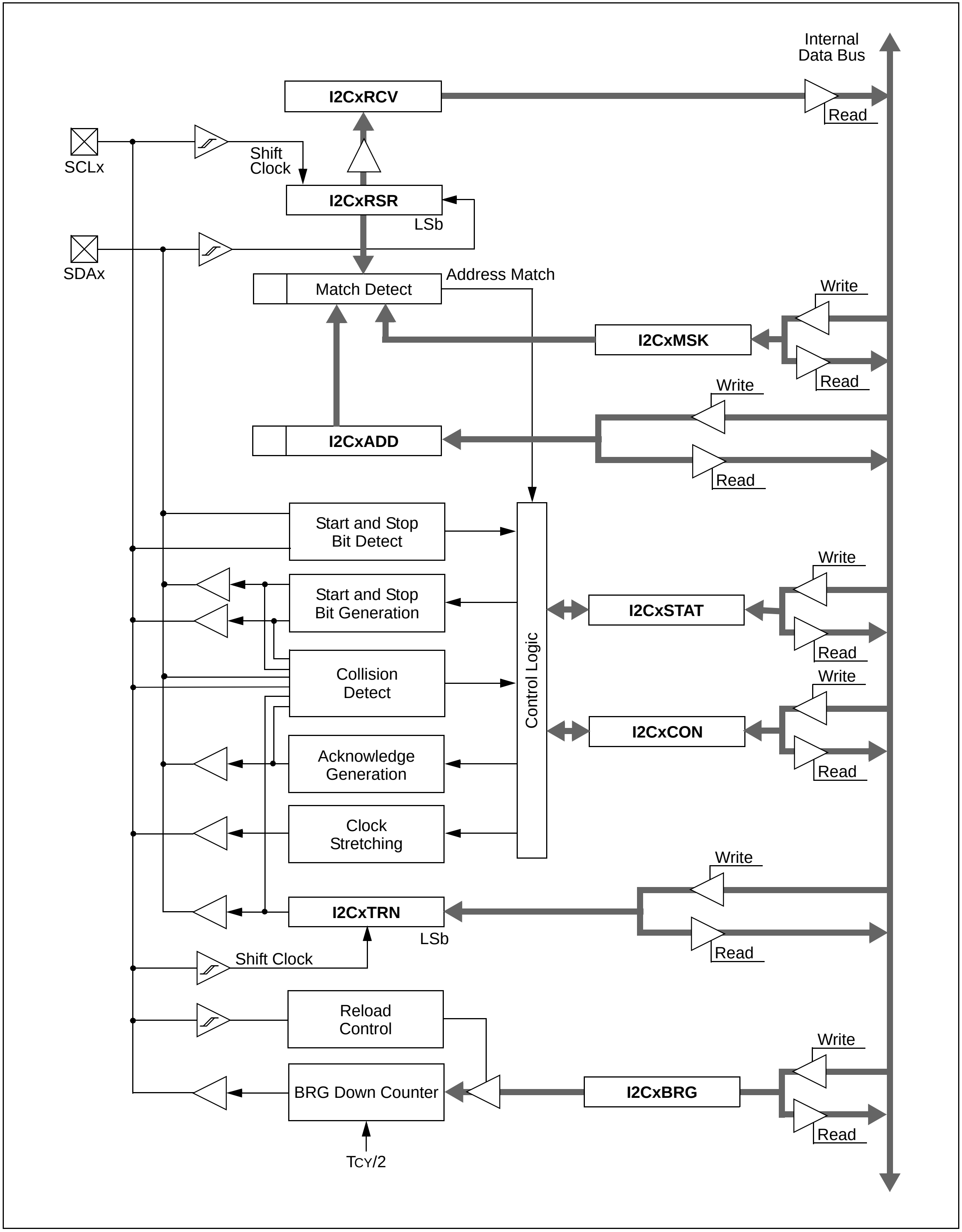
16.2 I^2C Registers

I2CxCON and I2CxSTAT are control and status registers, respectively. The I2CxCON register is readable and writable. The lower six bits of I2CxSTAT are read-only. The remaining bits of the I2CSTAT are read/write.

- I2CxRSR is the shift register used for shifting data.
- 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.
- 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 16-1: I²C™ BLOCK DIAGRAM (x = 1)



REGISTER 16-1: 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

Legend:	U = Unimplemented bit, read as ‘0’		
R = Readable bit	W = Writable bit	HS = Set in hardware	HC = Cleared in hardware
-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^2C 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^2C™ 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^2C 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^2C slave)

Used in conjunction with SCLREL bit.
1 = Enable software or receive clock stretching
0 = Disable software or receive clock stretching

REGISTER 16-1: I2CxCON: I2Cx CONTROL REGISTER (CONTINUED)

bit 5	**ACKDT:** Acknowledge Data bit (when operating as I^{2}C master, applicable during master receive) Value that will be transmitted when the software initiates an Acknowledge sequence. `1` = Send NACK during Acknowledge `0` = Send ACK during Acknowledge
bit 4	**ACKEN:** Acknowledge Sequence Enable bit (when operating as I^{2}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^{2}C master) `1` = Enables Receive mode for I^{2}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^{2}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^{2}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^{2}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 16-2: I2CxSTAT: I2Cx STATUS REGISTER

R-0 HSC	R-0 HSC	U-0	U-0	U-0	R/C-0 HS	R-0 HSC	R-0 HSC
ACKSTAT	TRSTAT	—	—	—	BCL	GCSTAT	ADD10
bit 15							bit 8

R/C-0 HS	R/C-0 HS	R-0 HSC	R/C-0 HSC	R/C-0 HSC	R-0 HSC	R-0 HSC	R-0 HSC
IWCOL	I2COV	D_A	P	S	R_W	RBF	TBF
bit 7							bit 0

Legend:	U = Unimplemented bit, read as ‘0’		
R = Readable bit	W = Writable bit	HS = Set in hardware	HSC = Hardware set/cleared
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15	**ACKSTAT:** Acknowledge Status bit (when operating as I^{2}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^{2}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^{2}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^{2}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 16-2: 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^{2}C slave) `1` = Read – indicates data transfer is output from slave `0` = Write – indicates data transfer is input to slave Hardware set or clear after reception of I^{2}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 16-3: I2CxMSK: I2Cx SLAVE MODE ADDRESS MASK REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	AMSK9	AMSK8
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
AMSK7	AMSK6	AMSK5	AMSK4	AMSK3	AMSK2	AMSK1	AMSK0
bit 7							bit 0

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

bit 15-10	**Unimplemented:** Read as ‘`0`’
bit 9-0	**AMSKx:** Mask for Address bit x Select bit `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:

17.0 UNIVERSAL ASYNCHRONOUS RECEIVER TRANSMITTER (UART)

Note 1: This data sheet summarizes the features of the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 family 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) of the *"dsPIC33F/PIC24H Family Reference Manual"*, which is available on the Microchip website (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 dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 device family. 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 $\overline{\text{UxCTS}}$ and $\overline{\text{UxRTS}}$ pins and also includes an IrDA® encoder and decoder.

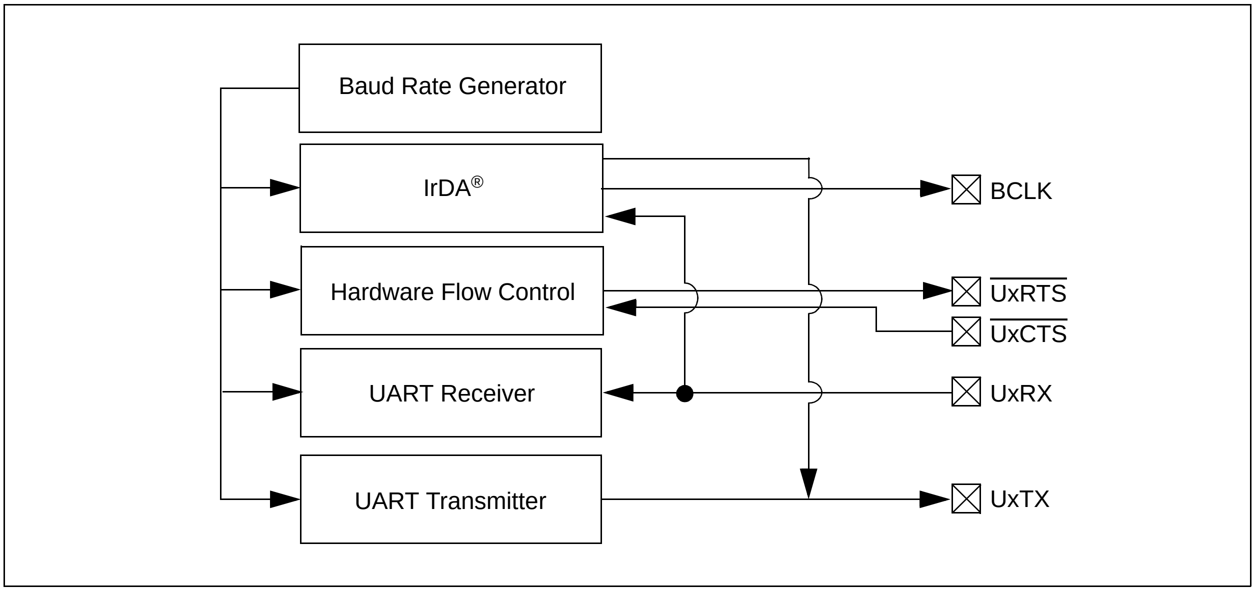
The primary features of the UART module are:

- 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 $\overline{\text{UxCTS}}$ and $\overline{\text{UxRTS}}$ pins
- Fully Integrated Baud Rate Generator with 16-bit prescaler
- Baud rates ranging from 10 Mbps to 38 bps at 40 MIPS
- 4-deep first-in-first-out (FIFO) Transmit Data Buffer
- 4-Deep FIFO Receive Data Buffer
- Parity, framing and buffer overrun error detection
- Support for 9-bit mode with Address Detect (9th bit = 1)
- Transmit and Receive interrupts
- A separate interrupt for all UART error conditions
- Loopback mode for diagnostic support
- Support for Sync and Break characters
- Support for automatic baud rate detection
- IrDA® encoder and decoder logic
- 16x baud clock output for IrDA® support

A simplified block diagram of the UART module is shown in Figure 17-1. The UART module consists of these key hardware elements:

- Baud Rate Generator
- Asynchronous Transmitter
- Asynchronous Receiver

FIGURE 17-1: UART SIMPLIFIED BLOCK DIAGRAM



REGISTER 17-1: UxMODE: UARTx MODE REGISTER

R/W-0	U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
UARTEN[1]	—	USIDL	IREN[2]	RTSMD	—	UEN<1:0>	
bit 15							bit 8

R/W-0 HC	R/W-0	R/W-0, HC	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
WAKE	LPBACK	ABAUD	URXINV	BRGH	PDSEL<1:0>		STSEL
bit 7							bit 0

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

bit 15 **UARTEN:** UARTx Enable bit[1]
- `1` = UARTx is enabled; all UARTx pins are controlled by UARTx as defined by UEN<1:0>
- `0` = UARTx is disabled; all UARTx pins are controlled by port latches; UARTx power consumption minimal

bit 14 **Unimplemented:** Read as ‘`0`’

bit 13 **USIDL:** Stop in Idle Mode bit
- `1` = Discontinue module operation when device enters Idle mode
- `0` = Continue module operation in Idle mode

bit 12 **IREN:** IrDA® Encoder and Decoder Enable bit[2]
- `1` = IrDA® encoder and decoder 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` = UxRTS pin in Flow Control mode

bit 10 **Unimplemented:** Read as ‘`0`’

bit 9-8 **UEN<1:0>:** UARTx Enable bits
- `11` = UxTX, UxRX and BCLK pins are enabled and used; $\overline{\text{UxCTS}}$ 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; $\overline{\text{UxCTS}}$ and $\overline{\text{UxRTS}}$/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

bit 4 **URXINV:** Receive Polarity Inversion bit
- `1` = UxRX Idle state is ‘`0`’
- `0` = UxRX Idle state is ‘`1`’

Note 1: Refer to **Section 17. “UART”** (DS70188) in the *“dsPIC33F/PIC24H Family Reference Manual”* for information on enabling the UART module for receive or transmit operation.

2: This feature is only available for the 16x BRG mode (BRGH = `0`).

REGISTER 17-1: UxMODE: UARTx MODE REGISTER (CONTINUED)

bit 3	**BRGH:** High Baud Rate Enable bit `1` = BRG generates 4 clocks per bit period (4x baud clock, High-Speed mode) `0` = BRG generates 16 clocks per bit period (16x baud clock, Standard mode)
bit 2-1	**PDSEL<1:0>:** Parity and Data Selection bits `11` = 9-bit data, no parity `10` = 8-bit data, odd parity `01` = 8-bit data, even parity `00` = 8-bit data, no parity
bit 0	**STSEL:** Stop Bit Selection bit `1` = Two Stop 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.

2: This feature is only available for the 16x BRG mode (BRGH = `0`).

REGISTER 17-2: 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[1]	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

Legend:	HC = Hardware cleared	C = Clear 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 15,13 **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® 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[1]

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.

REGISTER 17-2: 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 (read/clear 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.

NOTES:

18.0 10-BIT/12-BIT ANALOG-TO-DIGITAL CONVERTER (ADC)

Note 1:	This data sheet summarizes the features of the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 16. "Analog-to-Digital Converter (ADC)"** (DS70183) of the *"dsPIC33F/PIC24H Family Reference Manual"*, which is available on the Microchip website (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 dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 devices have up to 13 Analog-to-Digital Conversion (ADC) module input channels.

The AD12B bit (AD1CON1<10>) allows each of the ADC modules to be configured as either a 10-bit, 4-sample-and-hold ADC (default configuration) or a 12-bit, 1-sample-and-hold ADC.

Note:	The ADC module must be disabled before the AD12B bit can be modified.

18.1 Key Features

The 10-bit ADC configuration has the following key features:

- Successive Approximation (SAR) conversion
- Conversion speeds of up to 1.1 Msps
- Up to 13 analog input pins
- External voltage reference input pins
- Simultaneous sampling of up to four analog input pins
- Automatic Channel Scan mode
- Selectable conversion trigger source
- Selectable Buffer Fill modes
- Four result alignment options (signed/unsigned, fractional/integer)
- Operation during CPU Sleep and Idle modes
- 16-word conversion result buffer

The 12-bit ADC configuration supports all the above features, except:

- In the 12-bit configuration, conversion speeds of up to 500 ksps are supported.
- There is only one sample and hold (S&H) amplifier in the 12-bit configuration, so simultaneous sampling of multiple channels is not supported.

Depending on the particular device pinout, the ADC can have up to 13 analog input pins, designated AN0 through AN12. In addition, there are two analog input pins for external voltage reference connections. These voltage reference inputs can be shared with other analog input pins.

The actual number of analog input pins and external voltage reference input configuration will depend on the specific device. Refer to the specific device data sheet for further details.

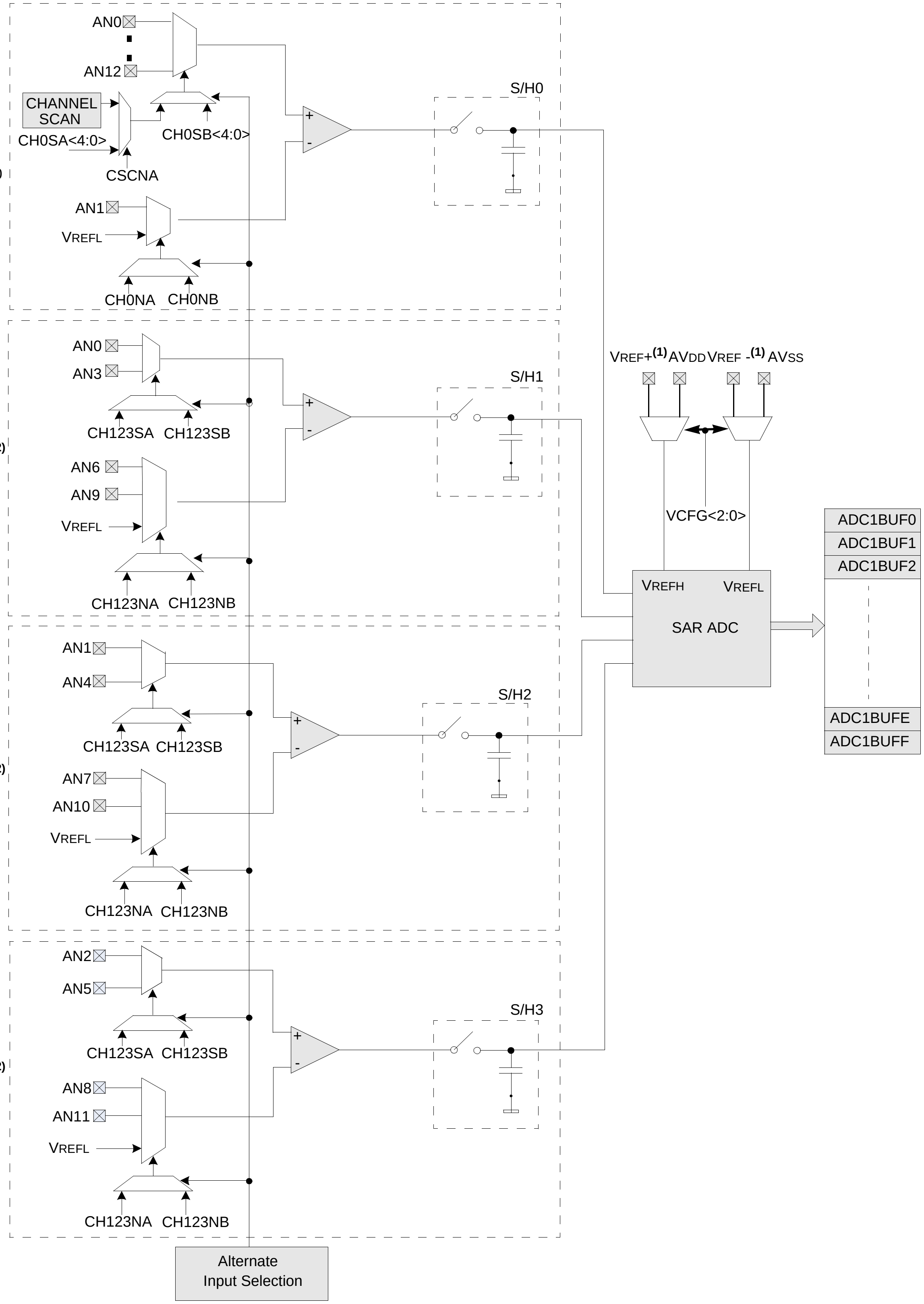
A block diagram of the ADC for the dsPIC33FJ16GP304 and dsPIC33FJ32GP204 devices is shown in Figure 18-1. A block diagram of the ADC for the dsPIC33FJ32GP202 device is shown in Figure 18-2.

18.2 ADC Initialization

To configure the ADC module:

1. Select port pins as analog inputs (AD1PCFGH<15:0> or AD1PCFGL<15:0>).
2. Select voltage reference source to match expected range on analog inputs (AD1CON2<15:13>).
3. Select the analog conversion clock to match desired data rate with processor clock (AD1CON3<7:0>).
4. Determine how many sample-and-hold channels will be used (AD1CON2<9:8> and AD1PCFGH<15:0> or AD1PCFGL<15:0>).
5. Select the appropriate sample/conversion sequence (AD1CON1<7:5> and AD1CON3<12:8>).
6. Select the way conversion results are presented in the buffer (AD1CON1<9:8>).
    e) Turn on the ADC module (AD1CON1<15>).
7. Configure ADC interrupt (if required):
    a) Clear the AD1IF bit.
    b) Select ADC interrupt priority.

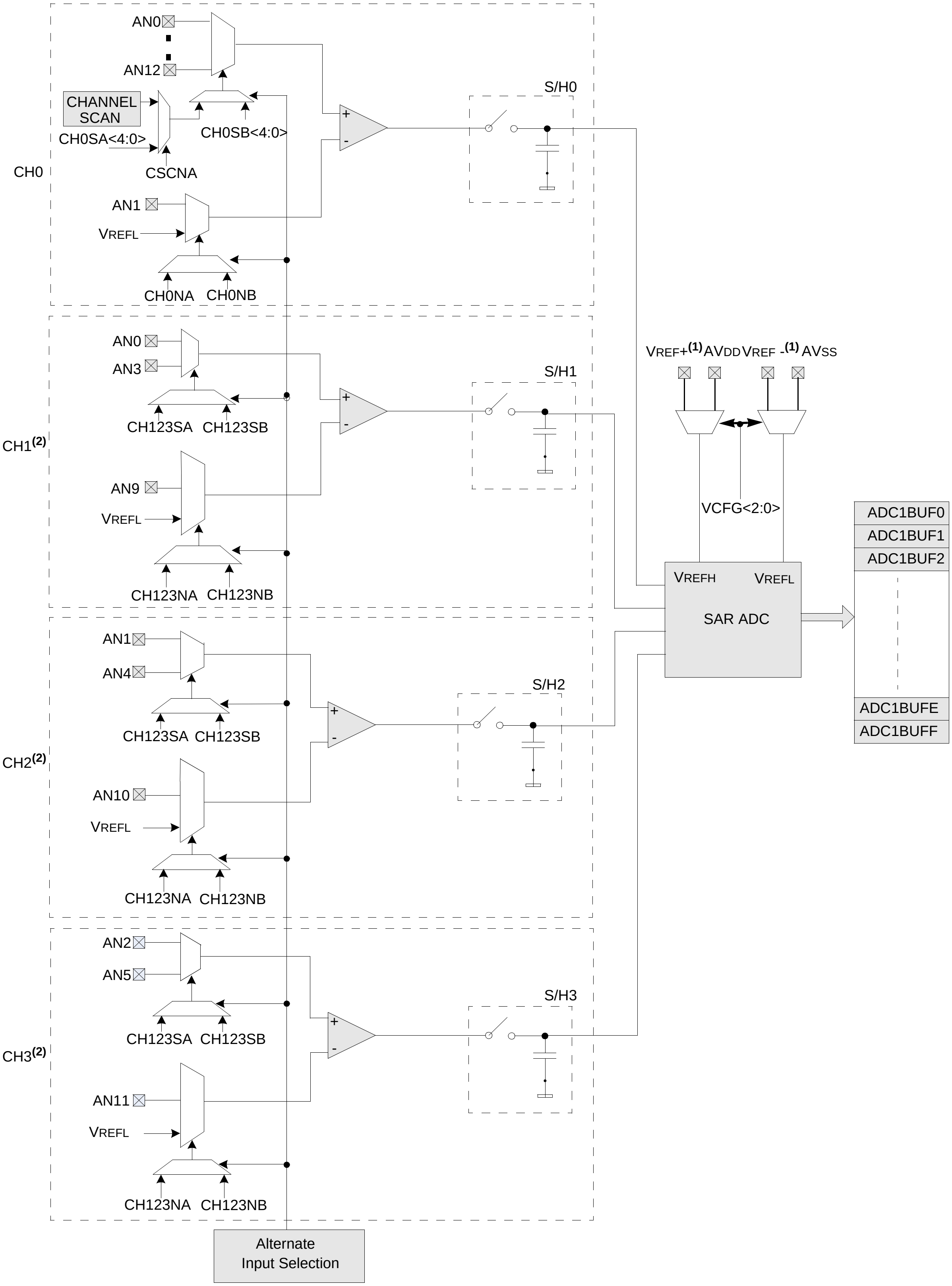
FIGURE 18-1: ADC1 MODULE BLOCK DIAGRAM FOR dsPIC33FJ16GP304 AND dsPIC33FJ32GP204 DEVICES



Note 1: VREF+, VREF- inputs can be multiplexed with other analog inputs.

2: Channels 1, 2 and 3 are not applicable for the 12-bit mode of operation.

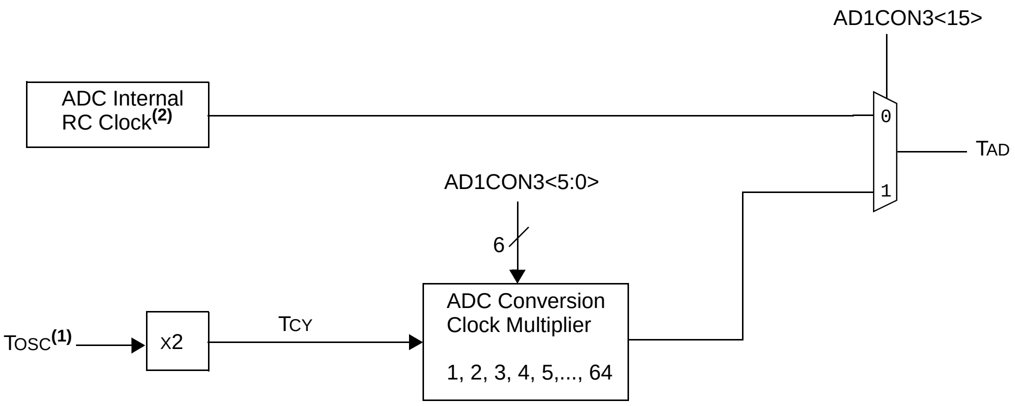
FIGURE 18-2: ADC1 MODULE BLOCK DIAGRAM FOR dsPIC33FJ32GP202 DEVICES



Note 1: VREF+, VREF- inputs can be multiplexed with other analog inputs.

2: Channels 1, 2 and 3 are not applicable for the 12-bit mode of operation.

FIGURE 18-3: ADC CONVERSION CLOCK PERIOD BLOCK DIAGRAM



Note 1: Refer to Figure 8-2 for the derivation of FOSC when the PLL is enabled. If the PLL is not used, FOSC is equal to the clock frequency. TOSC = 1/FOSC.

2: See the ADC Electrical Characteristics for the exact RC clock value.

REGISTER 18-1: AD1CON1: ADC1 CONTROL REGISTER 1

R/W-0	U-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0
ADON	—	ADSIDL	—	—	AD12B	FORM<1:0>	
bit 15							bit 8

R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0 HC,HS	R/C-0 HC, HS
SSRC<2:0>			—	SIMSAM	ASAM	SAMP	DONE
bit 7							bit 0

Legend:	HC = Cleared by hardware	HS = Set by hardware	C = Clear 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 **ADON:** ADC Operating Mode bit

`1` = ADC module is operating
`0` = ADC is off

bit 14 **Unimplemented:** Read as ‘`0`’

bit 13 **ADSIDL:** Stop in Idle Mode bit

`1` = Discontinue module operation when device enters Idle mode
`0` = Continue module operation in Idle mode

bit 12-11 **Unimplemented:** Read as ‘`0`’

bit 10 **AD12B:** 10-bit or 12-bit Operation Mode bit

`1` = 12-bit, 1-channel ADC operation
`0` = 10-bit, 4-channel ADC operation

bit 9-8 **FORM<1:0>:** Data Output Format bits

For 10-bit operation:
`11` = Signed fractional (DOUT = `sddd dddd dd00 0000`, where s = .NOT.d<9>)
`10` = Fractional (DOUT = `dddd dddd dd00 0000`)
`01` = Signed integer (DOUT = `ssss sssd dddd dddd`, where s = .NOT.d<9>)
`00` = Integer (DOUT = `0000 00dd dddd dddd`)

For 12-bit operation:
`11` = Signed fractional (DOUT = `sddd dddd dddd 0000`, where s = .NOT.d<11>)
`10` = Fractional (DOUT = `dddd dddd dddd 0000`)
`01` = Signed Integer (DOUT = `ssss sddd dddd dddd`, where s = .NOT.d<11>)
`00` = Integer (DOUT = `0000 dddd dddd dddd`)

bit 7-5 **SSRC<2:0>:** Sample Clock Source Select bits

`111` = Internal counter ends sampling and starts conversion (auto-convert)
`110` = Reserved
`101` = Motor Control PWM2 interval ends sampling and starts conversion
`100` = Reserved
`011` = Motor Control PWM1 interval ends sampling and starts conversion
`010` = GP timer 3 compare ends sampling and starts conversion
`001` = Active transition on INT0 pin ends sampling and starts conversion
`000` = Clearing sample bit ends sampling and starts conversion

bit 4 **Unimplemented:** Read as ‘`0`’

bit 3 **SIMSAM:** Simultaneous Sample Select bit (applicable only when CHPS<1:0> = `01` or `1x`)

When AD12B = 1, SIMSAM is: U-0, Unimplemented, Read as ‘`0`’
`1` = Samples CH0, CH1, CH2, CH3 simultaneously (when CHPS<1:0> = `1x`); or
Samples CH0 and CH1 simultaneously (when CHPS<1:0> = `01`)
`0` = Samples multiple channels individually in sequence

REGISTER 18-1: AD1CON1: ADC1 CONTROL REGISTER 1 (CONTINUED)

bit 2	**ASAM:** ADC Sample Auto-Start bit `1` = Sampling begins immediately after last conversion. SAMP bit is auto-set `0` = Sampling begins when SAMP bit is set
bit 1	**SAMP:** ADC Sample Enable bit `1` = ADC sample-and-hold amplifiers are sampling `0` = ADC sample-and-hold amplifiers are holding If ASAM = `0`, software can write ‘`1`’ to begin sampling. Automatically set by hardware if ASAM = `1`. If SSRC = `000`, software can write ‘`0`’ to end sampling and start conversion. If SSRC ≠ `000`, automatically cleared by hardware to end sampling and start conversion.
bit 0	**DONE:** ADC Conversion Status bit `1` = ADC conversion cycle is completed `0` = ADC conversion not started or in progress Automatically set by hardware when ADC conversion is complete. Software can write ‘`0`’ to clear DONE status (software not allowed to write ‘`1`’). Clearing this bit will NOT affect any operation in progress. Automatically cleared by hardware at start of a new conversion.

REGISTER 18-2: AD1CON2: ADC1 CONTROL REGISTER 2

R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0
VCFG<2:0>			—	—	CSCNA	CHPS<1:0>	
bit 15							bit 8

R-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BUFS	—	SMPI<3:0>				BUFM	ALTS
bit 7							bit 0

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

bit 15-13 **VCFG<2:0>:** Converter Voltage Reference Configuration bits

	ADREF+	**ADREF-**
`000`	AVDD	AVSS
`001`	External VREF+	AVSS
`010`	AVDD	External VREF-
`011`	External VREF+	External VREF-
`1xx`	AVDD	Avss

bit 12-11 **Unimplemented:** Read as ‘0’

bit 10 **CSCNA:** Scan Input Selections for CH0+ during Sample A bit

1 = Scan inputs
0 = Do not scan inputs

bit 9-8 **CHPS<1:0>:** Select Channels Utilized bits

When AD12B = 1, CHPS<1:0> is: U-0, Unimplemented, Read as ‘0’
1x = Converts CH0, CH1, CH2 and CH3
01 = Converts CH0 and CH1
00 = Converts CH0

bit 7 **BUFS:** Buffer Fill Status bit (valid only when BUFM = 1)

1 = ADC is currently filling second half of buffer, user application should access data in the first half
0 = ADC is currently filling first half of buffer, user application should access data in the second half

bit 6 **Unimplemented:** Read as ‘0’

bit 5-2 **SMPI<3:0>:** Sample/Convert Sequences Per Interrupt Selection bits

1111 = Interrupts at the completion of conversion for each 16th sample/convert sequence
1110 = Interrupts at the completion of conversion for each 15th sample/convert sequence
•
•
•
0001 = Interrupts at the completion of conversion for each 2nd sample/convert sequence
0000 = Interrupts at the completion of conversion for each sample/convert sequence

bit 1 **BUFM:** Buffer Fill Mode Select bit

1 = Starts filling first half of buffer on first interrupt and the second half of buffer on next interrupt
0 = Always starts filling buffer from the beginning

bit 0 **ALTS:** Alternate Input Sample Mode Select bit

1 = Uses channel input selects for Sample A on first sample and Sample B on next sample
0 = Always uses channel input selects for Sample A

REGISTER 18-3: AD1CON3: ADC1 CONTROL REGISTER 3

R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADRC	—	—	SAMC<4:0>(1)				
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADCS<7:0>(2)							
bit 7							bit 0

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

bit 15 **ADRC:** ADC Conversion Clock Source bit

`1` = ADC internal RC clock
`0` = Clock derived from system clock

bit 14-13 **Unimplemented:** Read as ‘`0`’

bit 12-8 **SAMC<4:0>:** Auto Sample Time bits(1)

`11111` = 31 TAD
•
•
•
`00001` = 1 TAD
`00000` = 0 TAD

bit 7-0 **ADCS<7:0>:** ADC Conversion Clock Select bits(2)

`11111111` = Reserved
•
•
•
•
`01000000` = Reserved
`00111111` = TCY · (ADCS<7:0> + 1) = 64 · TCY = TAD
•
•
•
`00000010` = TCY · (ADCS<7:0> + 1) = 3 · TCY = TAD
`00000001` = TCY · (ADCS<7:0> + 1) = 2 · TCY = TAD
`00000000` = TCY · (ADCS<7:0> + 1) = 1 · TCY = TAD

Note 1: This bit only used if AD1CON1<7:5> (SSRC<2:0>) = `111`.
2: This bit is not used if AD1CON3<15> (ADRC) = `1`.

REGISTER 18-4: AD1CHS123: ADC1 INPUT CHANNEL 1, 2, 3 SELECT REGISTER

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	CH123NB<1:0>		CH123SB
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	CH123NA<1:0>		CH123SA
bit 7							bit 0

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

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

bit 10-9 **CH123NB<1:0>:** Channel 1, 2, 3 Negative Input Select for Sample B bits

dsPIC33FJ32GP202 devices only:
If AD12B = 1:
11 = Reserved
10 = Reserved
01 = Reserved
00 = Reserved

If AD12B = 0:
11 = CH1 negative input is AN9, CH2 negative input is AN10, CH3 negative input is AN11
10 = Reserved
01 = CH1, CH2, CH3 negative input is VREF-
00 = CH1, CH2, CH3 negative input is VREF-

dsPIC33FJ32GP204 and dsPIC33FJ16GP304 devices only:
If AD12B = 1:
11 = Reserved
10 = Reserved
01 = Reserved
00 = Reserved

If AD12B = 0:
11 = CH1 negative input is AN9, CH2 negative input is AN10, CH3 negative input is AN11
10 = CH1 negative input is AN6, CH2 negative input is AN7, CH3 negative input is AN8
01 = CH1, CH2, CH3 negative input is VREF-
00 = CH1, CH2, CH3 negative input is VREF-

bit 8 **CH123SB:** Channel 1, 2, 3 Positive Input Select for Sample B bit

If AD12B = 1:
1 = Reserved
0 = Reserved

If AD12B = 0:
1 = CH1 positive input is AN3, CH2 positive input is AN4, CH3 positive input is AN5
0 = CH1 positive input is AN0, CH2 positive input is AN1, CH3 positive input is AN2

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

REGISTER 18-4: AD1CHS123: ADC1 INPUT CHANNEL 1, 2, 3 SELECT REGISTER (CONTINUED)

bit 2-1 **CH123NA<1:0>:** Channel 1, 2, 3 Negative Input Select for Sample A bits

dsPIC33FJ32GP202 devices only:

If AD12B = `1`:
`11` = Reserved
`10` = Reserved
`01` = Reserved
`00` = Reserved

If AD12B = `0`:
`11` = CH1 negative input is AN9, CH2 negative input is AN10, CH3 negative input is AN11
`10` = Reserved
`01` = CH1, CH2, CH3 negative input is VREF-
`00` = CH1, CH2, CH3 negative input is VREF-

dsPIC33FJ32GP204 and dsPIC33FJ16GP304 devices only:

If AD12B = `1`:
`11` = Reserved
`10` = Reserved
`01` = Reserved
`00` = Reserved

If AD12B = `0`:
`11` = CH1 negative input is AN9, CH2 negative input is AN10, CH3 negative input is AN11
`10` = CH1 negative input is AN6, CH2 negative input is AN7, CH3 negative input is AN8
`01` = CH1, CH2, CH3 negative input is VREF-
`00` = CH1, CH2, CH3 negative input is VREF-

bit 0 **CH123SA:** Channel 1, 2, 3 Positive Input Select for Sample A bit

If AD12B = `1`:
`1` = Reserved
`0` = Reserved

If AD12B = `0`:
`1` = CH1 positive input is AN3, CH2 positive input is AN4, CH3 positive input is AN5
`0` = CH1 positive input is AN0, CH2 positive input is AN1, CH3 positive input is AN2

REGISTER 18-5: AD1CHS0: ADC1 INPUT CHANNEL 0 SELECT REGISTER

R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CH0NB	—	—	CH0SB<4:0>				
bit 15							bit 8

R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CH0NA	—	—	CH0SA<4:0>				
bit 7							bit 0

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

bit 15 **CH0NB:** Channel 0 Negative Input Select for Sample B bit

`1` = Channel 0 negative input is AN1
`0` = Channel 0 negative input is VREF-

bit 14-13 **Unimplemented:** Read as ‘`0`’

bit 12-8 **CH0SB<4:0>:** Channel 0 Positive Input Select for Sample B bits

dsPIC33FJ32GP204 and dsPIC33FJ16GP304 devices only:
`01100` = Channel 0 positive input is AN12
•
•
•
`00010` = Channel 0 positive input is AN2
`00001` = Channel 0 positive input is AN1
`00000` = Channel 0 positive input is AN0

dsPIC33FJ32GP202 devices only:
`01100` = Channel 0 positive input is AN12
•
•
•
`01000` = Reserved
`00111` = Reserved
`00110` = Reserved
•
•
•
`00010` = Channel 0 positive input is AN2
`00001` = Channel 0 positive input is AN1
`00000` = Channel 0 positive input is AN0

bit 7 **CH0NA:** Channel 0 Negative Input Select for Sample A bit

`1` = Channel 0 negative input is AN1
`0` = Channel 0 negative input is VREF-

bit 6-5 **Unimplemented:** Read as ‘`0`’

REGISTER 18-5: AD1CHS0: ADC1 INPUT CHANNEL 0 SELECT REGISTER (CONTINUED)

bit 4-0	**CH0SA<4:0>:** Channel 0 Positive Input Select for Sample A bits **dsPIC33FJ32GP204 and dsPIC33FJ16GP304 devices only:** `01100` = Channel 0 positive input is AN12 • • • `00010` = Channel 0 positive input is AN2 `00001` = Channel 0 positive input is AN1 `00000` = Channel 0 positive input is AN0 **dsPIC33FJ32GP202 devices only:** `01100` = Channel 0 positive input is AN12 • • • `01000` = Reserved `00111` = Reserved `00110` = Reserved • • • `00010` = Channel 0 positive input is AN2 `00001` = Channel 0 positive input is AN1 `00000` = Channel 0 positive input is AN0

REGISTER 18-6: AD1CSSL: ADC1 INPUT SCAN SELECT REGISTER LOW[1,2]

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	CSS12	CSS11	CSS10	CSS9	CSS8
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
CSS7	CSS6	CSS5	CSS4	CSS3	CSS2	CSS1	CSS0
bit 7							bit 0

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

bit 15-13	**Unimplemented:** Read as ‘`0`’
bit 12-0	**CSS<12:0>:** ADC Input Scan Selection bits `1` = Select ANx for input scan `0` = Skip ANx for input scan

Note 1: On devices without 13 analog inputs, all AD1CSSL bits can be selected by the user application. However, inputs selected for scan without a corresponding input on device converts VREFL.

2: CSSx = ANx, where x = 0 through 12.

REGISTER 18-7: AD1PCFGL: ADC1 PORT CONFIGURATION REGISTER LOW[1,2,3]

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	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 15-13	**Unimplemented:** Read as ‘`0`’
bit 12-0	**PCFG<12:0>:** ADC Port Configuration Control bits `1` = Port pin in Digital mode, port read input enabled, ADC input multiplexer connected to AVSS `0` = Port pin in Analog mode, port read input disabled, ADC samples pin voltage

Note 1: On devices without 13 analog inputs, all PCFG bits are R/W by user software. However, the PCFG bits are ignored on ports without a corresponding input on device.

2: PCFGx = ANx, where x = 0 through 12.

3: The PCFGx bits have no effect if the ADC module is disabled by setting ADxMD bit in the PMDx Register. In this case, all port pins multiplexed with ANx will be in Digital mode.

NOTES:

19.0 SPECIAL FEATURES

> **Note:** This data sheet summarizes the features of the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 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) for the latest dsPIC33F/PIC24H Family Reference Manual sections.

dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 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

19.1 Configuration Bits

dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 devices provide nonvolatile memory implementation for device configuration bits. Refer to **Section 25. "Device Configuration"** (DS70194) of the *"dsPIC33F/PIC24H Family Reference Manual"*, for more information on this implementation.

The Configuration bits can be programmed (read as '0'), or left unprogrammed (read as '1'), to select various device configurations. These bits are mapped starting at program memory location 0xF80000.

The Device Configuration register map is shown in Table 19-1.

The individual Configuration bit descriptions for the Configuration registers are shown in Table 19-2.

Note that address 0xF80000 is beyond the user program memory space. It belongs to the configuration memory space (0x800000-0xFFFFFF), which can only be accessed using table reads and table writes.

TABLE 19-1: 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>		IOL1WAY	—	—	OSCIOFNC	POSCMD<1:0>	
0xF8000A	FWDT	FWDTEN	WINDIS	—	WDTPRE	WDTPOST<3:0>			
0xF8000C	FPOR	Reserved(2)			ALTI2C	—	FPWRT<2:0>		
0xF8000E	FICD	Reserved(1)		JTAGEN	—	—	—	ICS<1:0>	
0xF80010	FUID0	User Unit ID Byte 0							
0xF80012	FUID1	User Unit ID Byte 1							
0xF80014	FUID2	User Unit ID Byte 2							
0xF80016	FUID3	User Unit ID Byte 3							

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 and always read as '1'.

TABLE 19-2: dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 CONFIGURATION BITS DESCRIPTION

Bit Field	Register	RTSP Effect	Description
BWRP	FBS	Immediate	Boot Segment Program Flash Write Protection `1` = Boot segment may be written `0` = Boot segment is write-protected
BSS<2:0>	FBS	Immediate	**dsPIC33FJ32GP202 and dsPIC33FJ32GP204 Devices Only** Boot Segment Program Flash Code Protection Size `X11` = No Boot program Flash segment Boot space is 768 Instruction Words (except interrupt vectors) `110` = Standard security; boot program Flash segment ends at 0x0007FE `010` = High security; boot program Flash segment ends at 0x0007FE Boot space is 3840 Instruction Words (except interrupt vectors) `101` = Standard security; boot program Flash segment, ends at 0x001FFE `001` = High security; boot program Flash segment ends at 0x001FFE Boot space is 7936 Instruction Words (except interrupt vectors) `100` = Standard security; boot program Flash segment ends at 0x003FFE `000` = High security; boot program Flash segment ends at 0x003FFE
BSS<2:0>	FBS	Immediate	**dsPIC33FJ16GP304 Devices Only** Boot Segment Program Flash Code Protection Size `X11` = No Boot program Flash segment Boot space is 768 Instruction Words (except interrupt vectors) `110` = Standard security; boot program Flash segment ends at 0x0007FE `010` = High security; boot program Flash segment ends at 0x0007FE Boot space is 3840 Instruction Words (except interrupt vectors) `101` = Standard security; boot program Flash segment, ends at 0x001FFE `001` = High security; boot program Flash segment ends at 0x001FFE Boot space is 5376 Instruction Words (except interrupt vectors) `100` = Standard security; boot program Flash segment ends at 0x002BFE `000` = High security; boot program Flash segment ends at 0x002BFE
GSS<1:0>	FGS	Immediate	General Segment Code-Protect bit `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

TABLE 19-2: dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 CONFIGURATION BITS DESCRIPTION (CONTINUED)

Bit Field	Register	RTSP Effect	Description
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
IOL1WAY	FOSC	Immediate	Peripheral Pin Select Configuration `1` = Allow only one re-configuration `0` = Allow multiple re-configurations
OSCIOFNC	FOSC	Immediate	OSC2 Pin Function bit (except in XT and HS modes) `1` = OSC2 is clock output `0` = OSC2 is general purpose digital I/O pin
POSCMD<1:0>	FOSC	Immediate	Primary Oscillator Mode Select bits `11` = Primary oscillator disabled `10` = HS Crystal Oscillator mode `01` = XT Crystal Oscillator mode `00` = EC (External Clock) mode
FWDTEN	FWDT	Immediate	Watchdog Timer Enable bit `1` = Watchdog Timer always enabled (LPRC oscillator cannot be disabled. Clearing the SWDTEN bit in the RCON register will have no effect.) `0` = Watchdog Timer enabled/disabled by user software (LPRC can be disabled by clearing the SWDTEN bit in the RCON register)
WINDIS	FWDT	Immediate	Watchdog Timer Window Enable bit `1` = Watchdog Timer in Non-Window mode `0` = Watchdog Timer in Window mode
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
ALTI2C	FPOR	Immediate	Alternate I^{2}C pins `1` = I^{2}C mapped to SDA1/SCL1 pins `0` = I^{2}C mapped to ASDA1/ASCL1 pins
FPWRT<2:0>	FPOR	Immediate	Power-on Reset Timer Value Select bits `111` = PWRT = 128 ms `110` = PWRT = 64 ms `101` = PWRT = 32 ms `100` = PWRT = 16 ms `011` = PWRT = 8 ms `010` = PWRT = 4 ms `001` = PWRT = 2 ms `000` = PWRT = Disabled
JTAGEN	FICD	Immediate	**JTAG Enable bit** `1` = JTAG enabled `0` = JTAG disabled
ICS<1:0>	FICD	Immediate	**ICD Communication Channel Select bits** `11` = Communicate on PGEC1 and PGED1 `10` = Communicate on PGEC2 and PGED2 `01` = Communicate on PGEC3 and PGED3 `00` = Reserved, do not use

19.2 On-Chip Voltage Regulator

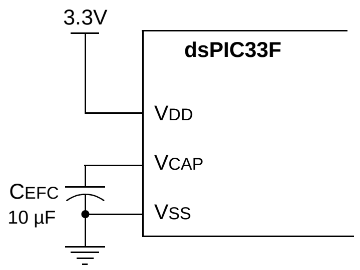
All of the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 devices power their core digital logic at a nominal 2.5V. This can create a conflict for designs that are required to operate at a higher typical voltage, such as 3.3V. To simplify system design, all devices in the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 family incorporate an on-chip regulator that allows the device to run its core logic from VDD.

The regulator provides power to the core from the other VDD pins. 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 19-1). This helps to maintain the stability of the regulator. The recommended value for the filter capacitor is provided in Table 22-13 located in **Section 22.1 "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 μs 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 19-1: CONNECTIONS FOR THE ON-CHIP VOLTAGE REGULATOR[1]



Note 1: These are typical operating voltages. Refer to Table 22-13 for the full operating ranges of VDD and VCAP.

2: It is important for the low-ESR capacitor to be placed as close as possible to the VCAP pin.

3: Typical VCAP pin voltage = 2.5V when VDD ≥ VDDMIN.

19.3 BOR: Brown-Out Reset

The Brown-out Reset (BOR) module is based on an internal voltage reference circuit that monitors the regulated voltage VCAP. The main purpose of the BOR module is to generate a device Reset when a brown-out condition occurs. Brown-out conditions are generally caused by glitches on the AC mains (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) will be applied before the internal Reset is released. If TPWRT = `0` and a crystal oscillator is being used, 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.

19.4 Watchdog Timer (WDT)

For dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 devices, the WDT is driven by the LPRC oscillator. When the WDT is enabled, the clock source is also enabled.

19.4.1 PRESCALER/POSTSCALER

The nominal WDT clock source from LPRC is 32 kHz. This feeds a prescaler than can be configured for either 5-bit (divide-by-32) or 7-bit (divide-by-128) operation. The prescaler is set by the WDTPRE Configuration bit. With a 32 kHz input, the prescaler yields a nominal WDT time-out period (T_{WDT}) 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 `PWRSAV` instruction is executed (i.e., Sleep or Idle mode is entered)
- When the device exits Sleep or Idle mode to resume normal operation
- By a `CLRWDT` instruction during normal execution

Note: The `CLRWDT` and `PWRSAV` instructions clear the prescaler and postscaler counts when executed.

19.4.2 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 device will wake the device and code execution will continue from where the `PWRSAV` instruction was executed. The corresponding SLEEP or IDLE bits (RCON<3:2>) will need to be cleared in software after the device wakes up.

19.4.3 ENABLING WDT

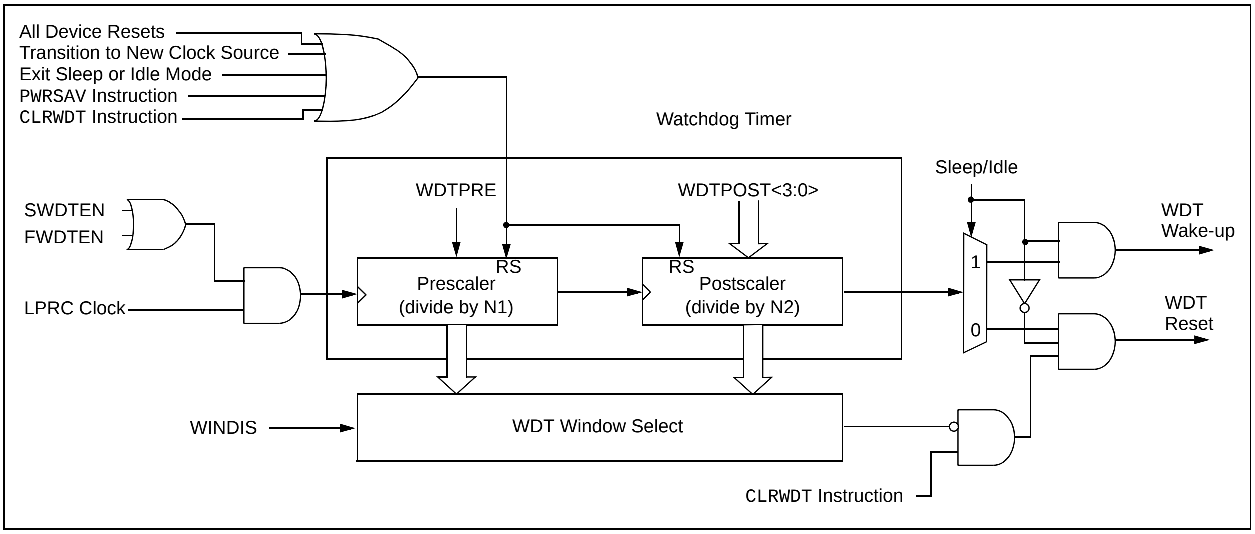
The WDT is enabled or disabled by the FWDTEN Configuration bit in the FWDT Configuration register. When the FWDTEN Configuration bit is set, the WDT is always enabled.

The WDT 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.

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.

FIGURE 19-2: WDT BLOCK DIAGRAM



19.5 JTAG Interface

dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 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.

19.6 Code Protection and CodeGuard™ Security

The dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 product families 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 the single chip.

The code protection features are controlled by the Configuration registers: FBS and FGS. The Secure segment and RAM is not implemented.

Note: Refer to **Section 23. *“CodeGuard™ Security”*** (DS70199) in the *“dsPIC33F/PIC24H Family Reference Manual”* for further information on usage, configuration and operation of CodeGuard Security.

TABLE 19-3: CODE FLASH SECURITY SEGMENT SIZES FOR 32 KBYTE DEVICES

CONFIG BITS		
BSS<2:0> = x11 **0K**	VS = 256 IW GS = 11008 IW	000000h 0001FEh 000200h 0007FEh 000800h 001FFEh 002000h 003FFEh 004000h 0057FEh
BSS<2:0> = x10 **256**	VS = 256 IW BS = 768 IW GS = 10240 IW	000000h 0001FEh 000200h 0007FEh 000800h 001FFEh 002000h 003FFEh 004000h 0057FEh
BSS<2:0> = x01 **768**	VS = 256 IW BS = 3840 IW GS = 7168 IW	000000h 0001FEh 000200h 0007FEh 000800h 001FFEh 002000h 003FFEh 004000h 0057FEh
BSS<2:0> = x00 **1792**	VS = 256 IW BS = 7936 IW GS = 3072 IW	000000h 0001FEh 000200h 0007FEh 000800h 001FFEh 002000h 003FFEh 004000h 0057FEh

TABLE 19-4: CODE FLASH SECURITY SEGMENT SIZES FOR 16 KBYTE DEVICES

CONFIG BITS		
BSS<2:0> = x11 **0K**	VS = 256 IW GS = 5376 IW	000000h 0001FEh 000200h 0007FEh 000800h 001FFEh 002000h 002BFEh
BSS<2:0> = x10 **256**	VS = 256 IW BS = 768 IW GS = 4608 IW	000000h 0001FEh 000200h 0007FEh 000800h 001FFEh 002000h 002BFEh
BSS<2:0> = x01 **768**	VS = 256 IW BS = 3840 IW GS = 1536 IW	000000h 0001FEh 000200h 0007FEh 000800h 001FFEh 002000h 002BFEh
BSS<2:0> = x00 **1792**	VS = 256 IW BS = 5376 IW	000000h 0001FEh 000200h 0007FEh 000800h 001FFEh 002000h 002BFEh

19.7 In-Circuit Serial Programming

dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 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) document 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

19.8 In-Circuit Debugger

When MPLAB® ICD 2 is selected as a debugger, the in-circuit debugging functionality is enabled. This function allows simple debugging functions when used with MPLAB IDE. Debugging functionality is controlled through the PGECx (Emulation/Debug Clock) and PGEDx (Emulation/Debug Data) pin functions.

Any 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 $\overline{\text{MCLR}}$, V_{DD}, V_{SS}, and the PGECx/PGEDx pin pair. In addition, when the feature is enabled, some of the resources are not available for general use. These resources include the first 80 bytes of data RAM and two I/O pins.

NOTES:

20.0 INSTRUCTION SET SUMMARY

Note: This data sheet summarizes the features of the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 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) 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 20-1 shows the general symbols used in describing the instructions.

The dsPIC33F instruction set summary in Table 20-2 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

Most instructions are a single word. Certain double-word instructions are designed to provide all of 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 `NOP`. 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 `NOP`. Notable exceptions are the `BRA` (unconditional/computed branch), indirect `CALL/GOTO`, all table reads and writes and `RETURN/RETFIE` instructions, which are single-word instructions but take two or three cycles. Certain instructions that involve skipping over the subsequent instruction require either two or three cycles if the skip is performed, depending on whether the instruction being skipped is a single-word or two-word instruction. Moreover, double-word moves require two cycles.

Note: For more details on the instruction set, refer to the *“16-bit MCU and DSC Programmer’s Reference Manual”* (DS70157).

TABLE 20-1: 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 ∈ {W13, [W13]+ = 2}
bit4	4-bit bit selection field (used in word addressed instructions) ∈ {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 ∈ {0x0000...0x1FFF}
lit1	1-bit unsigned literal ∈ {0,1}
lit4	4-bit unsigned literal ∈ {0...15}
lit5	5-bit unsigned literal ∈ {0...31}
lit8	8-bit unsigned literal ∈ {0...255}
lit10	10-bit unsigned literal ∈ {0...255} for Byte mode, {0:1023} for Word mode
lit14	14-bit unsigned literal ∈ {0...16384}
lit16	16-bit unsigned literal ∈ {0...65535}
lit23	23-bit unsigned literal ∈ {0...8388608}; LSb must be ‘`0`’
None	Field does not require an entry, may be blank
OA, OB, SA, SB	DSP Status bits: AccA Overflow, AccB Overflow, AccA Saturate, AccB Saturate
PC	Program Counter
Slit10	10-bit signed literal ∈ {-512...511}
Slit16	16-bit signed literal ∈ {-32768...32767}
Slit6	6-bit signed literal ∈ {-16...16}
Wb	Base W register ∈ {W0..W15}
Wd	Destination W register ∈ { Wd, [Wd], [Wd++], [Wd--], [++Wd], [--Wd] }
Wdo	Destination W register ∈ { Wnd, [Wnd], [Wnd++], [Wnd--], [++Wnd], [--Wnd], [Wnd+Wb] }
Wm,Wn	Dividend, Divisor working register pair (direct addressing)

TABLE 20-1: SYMBOLS USED IN OPCODE DESCRIPTIONS (CONTINUED)

Field	Description
Wm*Wm	Multiplicand and Multiplier working register pair for Square instructions ∈ {W4 * W4,W5 * W5,W6 * W6,W7 * W7}
Wm*Wn	Multiplicand and Multiplier working register pair for DSP instructions ∈ {W4 * W5,W4 * W6,W4 * W7,W5 * W6,W5 * W7,W6 * W7}
Wn	One of 16 working registers ∈ {W0..W15}
Wnd	One of 16 destination working registers ∈ {W0...W15}
Wns	One of 16 source working registers ∈ {W0...W15}
WREG	W0 (working register used in file register instructions)
Ws	Source W register ∈ { Ws, [Ws], [Ws++], [Ws--], [++Ws], [--Ws] }
Wso	Source W register ∈ { Wns, [Wns], [Wns++], [Wns--], [++Wns], [--Wns], [Wns+Wb] }
Wx	X data space prefetch address register for DSP instructions ∈ {[W8]+ = 6, [W8]+ = 4, [W8]+ = 2, [W8], [W8]- = 6, [W8]- = 4, [W8]- = 2, [W9]+ = 6, [W9]+ = 4, [W9]+ = 2, [W9], [W9]- = 6, [W9]- = 4, [W9]- = 2, [W9 + W12], none}
Wxd	X data space prefetch destination register for DSP instructions ∈ {W4...W7}
Wy	Y data space prefetch address register for DSP instructions ∈ {[W10]+ = 6, [W10]+ = 4, [W10]+ = 2, [W10], [W10]- = 6, [W10]- = 4, [W10]- = 2, [W11]+ = 6, [W11]+ = 4, [W11]+ = 2, [W11], [W11]- = 6, [W11]- = 4, [W11]- = 2, [W11 + W12], none}
Wyd	Y data space prefetch destination register for DSP instructions ∈ {W4...W7}

TABLE 20-2: INSTRUCTION SET OVERVIEW

Base Instr #	Assembly Mnemonic	Assembly Syntax	Description	# of Words	# of Cycles	Status Flags Affected
1	ADD	ADD Acc	Add Accumulators	1	1	OA,OB,SA,SB
		ADD f	f = f + WREG	1	1	C,DC,N,OV,Z
		ADD f,WREG	WREG = f + WREG	1	1	C,DC,N,OV,Z
		ADD #lit10,Wn	Wd = lit10 + Wd	1	1	C,DC,N,OV,Z
		ADD Wb,Ws,Wd	Wd = Wb + Ws	1	1	C,DC,N,OV,Z
		ADD Wb,#lit5,Wd	Wd = Wb + lit5	1	1	C,DC,N,OV,Z
		ADD Wso,#Slit4,Acc	16-bit Signed Add to Accumulator	1	1	OA,OB,SA,SB
2	ADDC	ADDC f	f = f + WREG + (C)	1	1	C,DC,N,OV,Z
		ADDC f,WREG	WREG = f + WREG + (C)	1	1	C,DC,N,OV,Z
		ADDC #lit10,Wn	Wd = lit10 + Wd + (C)	1	1	C,DC,N,OV,Z
		ADDC Wb,Ws,Wd	Wd = Wb + Ws + (C)	1	1	C,DC,N,OV,Z
		ADDC Wb,#lit5,Wd	Wd = Wb + lit5 + (C)	1	1	C,DC,N,OV,Z
3	AND	AND f	f = f .AND. WREG	1	1	N,Z
		AND f,WREG	WREG = f .AND. WREG	1	1	N,Z
		AND #lit10,Wn	Wd = lit10 .AND. Wd	1	1	N,Z
		AND Wb,Ws,Wd	Wd = Wb .AND. Ws	1	1	N,Z
		AND Wb,#lit5,Wd	Wd = Wb .AND. lit5	1	1	N,Z
4	ASR	ASR f	f = Arithmetic Right Shift f	1	1	C,N,OV,Z
		ASR f,WREG	WREG = Arithmetic Right Shift f	1	1	C,N,OV,Z
		ASR Ws,Wd	Wd = Arithmetic Right Shift Ws	1	1	C,N,OV,Z
		ASR Wb,Wns,Wnd	Wnd = Arithmetic Right Shift Wb by Wns	1	1	N,Z
		ASR Wb,#lit5,Wnd	Wnd = Arithmetic Right Shift Wb by lit5	1	1	N,Z
5	BCLR	BCLR f,#bit4	Bit Clear f	1	1	None
		BCLR Ws,#bit4	Bit Clear Ws	1	1	None
6	BRA	BRA C,Expr	Branch if Carry	1	1 (2)	None
		BRA GE,Expr	Branch if greater than or equal	1	1 (2)	None
		BRA GEU,Expr	Branch if unsigned greater than or equal	1	1 (2)	None
		BRA GT,Expr	Branch if greater than	1	1 (2)	None
		BRA GTU,Expr	Branch if unsigned greater than	1	1 (2)	None
		BRA LE,Expr	Branch if less than or equal	1	1 (2)	None
		BRA LEU,Expr	Branch if unsigned less than or equal	1	1 (2)	None
		BRA LT,Expr	Branch if less than	1	1 (2)	None
		BRA LTU,Expr	Branch if unsigned less than	1	1 (2)	None
		BRA N,Expr	Branch if Negative	1	1 (2)	None
		BRA NC,Expr	Branch if Not Carry	1	1 (2)	None
		BRA NN,Expr	Branch if Not Negative	1	1 (2)	None
		BRA NOV,Expr	Branch if Not Overflow	1	1 (2)	None
		BRA NZ,Expr	Branch if Not Zero	1	1 (2)	None
		BRA OA,Expr	Branch if Accumulator A overflow	1	1 (2)	None
		BRA OB,Expr	Branch if Accumulator B overflow	1	1 (2)	None
		BRA OV,Expr	Branch if Overflow	1	1 (2)	None
		BRA SA,Expr	Branch if Accumulator A saturated	1	1 (2)	None
		BRA SB,Expr	Branch if Accumulator B saturated	1	1 (2)	None
		BRA Expr	Branch Unconditionally	1	2	None
		BRA Z,Expr	Branch if Zero	1	1 (2)	None
		BRA Wn	Computed Branch	1	2	None
7	BSET	BSET f,#bit4	Bit Set f	1	1	None
		BSET Ws,#bit4	Bit Set Ws	1	1	None
8	BSW	BSW.C Ws,Wb	Write C bit to Ws<Wb>	1	1	None
		BSW.Z Ws,Wb	Write Z bit to Ws<Wb>	1	1	None
9	BTG	BTG f,#bit4	Bit Toggle f	1	1	None
		BTG Ws,#bit4	Bit Toggle Ws	1	1	None

TABLE 20-2: INSTRUCTION SET OVERVIEW (CONTINUED)

Base Instr #	Assembly Mnemonic	Assembly Syntax	Description	# of Words	# of Cycles	Status Flags Affected
10	BTSC	BTSC f,#bit4	Bit Test f, Skip if Clear	1	1 (2 or 3)	None
		BTSC Ws,#bit4	Bit Test Ws, Skip if Clear	1	1 (2 or 3)	None
11	BTSS	BTSS f,#bit4	Bit Test f, Skip if Set	1	1 (2 or 3)	None
		BTSS Ws,#bit4	Bit Test Ws, Skip if Set	1	1 (2 or 3)	None
12	BTST	BTST f,#bit4	Bit Test f	1	1	Z
		BTST.C Ws,#bit4	Bit Test Ws to C	1	1	C
		BTST.Z Ws,#bit4	Bit Test Ws to Z	1	1	Z
		BTST.C Ws,Wb	Bit Test Ws<Wb> to C	1	1	C
		BTST.Z Ws,Wb	Bit Test Ws<Wb> to Z	1	1	Z
13	BTSTS	BTSTS f,#bit4	Bit Test then Set f	1	1	Z
		BTSTS.C Ws,#bit4	Bit Test Ws to C, then Set	1	1	C
		BTSTS.Z Ws,#bit4	Bit Test Ws to Z, then Set	1	1	Z
14	CALL	CALL lit23	Call subroutine	2	2	None
		CALL Wn	Call indirect subroutine	1	2	None
15	CLR	CLR f	f = 0x0000	1	1	None
		CLR WREG	WREG = 0x0000	1	1	None
		CLR Ws	Ws = 0x0000	1	1	None
		CLR Acc,Wx,Wxd,Wy,Wyd,AWB	Clear Accumulator	1	1	OA,OB,SA,SB
16	CLRWDT	CLRWDT	Clear Watchdog Timer	1	1	WDTO,Sleep
17	COM	COM f	f = $\overline{f}$	1	1	N,Z
		COM f,WREG	WREG = $\overline{f}$	1	1	N,Z
		COM Ws,Wd	Wd = $\overline{Ws}$	1	1	N,Z
18	CP	CP f	Compare f with WREG	1	1	C,DC,N,OV,Z
		CP Wb,#lit5	Compare Wb with lit5	1	1	C,DC,N,OV,Z
		CP Wb,Ws	Compare Wb with Ws (Wb – Ws)	1	1	C,DC,N,OV,Z
19	CP0	CP0 f	Compare f with 0x0000	1	1	C,DC,N,OV,Z
		CP0 Ws	Compare Ws with 0x0000	1	1	C,DC,N,OV,Z
20	CPB	CPB f	Compare f with WREG, with Borrow	1	1	C,DC,N,OV,Z
		CPB Wb,#lit5	Compare Wb with lit5, with Borrow	1	1	C,DC,N,OV,Z
		CPB Wb,Ws	Compare Wb with Ws, with Borrow (Wb - Ws - $\overline{C}$)	1	1	C,DC,N,OV,Z
21	CPSEQ	CPSEQ Wb, Wn	Compare Wb with Wn, skip if =	1	1 (2 or 3)	None
22	CPSGT	CPSGT Wb, Wn	Compare Wb with Wn, skip if >	1	1 (2 or 3)	None
23	CPSLT	CPSLT Wb, Wn	Compare Wb with Wn, skip if <	1	1 (2 or 3)	None
24	CPSNE	CPSNE Wb, Wn	Compare Wb with Wn, skip if ≠	1	1 (2 or 3)	None
25	DAW	DAW Wn	Wn = decimal adjust Wn	1	1	C
26	DEC	DEC f	f = f - 1	1	1	C,DC,N,OV,Z
		DEC f,WREG	WREG = f - 1	1	1	C,DC,N,OV,Z
		DEC Ws,Wd	Wd = Ws - 1	1	1	C,DC,N,OV,Z
27	DEC2	DEC2 f	f = f - 2	1	1	C,DC,N,OV,Z
		DEC2 f,WREG	WREG = f - 2	1	1	C,DC,N,OV,Z
		DEC2 Ws,Wd	Wd = Ws - 2	1	1	C,DC,N,OV,Z
28	DISI	DISI #lit14	Disable Interrupts for k instruction cycles	1	1	None

TABLE 20-2: 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 20-2: 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 = $\overline{f}$ + 1	1	1	C,DC,N,OV,Z
		NEG f,WREG	WREG = $\overline{f}$ + 1	1	1	C,DC,N,OV,Z
		NEG Ws,Wd	Wd = $\overline{Ws}$ + 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	PWRSAV	PWRSAV #lit1	Go into Sleep or Idle mode	1	1	WDTO,Sleep
57	RCALL	RCALL Expr	Relative Call	1	2	None
		RCALL Wn	Computed Call	1	2	None
58	REPEAT	REPEAT #lit14	Repeat Next Instruction lit14 + 1 times	1	1	None
		REPEAT Wn	Repeat Next Instruction (Wn) + 1 times	1	1	None
59	RESET	RESET	Software device Reset	1	1	None
60	RETFIE	RETFIE	Return from interrupt	1	3 (2)	None
61	RETLW	RETLW #lit10,Wn	Return with literal in Wn	1	3 (2)	None
62	RETURN	RETURN	Return from Subroutine	1	3 (2)	None
63	RLC	RLC f	f = Rotate Left through Carry f	1	1	C,N,Z
		RLC f,WREG	WREG = Rotate Left through Carry f	1	1	C,N,Z
		RLC Ws,Wd	Wd = Rotate Left through Carry Ws	1	1	C,N,Z
64	RLNC	RLNC f	f = Rotate Left (No Carry) f	1	1	N,Z
		RLNC f,WREG	WREG = Rotate Left (No Carry) f	1	1	N,Z
		RLNC Ws,Wd	Wd = Rotate Left (No Carry) Ws	1	1	N,Z
65	RRC	RRC f	f = Rotate Right through Carry f	1	1	C,N,Z
		RRC f,WREG	WREG = Rotate Right through Carry f	1	1	C,N,Z
		RRC Ws,Wd	Wd = Rotate Right through Carry Ws	1	1	C,N,Z

TABLE 20-2: INSTRUCTION SET OVERVIEW (CONTINUED)

Base Instr #	Assembly Mnemonic	Assembly Syntax	Description	# of Words	# of Cycles	Status Flags Affected
66	RRNC	RRNC f	f = Rotate Right (No Carry) f	1	1	N,Z
		RRNC f,WREG	WREG = Rotate Right (No Carry) f	1	1	N,Z
		RRNC Ws,Wd	Wd = Rotate Right (No Carry) Ws	1	1	N,Z
67	SAC	SAC Acc,#Slit4,Wdo	Store Accumulator	1	1	None
		SAC.R Acc,#Slit4,Wdo	Store Rounded Accumulator	1	1	None
68	SE	SE Ws,Wnd	Wnd = sign-extended Ws	1	1	C,N,Z
69	SETM	SETM f	f = 0xFFFF	1	1	None
		SETM WREG	WREG = 0xFFFF	1	1	None
		SETM Ws	Ws = 0xFFFF	1	1	None
70	SFTAC	SFTAC Acc,Wn	Arithmetic Shift Accumulator by (Wn)	1	1	OA,OB,OAB, SA,SB,SAB
		SFTAC Acc,#Slit6	Arithmetic Shift Accumulator by Slit6	1	1	OA,OB,OAB, SA,SB,SAB
71	SL	SL f	f = Left Shift f	1	1	C,N,OV,Z
		SL f,WREG	WREG = Left Shift f	1	1	C,N,OV,Z
		SL Ws,Wd	Wd = Left Shift Ws	1	1	C,N,OV,Z
		SL Wb,Wns,Wnd	Wnd = Left Shift Wb by Wns	1	1	N,Z
		SL Wb,#lit5,Wnd	Wnd = Left Shift Wb by lit5	1	1	N,Z
72	SUB	SUB Acc	Subtract Accumulators	1	1	OA,OB,OAB, SA,SB,SAB
		SUB f	f = f - WREG	1	1	C,DC,N,OV,Z
		SUB f,WREG	WREG = f - WREG	1	1	C,DC,N,OV,Z
		SUB #lit10,Wn	Wn = Wn - lit10	1	1	C,DC,N,OV,Z
		SUB Wb,Ws,Wd	Wd = Wb - Ws	1	1	C,DC,N,OV,Z
		SUB Wb,#lit5,Wd	Wd = Wb - lit5	1	1	C,DC,N,OV,Z
73	SUBB	SUBB f	f = f - WREG - ($\overline{C}$)	1	1	C,DC,N,OV,Z
		SUBB f,WREG	WREG = f - WREG - ($\overline{C}$)	1	1	C,DC,N,OV,Z
		SUBB #lit10,Wn	Wn = Wn - lit10 - ($\overline{C}$)	1	1	C,DC,N,OV,Z
		SUBB Wb,Ws,Wd	Wd = Wb - Ws - ($\overline{C}$)	1	1	C,DC,N,OV,Z
		SUBB Wb,#lit5,Wd	Wd = Wb - lit5 - ($\overline{C}$)	1	1	C,DC,N,OV,Z
74	SUBR	SUBR f	f = WREG - f	1	1	C,DC,N,OV,Z
		SUBR f,WREG	WREG = WREG - f	1	1	C,DC,N,OV,Z
		SUBR Wb,Ws,Wd	Wd = Ws - Wb	1	1	C,DC,N,OV,Z
		SUBR Wb,#lit5,Wd	Wd = lit5 - Wb	1	1	C,DC,N,OV,Z
75	SUBBR	SUBBR f	f = WREG - f - ($\overline{C}$)	1	1	C,DC,N,OV,Z
		SUBBR f,WREG	WREG = WREG - f - ($\overline{C}$)	1	1	C,DC,N,OV,Z
		SUBBR Wb,Ws,Wd	Wd = Ws - Wb - ($\overline{C}$)	1	1	C,DC,N,OV,Z
		SUBBR Wb,#lit5,Wd	Wd = lit5 - Wb - ($\overline{C}$)	1	1	C,DC,N,OV,Z
76	SWAP	SWAP.b Wn	Wn = nibble swap Wn	1	1	None
		SWAP Wn	Wn = byte swap Wn	1	1	None
77	TBLRDH	TBLRDH Ws,Wd	Read Prog<23:16> to Wd<7:0>	1	2	None
78	TBLRDL	TBLRDL Ws,Wd	Read Prog<15:0> to Wd	1	2	None
79	TBLWTH	TBLWTH Ws,Wd	Write Ws<7:0> to Prog<23:16>	1	2	None
80	TBLWTL	TBLWTL Ws,Wd	Write Ws to Prog<15:0>	1	2	None
81	ULNK	ULNK	Unlink Frame Pointer	1	1	None
82	XOR	XOR f	f = f .XOR. WREG	1	1	N,Z
		XOR f,WREG	WREG = f .XOR. WREG	1	1	N,Z
		XOR #lit10,Wn	Wd = lit10 .XOR. Wd	1	1	N,Z
		XOR Wb,Ws,Wd	Wd = Wb .XOR. Ws	1	1	N,Z
		XOR Wb,#lit5,Wd	Wd = Wb .XOR. lit5	1	1	N,Z
83	ZE	ZE Ws,Wnd	Wnd = Zero-extend Ws	1	1	C,Z,N

21.0 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

21.1 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.

21.2 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.

21.3 HI-TECH C for Various Device Families

The HI-TECH C Compiler code development systems are complete ANSI C compilers for Microchip's PIC family of microcontrollers and the dsPIC family of digital signal controllers. These compilers provide powerful integration capabilities, omniscient code generation and ease of use.

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

The compilers include a macro assembler, linker, preprocessor, and one-step driver, and can run on multiple platforms.

21.4 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

21.5 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

21.6 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

21.7 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.

21.8 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.

21.9 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.

21.10 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.

21.11 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.

21.12 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.

21.13 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) for the complete list of demonstration, development and evaluation kits.

22.0 ELECTRICAL CHARACTERISTICS

This section provides an overview of dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 electrical characteristics. Additional information will be provided in future revisions of this document as it becomes available.

Absolute maximum ratings for the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 family 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(1)

Ambient temperature under bias	-40°C to +125°C
Storage temperature	-65°C to +160°C
Voltage on VDD with respect to VSS	-0.3V to +4.0V
Voltage on any pin that is not 5V tolerant with respect to VSS(4)	-0.3V to (VDD + 0.3V)
Voltage on any 5V tolerant pin with respect to VSS when VDD ≥ 3.0V(4)	-0.3V to +5.6V
Voltage on any 5V tolerant pin with respect to Vss when VDD < 3.0V(4)	-0.3V to 3.6V
Maximum current out of VSS pin	300 mA
Maximum current into VDD pin(2)	250 mA
Maximum output current sunk by any I/O pin(3)	4 mA
Maximum output current sourced by any I/O pin(3)	4 mA
Maximum current sunk by all ports	200 mA
Maximum current sourced by all ports(2)	200 mA

Note 1: Stresses above those listed under “Absolute Maximum Ratings” can cause permanent damage to the device. This is a stress rating only, and functional operation of the device at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure to maximum rating conditions for extended periods can affect device reliability.

2: Maximum allowable current is a function of device maximum power dissipation (see Table 22-2).

3: Exceptions are CLKOUT, which is able to sink/source 25 mA, and the VREF+, VREF-, SCLx, SDAx, PGECx and PGEDx pins, which are able to sink/source 12 mA.

4: Refer to the **“Pin Diagrams”** section for 5V tolerant pins.

22.1 DC Characteristics

TABLE 22-1: OPERATING MIPS VS. VOLTAGE

Characteristic	V_{DD} Range (in Volts)	Temp Range (in °C)	Max MIPS
			dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304
	3.0-3.6V	-40°C to +85°C	40
	3.0-3.6V	-40°C to +125°C	40

TABLE 22-2: THERMAL OPERATING CONDITIONS

Rating	Symbol	Min	Typ	Max	Unit
Industrial Temperature Devices					
Operating Junction Temperature Range	T_J	-40	—	+125	°C
Operating Ambient Temperature Range	T_A	-40	—	+85	°C
Extended Temperature Devices					
Operating Junction Temperature Range	T_J	-40	—	+155	°C
Operating Ambient Temperature Range	T_A	-40	—	+125	°C
Power Dissipation: Internal chip power dissipation: $P_{INT} = V_{DD} \times (I_{DD} - \Sigma I_{OH})$ I/O Pin Power Dissipation: $I/O = \Sigma (\{V_{DD} - V_{OH}\} \times I_{OH}) + \Sigma (V_{OL} \times I_{OL})$	P_D	$P_{INT} + P_{I/O}$			W
Maximum Allowed Power Dissipation	P_{DMAX}	$(T_J - T_A)/\theta_{JA}$			W

TABLE 22-3: THERMAL PACKAGING CHARACTERISTICS

Characteristic	Symbol	Typ	Max	Unit	Notes
Package Thermal Resistance, 44-pin QFN	θ_{JA}	32	—	°C/W	**1**
Package Thermal Resistance, 44-pin TFQP	θ_{JA}	45	—	°C/W	**1**
Package Thermal Resistance, 28-pin SPDIP	θ_{JA}	45	—	°C/W	**1**
Package Thermal Resistance, 28-pin SOIC	θ_{JA}	50	—	°C/W	**1**
Package Thermal Resistance, 28-pin SSOP	θ_{JA}	71	—	°C/W	**1**
Package Thermal Resistance, 28-pin QFN-S	θ_{JA}	35	—	°C/W	**1**

Note 1: Junction to ambient thermal resistance, Theta-JA (θ_{JA}) numbers are achieved by package simulations.

TABLE 22-4: 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[1]**	**Max**	**Units**	**Conditions**
Operating Voltage							
DC10	**Supply Voltage**						
	VDD	—	3.0	—	3.6	V	Industrial and Extended
DC12	VDR	**RAM Data Retention Voltage[2]**	1.8	—	—	V	—
DC16	VPOR	**VDD Start Voltage** to ensure internal Power-on Reset signal	—	—	VSS	V	—
DC17	SVDD	**VDD Rise Rate** to ensure internal Power-on Reset signal	0.03	—	—	V/ms	0-3.0V in 0.1s

Note 1: Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.

2: This is the limit to which VDD can be lowered without losing RAM data.

TABLE 22-5: 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(1)**	**Max**	**Units**	**Conditions**		
Operating Current (IDD)(2)						
DC20d	20	30	mA	-40°C	3.3V	10 MIPS(3)
DC20a	19	22	mA	+25°C		
DC20b	19	25	mA	+85°C		
DC20c	19	30	mA	+125°C		
DC21d	28	40	mA	-40°C	3.3V	16 MIPS(3)
DC21a	27	30	mA	+25°C		
DC21b	27	32	mA	+85°C		
DC21c	27	36	mA	+125°C		
DC22d	33	50	mA	-40°C	3.3V	20 MIPS(3)
DC22a	33	40	mA	+25°C		
DC22b	33	40	mA	+85°C		
DC22c	33	50	mA	+125°C		
DC23d	44	60	mA	-40°C	3.3V	30 MIPS(3)
DC23a	43	50	mA	+25°C		
DC23b	42	55	mA	+85°C		
DC23c	41	65	mA	+125°C		
DC24d	55	75	mA	-40°C	3.3V	40 MIPS
DC24a	54	65	mA	+25°C		
DC24b	52	70	mA	+85°C		
DC24c	51	80	mA	+125°C		

Note 1: Data in “Typical” column is at 3.3V, 25°C unless otherwise stated.

2: The supply current is mainly 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: OSC1 driven with external square wave from rail to rail. All I/O pins are configured as inputs and pulled to VSS. $\overline{\text{MCLR}}$ = 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 (PMD bits are all zeroed).

3: These parameters are characterized, but are not tested in manufacturing.

TABLE 22-6: DC CHARACTERISTICS: IDLE CURRENT (I_{IDLE})

DC CHARACTERISTICS			**Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤ T_A ≤ +85°C for Industrial -40°C ≤ T_A ≤ +125°C for Extended			
Parameter No.	**Typical(1)**	**Max**	**Units**	**Conditions**		
Idle Current (I_{IDLE}): Core OFF Clock ON Base Current(2)						
DC40d	7	20	mA	-40°C	3.3V	10 MIPS(3)
DC40a	6	7	mA	+25°C		
DC40b	6	10	mA	+85°C		
DC40c	6	20	mA	+125°C		
DC41d	10	20	mA	-40°C	3.3V	16 MIPS(3)
DC41a	8	9	mA	+25°C		
DC41b	8	10	mA	+85°C		
DC41c	8	20	mA	+125°C		
DC42d	11	20	mA	-40°C	3.3V	20 MIPS(3)
DC42a	10	10	mA	+25°C		
DC42b	10	12	mA	+85°C		
DC42c	10	20	mA	+125°C		
DC43d	14	25	mA	-40°C	3.3V	30 MIPS(3)
DC43a	13	14	mA	+25°C		
DC43b	13	15	mA	+85°C		
DC43c	13	25	mA	+125°C		
DC44d	14	25	mA	-40°C	3.3V	40 MIPS
DC44a	17	20	mA	+25°C		
DC44b	17	20	mA	+85°C		
DC44c	18	30	mA	+125°C		

Note 1: Data in “Typical” column is at 3.3V, 25°C unless otherwise stated.

2: Base I_{IDLE} current is measured with core off, clock on and all modules turned off. Peripheral Module Disable SFR registers are zeroed. All I/O pins are configured as inputs and pulled to V_{SS}.

3: These parameters are characterized, but are not tested in manufacturing.

TABLE 22-7: DC CHARACTERISTICS: POWER-DOWN CURRENT (I_{PD})

DC CHARACTERISTICS			**Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤ T_A ≤ +85°C for Industrial -40°C ≤ T_A ≤ +125°C for Extended			
Parameter No.	**Typical**[1]	**Max**	**Units**	**Conditions**		
Power-Down Current (I_{PD})[2]						
DC60d	55	500	μA	-40°C	3.3V	Base Power-Down Current[3,4]
DC60a	63	300	μA	+25°C		
DC60b	85	350	μA	+85°C		
DC60c	146	600	μA	+125°C		
DC61d	8	15	μA	-40°C	3.3V	Watchdog Timer Current: ΔI_{WDT}[3,5]
DC61a	2	3	μA	+25°C		
DC61b	2	2	μA	+85°C		
DC61c	3	5	μA	+125°C		

Note 1: Data in the Typical column is at 3.3V, 25°C unless otherwise stated.

2: Base I_{PD} is measured with all peripherals and clocks shut down. All I/Os are configured as inputs and pulled to V_{SS}. WDT, etc., are all switched off, and VREGS (RCON<8>) = 1.

3: The Δ current is the additional current consumed when the module is enabled. This current should be added to the base I_{PD} current.

4: These currents are measured on the device containing the most memory in this family.

5: These parameters are characterized, but are not tested in manufacturing.

TABLE 22-8: DC CHARACTERISTICS: DOZE CURRENT (I_{DOZE})

DC CHARACTERISTICS			**Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤ T_A ≤ +85°C for Industrial -40°C ≤ T_A ≤ +125°C for Extended				
Parameter No.	**Typical**[1,2]	**Max**	**Doze Ratio**	**Units**	**Conditions**		
DC73a	41	51	1:2	mA	-40°C	3.3V	40 MIPS
DC73f	20	28	1:64	mA			
DC73g	19	24	1:128	mA			
DC70a	40	46	1:2	mA	+25°C	3.3V	40 MIPS
DC70f	18	20	1:64	mA			
DC70g	18	20	1:128	mA			
DC71a	40	46	1:2	mA	+85°C	3.3V	40 MIPS
DC71f	18	25	1:64	mA			
DC71g	18	20	1:128	mA			
DC72a	39	55	1:2	mA	+125°C	3.3V	40 MIPS
DC72f	18	30	1:64	mA			
DC72g	18	25	1:128	mA			

Note 1: Data in the Typical column is at 3.3V, 25°C unless otherwise stated.

2: Parameters with Doze ratios of 1:2 and 1:64 are characterized, but are not tested in manufacturing.

TABLE 22-9: 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(1)**	**Max**	**Units**	**Conditions**
	VIL	**Input Low Voltage**					
DI10		I/O pins	VSS	—	0.2 VDD	V	
DI15		$\overline{\text{MCLR}}$	VSS	—	0.2 VDD	V	
DI16		I/O Pins with OSC1 or SOSCI	VSS	—	0.2 VDD	V	
DI18		SDAx, SCLx	VSS	—	0.3 VDD	V	SMBus disabled
DI19		SDAx, SCLx	VSS	—	0.8	V	SMBus enabled
	VIH	**Input High Voltage**					
DI20		I/O Pins Not 5V Tolerant(4)	0.7 VDD	—	VDD	V	—
		I/O Pins 5V Tolerant(4)	0.7 VDD	—	5.5	V	
DI28		SDAx, SCLx	0.7 VDD	—	5.5	V	SMBus disabled
DI29		SDAx, SCLx	2.1	—	5.5	V	SMBus enabled
	ICNPU	**CNx Pull-up Current**					
DI30			50	250	400	μA	VDD = 3.3V, VPIN = VSS
	IIL	**Input Leakage Current(2,3)**					
DI50		I/O Pins 5V Tolerant(4)	—	—	±2	μA	VSS ≤VPIN ≤VDD, Pin at high-impedance
DI51		I/O Pins Not 5V Tolerant(4)	—	—	±1	μA	VSS ≤VPIN ≤VDD, Pin at high-impedance, -40°C ≤TA ≤+85°C
DI51a		I/O Pins Not 5V Tolerant(4)	—	—	±2	μA	Shared with external reference pins, -40°C ≤TA ≤+85°C
DI51b		I/O Pins Not 5V Tolerant(4)	—	—	±3.5	μA	VSS ≤VPIN ≤VDD, Pin at high-impedance, -40°C ≤TA ≤+125°C
DI51c		I/O Pins Not 5V Tolerant(4)	—	—	±8	μA	Analog pins shared with external reference pins, -40°C ≤TA ≤+125°C
DI55		$\overline{\text{MCLR}}$	—	—	±2	μA	VSS ≤VPIN ≤VDD
DI56		OSC1	—	—	±2	μA	VSS ≤VPIN ≤VDD, 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 $\overline{\text{MCLR}}$ pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.

3: Negative current is defined as current sourced by the pin.

4: See **“Pin Diagrams”** for a list of digital-only and analog pins.

5: VIL source < (VSS – 0.3). Characterized but not tested.

6: Non-5V tolerant pins VIH source > (VDD + 0.3), 5V tolerant pins VIH source > 5V or devices with USB, “D+” and “D-“ VIH source > (VUSB + 0.3). Characterized but not tested.

7: Digital 5V tolerant pins cannot tolerate any “positive” input injection current from input sources > 5V.

8: Injection currents > | 0 | can affect the ADC results by approximately 4-6 counts.

9: Any number and/or combination of I/O pins not excluded under IICL or IICH conditions are permitted provided the mathematical “absolute instantaneous” sum of the input injection currents from all pins do not exceed the specified limit. Characterized but not tested.

TABLE 22-9: DC CHARACTERISTICS: I/O PIN INPUT SPECIFICATIONS (CONTINUED)

DC CHARACTERISTICS			**Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)** Operating temperature $-40°C \leq T_A \leq +85°C$ for Industrial $-40°C \leq T_A \leq +125°C$ for Extended				
Param No.	**Symbol**	**Characteristic**	**Min**	**Typ**[1]	**Max**	**Units**	**Conditions**
DI60a	I_{ICL}	**Input Low Injection Current**	0	—	-5[5,8]	mA	All pins except V_{DD}, V_{SS}, AV_{DD}, AV_{SS}, $\overline{MCLR}$, V_{CAP}, SOSCI, SOSCO, and RB14
DI60b	I_{ICH}	**Input High Injection Current**	0	—	+5[6,7,8]	mA	All pins except V_{DD}, V_{SS}, AV_{DD}, AV_{SS}, $\overline{MCLR}$, V_{CAP}, SOSCI, SOSCO, RB14, and digital 5V-tolerant designated pins
DI60c	$\sum I_{ICT}$	**Total Input Injection Current** **(sum of all I/O and control pins)**	-20[9]	—	+20[9]	mA	Absolute instantaneous sum of all ± input injection currents from all I/O pins $(\mid I_{ICL} + \mid I_{ICH} \mid) \leq \sum I_{ICT}$

Note 1: Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.

2: The leakage current on the $\overline{MCLR}$ pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.

3: Negative current is defined as current sourced by the pin.

4: See **“Pin Diagrams”** for a list of digital-only and analog pins.

5: V_{IL} source < (V_{SS} – 0.3). Characterized but not tested.

6: Non-5V tolerant pins V_{IH} source > (V_{DD} + 0.3), 5V tolerant pins V_{IH} source > 5V or devices with USB, “D+” and “D-“ V_{IH} source > (V_{USB} + 0.3). Characterized but not tested.

7: Digital 5V tolerant pins cannot tolerate any “positive” input injection current from input sources > 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_{ICL} or I_{ICH} 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 22-10: DC CHARACTERISTICS: I/O PIN OUTPUT SPECIFICATIONS

DC CHARACTERISTICS			**Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤ T_A ≤ +85°C for Industrial -40°C ≤ T_A ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristic**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
	V_{OL}	**Output Low Voltage**					
DO10		I/O ports	—	—	0.4	V	I_{OL} = 2mA, V_{DD} = 3.3V
DO16		OSC2/CLKO	—	—	0.4	V	I_{OL} = 2mA, V_{DD} = 3.3V
	V_{OH}	**Output High Voltage**					
DO20		I/O ports	2.40	—	—	V	I_{OH} = -2.3 mA, V_{DD} = 3.3V
DO26		OSC2/CLKO	2.41	—	—	V	I_{OH} = -1.3 mA, V_{DD} = 3.3V

TABLE 22-11: ELECTRICAL CHARACTERISTICS: BOR

DC CHARACTERISTICS			**Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤ T_A ≤ +85°C for Industrial -40°C ≤ T_A ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristic[1]**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
BO10	V_{BOR}	BOR Event on V_{DD} transition high-to-low	2.40	—	2.55	V	V_{DD}

Note 1: Parameters are for design guidance only and are not tested in manufacturing.

TABLE 22-12: DC CHARACTERISTICS: PROGRAM MEMORY

DC CHARACTERISTICS			**Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤ T_A ≤ +85°C for Industrial -40°C ≤ T_A ≤ +125°C for Extended					
Param No.	**Symbol**	**Characteristic(3)**	**Min**	**Typ(1)**	**Max**	**Units**	**Conditions**	
		Program Flash Memory						
D130	E_P	Cell Endurance	10,000	—	—	E/W	-40° C to +125° C	
D131	V_{PR}	V_{DD} for Read	V_{MIN}	—	3.6	V	V_{MIN} = Minimum operating voltage	
D132B	V_{PEW}	V_{DD} for Self-Timed Write	V_{MIN}	—	3.6	V	V_{MIN} = Minimum operating voltage	
D134	T_{RETD}	Characteristic Retention	20	—	—	Year	Provided no other specifications are violated, -40° C to +125° C	
D135	I_{DDP}	Supply Current during Programming	—	10	—	mA		
D136a	T_{RW}	Row Write Time	1.32	—	1.74	ms	T_{RW} = 11064 FRC cycles, T_A = +85°C, See **Note 2**	
D136b	T_{RW}	Row Write Time	1.28	—	1.79	ms	T_{RW} = 11064 FRC cycles, T_A = +125°C, See **Note 2**	
D137a	T_{PE}	Page Erase Time	20.1	—	26.5	ms	T_{PE} = 168517 FRC cycles, T_A = +85°C, See **Note 2**	
D137b	T_{PE}	Page Erase Time	19.5	—	27.3	ms	T_{PE} = 168517 FRC cycles, T_A = +125°C, See **Note 2**	
D138a	T_{WW}	Word Write Cycle Time	42.3	—	55.9	µs	T_{WW} = 355 FRC cycles, T_A = +85°C, See **Note 2**	
D138b	T_{WW}	Word Write Cycle Time	41.1	—	57.6	µs	T_{WW} = 355 FRC cycles, T_A = +125°C, See **Note 2**	

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 22-18) and the value of the FRC Oscillator Tuning register (see Register 8-4). For complete details on calculating the Minimum and Maximum time see **Section 5.3 “Programming Operations”**.

3: These parameters are assured by design, but are not characterized or tested in manufacturing.

TABLE 22-13: INTERNAL VOLTAGE REGULATOR SPECIFICATIONS

Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ T_A ≤ +85°C for Industrial -40°C ≤ T_A ≤ +125°C for Extended							
Param No.	**Symbol**	**Characteristics**	**Min**	**Typ**	**Max**	**Units**	**Comments**
—	C_{EFC}	External Filter Capacitor Value(1)	4.7	10	—	µF	Capacitor must be low series resistance (< 5 ohms)

Note 1: Typical V_{CAP} voltage = 2.5V when $V_{DD} \geq V_{DDMIN}$.

22.2 AC Characteristics and Timing Parameters

The information contained in this section defines dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 AC characteristics and timing parameters.

TABLE 22-14: TEMPERATURE AND VOLTAGE SPECIFICATIONS – AC

AC CHARACTERISTICS	**Standard Operating Conditions: 3.0V to 3.6V** **(unless otherwise stated)** Operating temperature -40°C ≤ T_A ≤ +85°C for Industrial -40°C ≤ T_A ≤ +125°C for Extended Operating voltage V_{DD} range as described in Table 22-1.

FIGURE 22-1: LOAD CONDITIONS FOR DEVICE TIMING SPECIFICATIONS

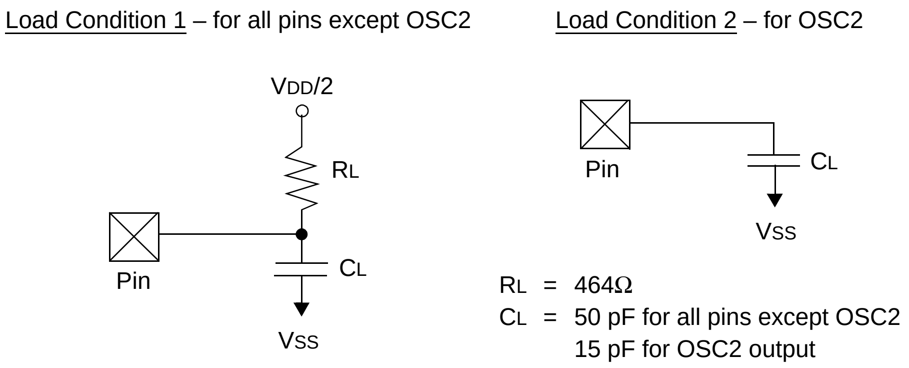


TABLE 22-15: CAPACITIVE LOADING REQUIREMENTS ON OUTPUT PINS

Param No.	Symbol	Characteristic	Min	Typ	Max	Units	Conditions
DO50	C_{OSC2}	OSC2/SOSC2 pin	—	—	15	pF	In XT and HS modes when external clock is used to drive OSC1
DO56	C_{IO}	All I/O pins and OSC2	—	—	50	pF	EC mode
DO58	C_B	SCLx, SDAx	—	—	400	pF	In I^{2}C™ mode

FIGURE 22-2: EXTERNAL CLOCK TIMING

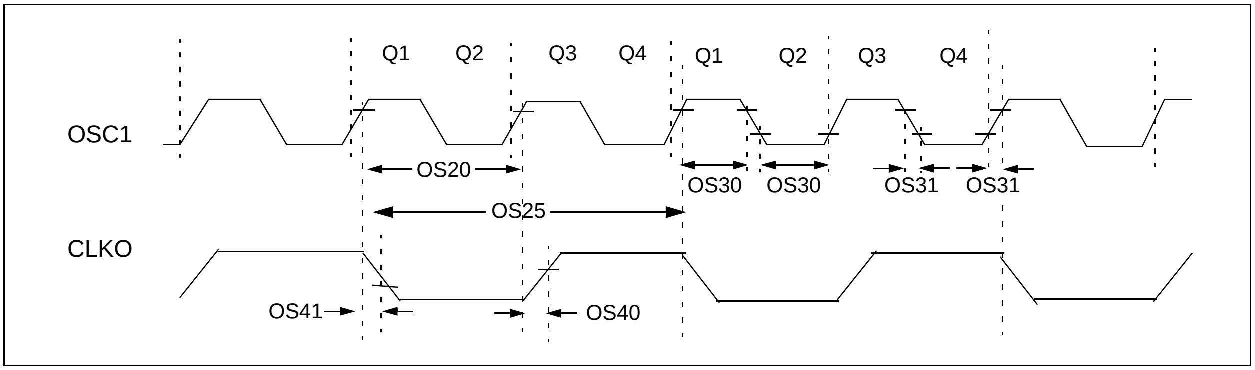


TABLE 22-16: EXTERNAL CLOCK TIMING REQUIREMENTS

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.	**Symbol**	**Characteristic**	**Min**	**Typ$^{(1)}$**	**Max**	**Units**	**Conditions**
OS10	FIN	External CLKI Frequency$^{(4)}$ (External clocks allowed only in EC and ECPLL modes)	DC	—	40	MHz	EC
		Oscillator Crystal Frequency$^{(5)}$	3.5 10 	— — —	10 40 33	MHz MHz kHz	XT HS SOSC
OS20	TOSC	TOSC = 1/FOSC$^{(4)}$	12.5	—	DC	ns	—
OS25	TCY	Instruction Cycle Time$^{(2,4)}$	25	—	DC	ns	—
OS30	TosL, TosH	External Clock in (OSC1)$^{(5)}$ High or Low Time	0.375 x TOSC	—	0.625 x TOSC	ns	EC
OS31	TosR, TosF	External Clock in (OSC1)$^{(5)}$ Rise or Fall Time	—	—	20	ns	EC
OS40	TckR	CLKO Rise Time$^{(3,5)}$	—	5.2	—	ns	—
OS41	TckF	CLKO Fall Time$^{(3,5)}$	—	5.2	—	ns	—
OS42	GM	External Oscillator Transconductance$^{(6)}$	14	16	18	mA/V	VDD = 3.3V TA = +25ºC

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 can result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at “min.” values with an external clock applied to the OSC1/CLKI pin. When an external clock input is used, the “max.” cycle time limit is “DC” (no clock) for all devices.

3: Measurements are taken in EC mode. The CLKO signal is measured on the OSC2 pin.

4: These parameters are characterized by similarity, but are tested in manufacturing at FIN = 40 MHz only.

5: These parameters are characterized by similarity, but are not tested in manufacturing.

6: Data for this parameter is preliminary. This parameter is characterized, but is not tested in manufacturing.

TABLE 22-17: PLL CLOCK TIMING SPECIFICATIONS (VDD = 3.0V TO 3.6V)

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(1)**	**Max**	**Units**	**Conditions**
OS50	FPLLI	PLL Voltage Controlled Oscillator (VCO) Input Frequency Range(2)	0.8	—	8	MHz	ECPLL, XTPLL modes
OS51	FSYS	On-Chip VCO System Frequency(3)	100	—	200	MHz	—
OS52	TLOCK	PLL Start-up Time (Lock Time)(3)	0.9	1.5	3.1	ms	—
OS53	DCLK	CLKO Stability (Jitter)(3)	-3	0.5	3	%	Measured over 100 ms period

Note 1: Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.

2: These parameters are characterized by similarity, but are tested in manufacturing at 7.7 MHz input only.

3: 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:

$$\text{Peripheral Clock Jitter} = \frac{D_{CLK}}{\sqrt{\left(\frac{F_{OSC}}{\text{Peripheral Bit Rate Clock}}\right)}}$$

For example: Fosc = 32 MHz, DCLK = 3%, SPI bit rate clock, (i.e., SCK) is 2 MHz.

$$\text{SPI SCK Jitter} = \left[\frac{D_{CLK}}{\sqrt{\left(\frac{32\ MHz}{2\ MHz}\right)}}\right] = \left[\frac{3\%}{\sqrt{16}}\right] = \left[\frac{3\%}{4}\right] = 0.75\%$$

TABLE 22-18: AC CHARACTERISTICS: INTERNAL RC 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**	
	Internal FRC Accuracy @ FRC Frequency = 7.37 MHz(1)						
F20a	FRC	-2	—	+2	%	-40°C ≤TA ≤+85°C	VDD = 3.0-3.6V
F20b	FRC	-5	—	+5	%	-40°C ≤TA ≤+125°C	VDD = 3.0-3.6V

Note 1: Frequency calibrated at 25°C and 3.3V. TUN bits can be used to compensate for temperature drift.

TABLE 22-19: INTERNAL RC 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**	
	LPRC @ 32.768 kHz(1,2)						
F21a	LPRC	-15	±6	+15	%	-40°C ≤TA ≤+85°C	VDD = 3.0-3.6V

Note 1: Change of LPRC frequency as VDD changes.

2: LPRC impacts the Watchdog Timer Time-out Period (TWDT1). See **Section 19.4 “Watchdog Timer (WDT)”** for more information.

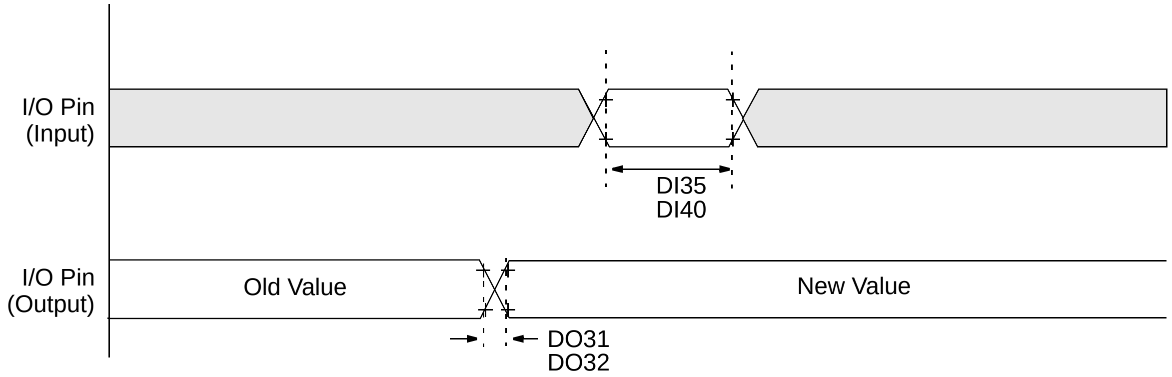
TABLE 22-19: INTERNAL RC ACCURACY

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**	
F21b	LPRC	-40	—	+40	%	-40°C ≤TA ≤+125°C	VDD = 3.0-3.6V

Note 1: Change of LPRC frequency as VDD changes.

2: LPRC impacts the Watchdog Timer Time-out Period (TWDT1). See **Section 19.4 “Watchdog Timer (WDT)”** for more information.

FIGURE 22-3: I/O TIMING CHARACTERISTICS



Note: Refer to Figure 22-1 for load conditions.

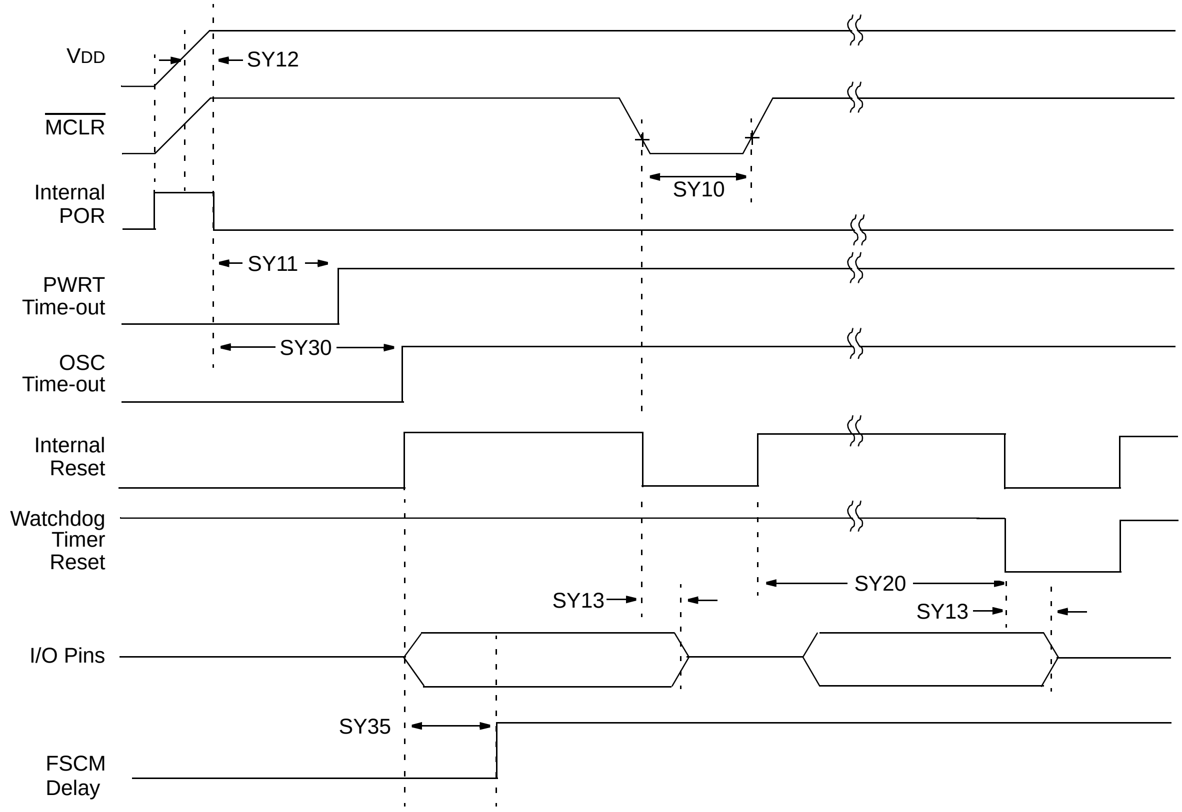
TABLE 22-20: I/O TIMING REQUIREMENTS

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(2)**	**Min**	**Typ(1)**	**Max**	**Units**	**Conditions**
DO31	TIOR	Port Output Rise Time	—	10	25	ns	—
DO32	TIOF	Port Output Fall Time	—	10	25	ns	—
DI35	TINP	INTx Pin High or Low Time (input)	25	—	—	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.

2: These parameters are characterized, but are not tested in manufacturing.

FIGURE 22-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING CHARACTERISTICS



Note: Refer to Figure 22-1 for load conditions.

TABLE 22-21: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER, POWER-UP TIMER TIMING REQUIREMENTS

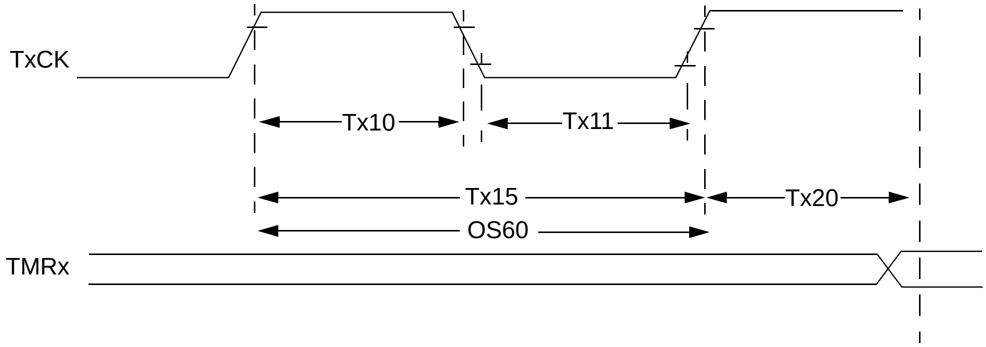
AC CHARACTERISTICS			**Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended				
Param No.	**Symbol**	**Characteristic**	**Min**	**Typ(2)**	**Max**	**Units**	**Conditions**
SY10	TMCL	$\overline{\text{MCLR}}$ Pulse-Width (low)(1)	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)	3	10	30	µs	-40°C to +85°C
SY13	TIOZ	I/O High-Impedance from $\overline{\text{MCLR}}$ Low or Watchdog Timer Reset(1)	0.68	0.72	1.2	µs	—
SY20	TWDT1	Watchdog Timer Time-out Period(1)	—	—	—	ms	See **Section 19.4 “Watchdog Timer (WDT)”** and LPRC parameter F21a (Table 22-19).
SY30	TOST	Oscillator Start-up Time	—	1024 TOSC	—	—	TOSC = OSC1 period
SY35	TFSCM	Fail-Safe Clock Monitor Delay(1)	—	500	900	µs	-40°C to +85°C

Note 1: These parameters are characterized but not tested in manufacturing.

2: Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.

3: These parameters are characterized, but are not tested in manufacturing.

FIGURE 22-5: TIMER1, 2 AND 3 EXTERNAL CLOCK TIMING CHARACTERISTICS



Note: Refer to Figure 22-1 for load conditions.

TABLE 22-22: TIMER1 EXTERNAL CLOCK TIMING REQUIREMENTS[1]

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[2]**		**Min**	**Typ**	**Max**	**Units**	**Conditions**
TA10	TTXH	TxCK High Time	Synchronous, no prescaler	0.5 TCY + 20	—	—	ns	Must also meet parameter TA15
			Synchronous, with prescaler	10	—	—	ns	
			Asynchronous	10	—	—	ns	
TA11	TTXL	TxCK Low Time	Synchronous, no prescaler	0.5 TCY + 20	—	—	ns	Must also meet parameter TA15
			Synchronous, with prescaler	10	—	—	ns	
			Asynchronous	10	—	—	ns	
TA15	TTXP	TxCK Input Period	Synchronous, no prescaler	TCY + 40	—	—	ns	—
			Synchronous, with prescaler	Greater of: 20 ns or (TCY + 40)/N	—	—	—	N = prescale value (1, 8, 64, 256)
			Asynchronous	20	—	—	ns	—
OS60	Ft1	SOSC1/T1CK Oscillator Input frequency Range (oscillator enabled by setting bit TCS (T1CON<1>))		DC	—	50	kHz	—
TA20	TCKEXTMRL	Delay from External TxCK Clock Edge to Timer Increment		0.5 TCY	—	1.5 TCY	—	—

Note 1: Timer1 is a Type A.

2: These parameters are characterized by similarity, but are not tested in manufacturing.

TABLE 22-23: TIMER2 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(1)		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.

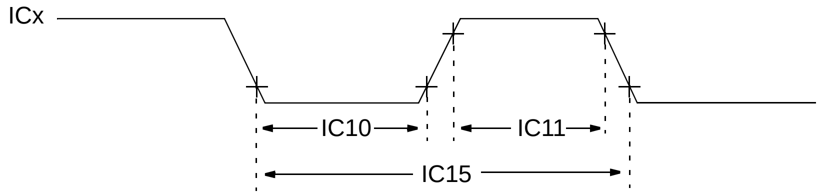
TABLE 22-24: TIMER3 EXTERNAL CLOCK TIMING REQUIREMENTS

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(1)		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	N = prescale value (1, 8, 64, 256)
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 22-6: INPUT CAPTURE (CAPx) TIMING CHARACTERISTICS



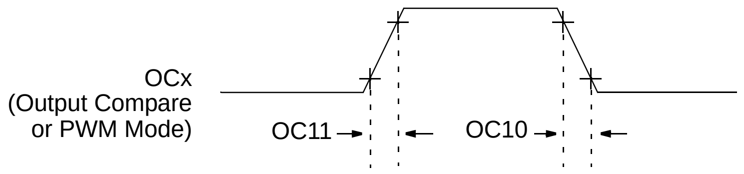
Note: Refer to Figure 22-1 for load conditions.

TABLE 22-25: 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(1)**		**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 22-7: OUTPUT COMPARE MODULE (OCx) TIMING CHARACTERISTICS



Note: Refer to Figure 22-1 for load conditions.

TABLE 22-26: 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(1)**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
OC10	TccF	OCx Output Fall Time	—	—	—	ns	See parameter D032
OC11	TccR	OCx Output Rise Time	—	—	—	ns	See parameter D031

Note 1: These parameters are characterized but not tested in manufacturing.

FIGURE 22-8: OC/PWM MODULE TIMING CHARACTERISTICS

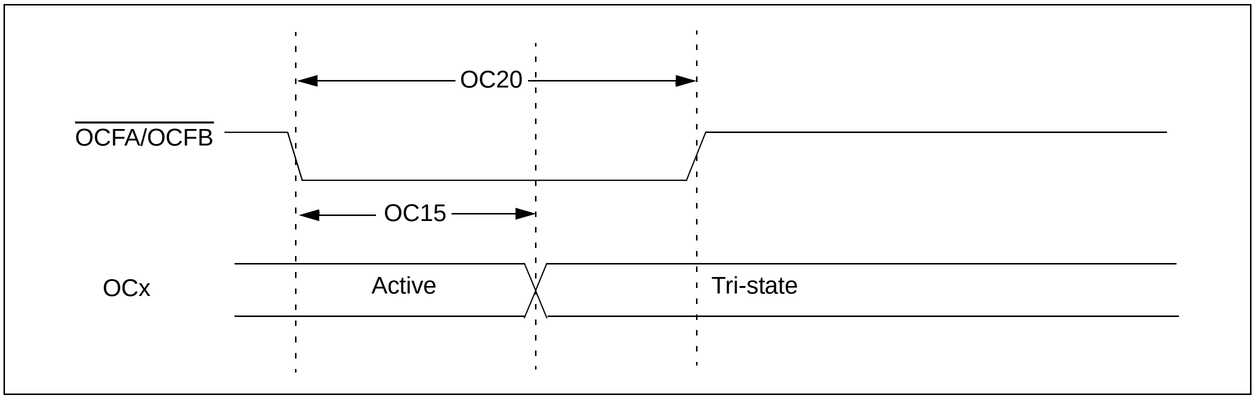


TABLE 22-27: SIMPLE OC/PWM MODE TIMING REQUIREMENTS

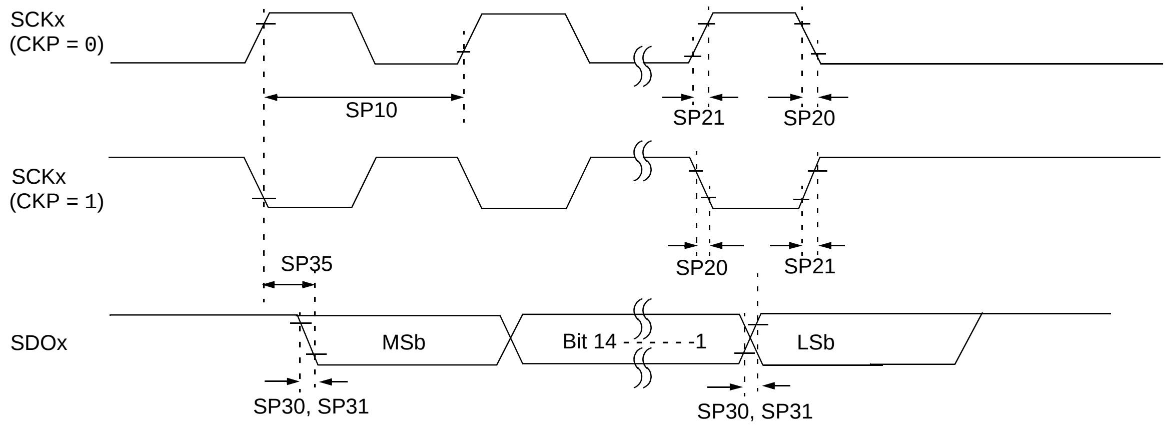
AC CHARACTERISTICS			**Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended				
Param No.	**Symbol**	**Characteristic[1]**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
OC15	TFD	Fault Input to PWM I/O Change	—	—	TCY + 20	ns	—
OC20	TFLT	Fault Input Pulse-Width	TCY + 20	—	—	ns	—

Note 1: These parameters are characterized but not tested in manufacturing.

TABLE 22-28: SPIx MAXIMUM DATA/CLOCK RATE SUMMARY

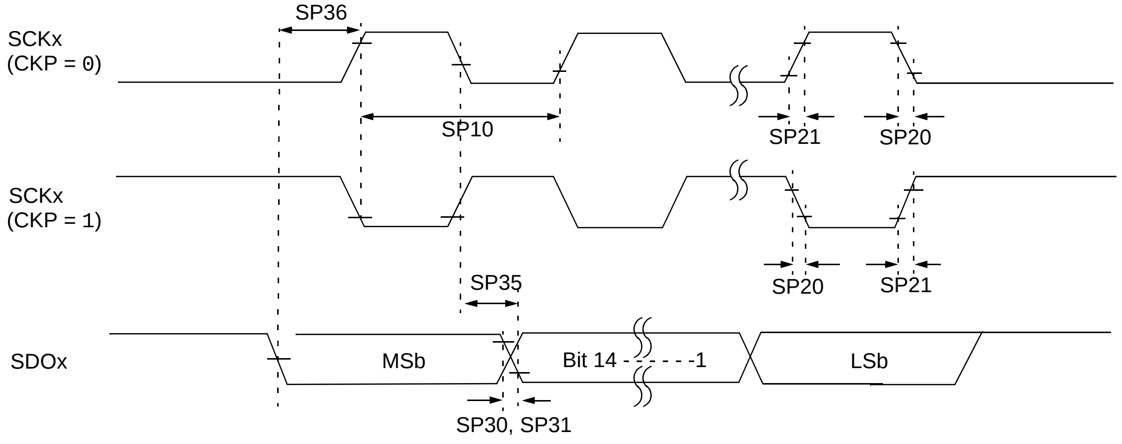
AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended			
Maximum Data Rate	**Master Transmit Only (Half-Duplex)**	**Master Transmit/Receive (Full-Duplex)**	**Slave Transmit/Receive (Full-Duplex)**	**CKE**	**CKP**	**SMP**
15 MHz	Table 22-29	—	—	0,1	0,1	0,1
9 MHz	—	Table 22-30	—	1	0,1	1
9 MHz	—	Table 22-31	—	0	0,1	1
15 MHz	—	—	Table 22-32	1	0	0
11 MHz	—	—	Table 22-33	1	1	0
15 MHz	—	—	Table 22-34	0	1	0
11 MHz	—	—	Table 22-35	0	0	0

FIGURE 22-9: SPIx MASTER MODE (HALF-DUPLEX, TRANSMIT ONLY CKE = 0) TIMING CHARACTERISTICS



Note: Refer to Figure 22-1 for load conditions.

FIGURE 22-10: SPIx MASTER MODE (HALF-DUPLEX, TRANSMIT ONLY CKE = 1) TIMING CHARACTERISTICS



Note: Refer to Figure 22-1 for load conditions.

TABLE 22-29: 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(1)**	**Min**	**Typ(2)**	**Max**	**Units**	**Conditions**
SP10	TscP	Maximum SCK Frequency	—	—	15	MHz	See **Note 3**
SP20	TscF	SCKx Output Fall Time	—	—	—	ns	See parameter DO32 and **Note 4**
SP21	TscR	SCKx Output Rise Time	—	—	—	ns	See parameter DO31 and **Note 4**
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See parameter DO32 and **Note 4**
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See parameter DO31 and **Note 4**
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	6	20	ns	—
SP36	TdiV2scH, TdiV2scL	SDOx Data Output Setup to First SCKx Edge	30	—	—	ns	—

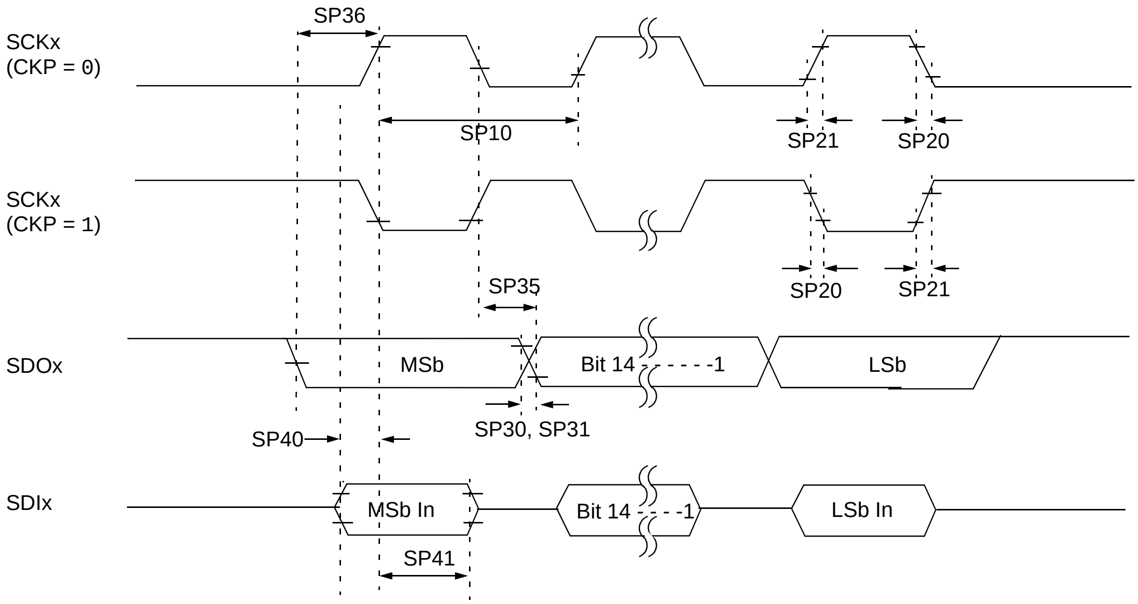
Note 1: These parameters are characterized, but are not tested in manufacturing.

2: Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.

3: The minimum clock period for SCKx is 66.7 ns. Therefore, the clock generated in Master mode must not violate this specification.

4: Assumes 50 pF load on all SPIx pins.

FIGURE 22-11: SPIx MASTER MODE (FULL-DUPLEX, CKE = 1, CKP = X, SMP = 1) TIMING CHARACTERISTICS



Note: Refer to Figure 22-1 for load conditions.

TABLE 22-30: 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°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended				
Param No.	**Symbol**	**Characteristic(1)**	**Min**	**Typ(2)**	**Max**	**Units**	**Conditions**
SP10	TscP	Maximum SCK Frequency	—	—	9	MHz	See **Note 3**
SP20	TscF	SCKx Output Fall Time	—	—	—	ns	See parameter DO32 and **Note 4**
SP21	TscR	SCKx Output Rise Time	—	—	—	ns	See parameter DO31 and **Note 4**
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See parameter DO32 and **Note 4**
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See parameter DO31 and **Note 4**
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	6	20	ns	—
SP36	TdoV2sc, TdoV2scL	SDOx Data Output Setup to First SCKx Edge	30	—	—	ns	—
SP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	30	—	—	ns	—
SP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	30	—	—	ns	—

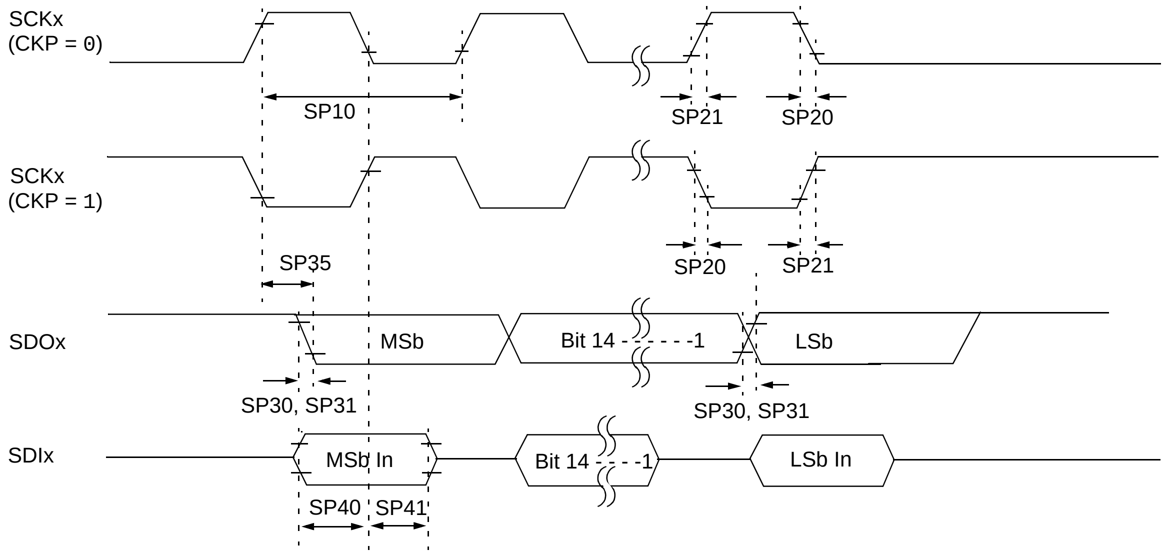
Note 1: These parameters are characterized, but are not tested in manufacturing.

2: Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.

3: The minimum clock period for SCKx is 111 ns. The clock generated in Master mode must not violate this specification.

4: Assumes 50 pF load on all SPIx pins.

FIGURE 22-12: SPIx MASTER MODE (FULL-DUPLEX, CKE = 0, CKP = X, SMP = 1) TIMING CHARACTERISTICS



Note: Refer to Figure 22-1 for load conditions.

TABLE 22-31: 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 ≤ T_A ≤ +85°C for Industrial -40°C ≤ T_A ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristic[1]**	**Min**	**Typ[2]**	**Max**	**Units**	**Conditions**
SP10	TscP	Maximum SCK Frequency	—	—	9	MHz	-40ºC to +125ºC and see **Note 3**
SP20	TscF	SCKx Output Fall Time	—	—	—	ns	See parameter DO32 and **Note 4**
SP21	TscR	SCKx Output Rise Time	—	—	—	ns	See parameter DO31 and **Note 4**
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See parameter DO32 and **Note 4**
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See parameter DO31 and **Note 4**
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	6	20	ns	—
SP36	TdoV2scH, TdoV2scL	SDOx Data Output Setup to First SCKx Edge	30	—	—	ns	—
SP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	30	—	—	ns	—
SP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	30	—	—	ns	—

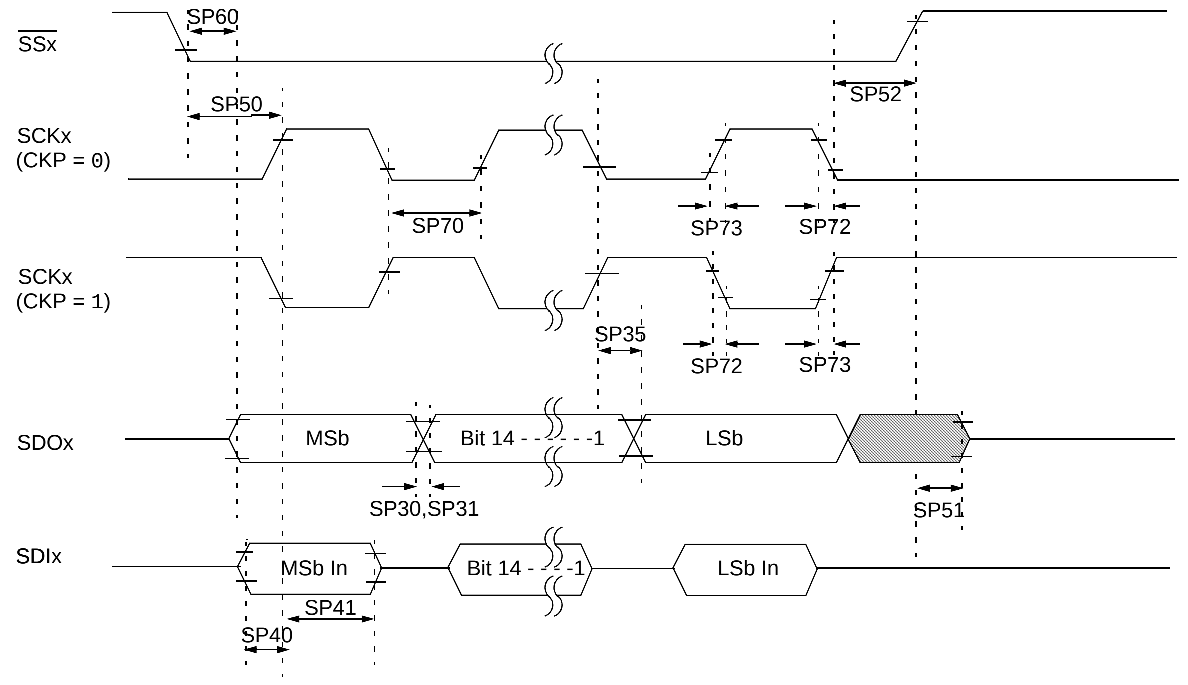
Note 1: These parameters are characterized, but are not tested in manufacturing.

2: Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.

3: The minimum clock period for SCKx is 111 ns. The clock generated in Master mode must not violate this specification.

4: Assumes 50 pF load on all SPIx pins.

FIGURE 22-13: SPIx SLAVE MODE (FULL-DUPLEX, CKE = `1`, CKP = `0`, SMP = `0`) TIMING CHARACTERISTICS



Note: Refer to Figure 22-1 for load conditions.

TABLE 22-32: 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 ≤ T_A ≤ +85°C for Industrial -40°C ≤ T_A ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristic[1]**	**Min**	**Typ[2]**	**Max**	**Units**	**Conditions**
SP70	TscP	Maximum SCK Input Frequency	—	—	15	MHz	See **Note 3**
SP72	TscF	SCKx Input Fall Time	—	—	—	ns	See parameter DO32 and **Note 4**
SP73	TscR	SCKx Input Rise Time	—	—	—	ns	See parameter DO31 and **Note 4**
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See parameter DO32 and **Note 4**
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See parameter DO31 and **Note 4**
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	6	20	ns	—
SP36	TdoV2scH, TdoV2scL	SDOx Data Output Setup to First SCKx Edge	30	—	—	ns	—
SP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	30	—	—	ns	—
SP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	30	—	—	ns	—
SP50	TssL2scH, TssL2scL	$\overline{\text{SSx}}$ ↓ to SCKx ↑ or SCKx Input	120	—	—	ns	—
SP51	TssH2doZ	$\overline{\text{SSx}}$ ↑ to SDOx Output High-Impedance[4]	10	—	50	ns	—
SP52	TscH2ssH TscL2ssH	$\overline{\text{SSx}}$ after SCKx Edge	1.5 TCY + 40	—	—	ns	See **Note 4**
SP60	TssL2doV	SDOx Data Output Valid after $\overline{\text{SSx}}$ Edge	—	—	50	ns	—

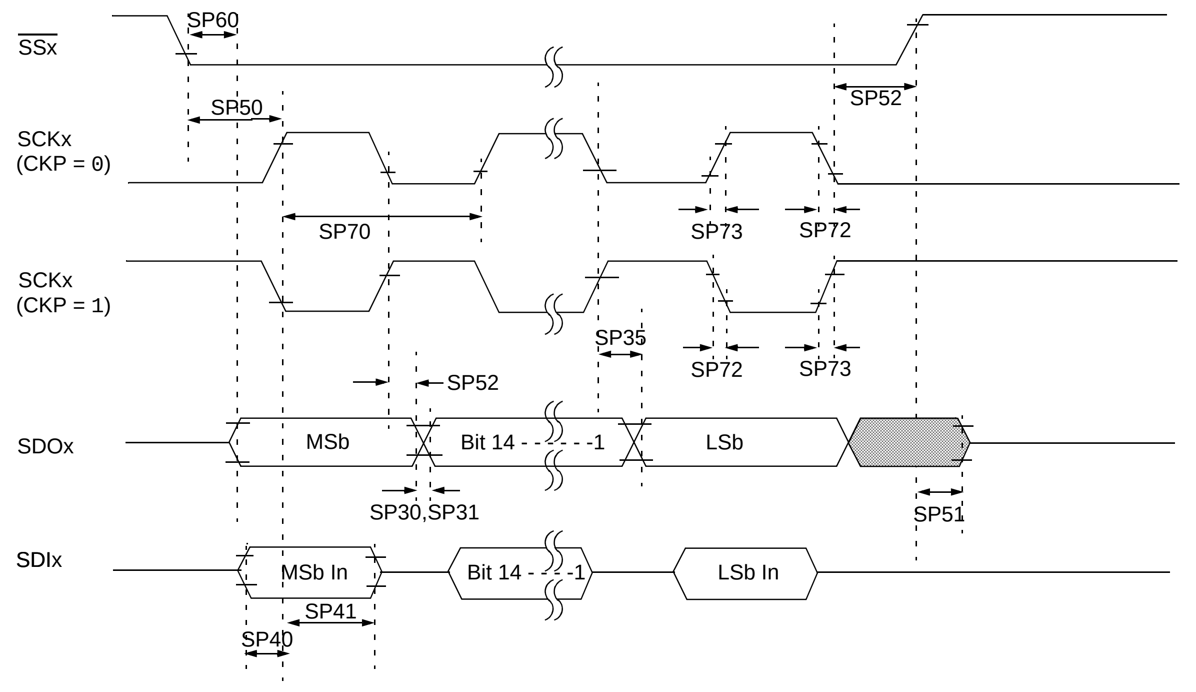
Note 1: These parameters are characterized, but are not tested in manufacturing.

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 22-14: SPIx SLAVE MODE (FULL-DUPLEX, CKE = 1, CKP = 1, SMP = 0) TIMING CHARACTERISTICS



Note: Refer to Figure 22-1 for load conditions.

TABLE 22-33: 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(1)**	**Min**	**Typ(2)**	**Max**	**Units**	**Conditions**
SP70	TscP	Maximum SCK Input Frequency	—	—	11	MHz	See **Note 3**
SP72	TscF	SCKx Input Fall Time	—	—	—	ns	See parameter DO32 and **Note 4**
SP73	TscR	SCKx Input Rise Time	—	—	—	ns	See parameter DO31 and **Note 4**
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See parameter DO32 and **Note 4**
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See parameter DO31 and **Note 4**
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	6	20	ns	—
SP36	TdoV2scH, TdoV2scL	SDOx Data Output Setup to First SCKx Edge	30	—	—	ns	—
SP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	30	—	—	ns	—
SP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	30	—	—	ns	—
SP50	TssL2scH, TssL2scL	$\overline{\text{SSx}}$ ↓ to SCKx ↑ or SCKx Input	120	—	—	ns	—
SP51	TssH2doZ	$\overline{\text{SSx}}$ ↑ to SDOx Output High-Impedance(4)	10	—	50	ns	—
SP52	TscH2ssH TscL2ssH	$\overline{\text{SSx}}$ after SCKx Edge	1.5 TCY + 40	—	—	ns	See **Note 4**
SP60	TssL2doV	SDOx Data Output Valid after $\overline{\text{SSx}}$ Edge	—	—	50	ns	—

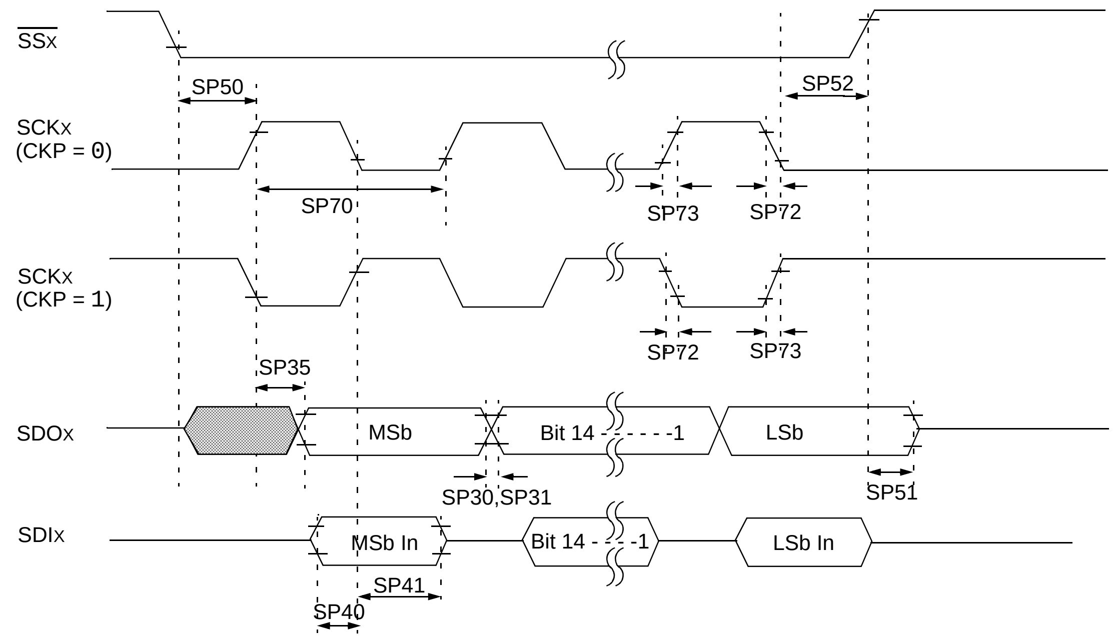
Note 1: These parameters are characterized, but are not tested in manufacturing.

2: Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.

3: The minimum clock period for SCKx is 91 ns. Therefore, the SCK clock generated by the Master must not violate this specification.

4: Assumes 50 pF load on all SPIx pins.

FIGURE 22-15: SPIx SLAVE MODE (FULL-DUPLEX CKE = 0, CKP = 1, SMP = 0) TIMING CHARACTERISTICS



Note: Refer to Figure 22-1 for load conditions.

TABLE 22-34: 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 ≤ T_A ≤ +85°C for Industrial -40°C ≤ T_A ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristic$^{(1)}$**	**Min**	**Typ$^{(2)}$**	**Max**	**Units**	**Conditions**
SP70	TscP	Maximum SCK Input Frequency	—	—	15	MHz	See **Note 3**
SP72	TscF	SCKx Input Fall Time	—	—	—	ns	See parameter DO32 and **Note 4**
SP73	TscR	SCKx Input Rise Time	—	—	—	ns	See parameter DO31 and **Note 4**
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See parameter DO32 and **Note 4**
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See parameter DO31 and **Note 4**
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	6	20	ns	—
SP36	TdoV2scH, TdoV2scL	SDOx Data Output Setup to First SCKx Edge	30	—	—	ns	—
SP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	30	—	—	ns	—
SP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	30	—	—	ns	—
SP50	TssL2scH, TssL2scL	$\overline{\text{SSx}}$ ↓ to SCKx ↑ or SCKx Input	120	—	—	ns	—
SP51	TssH2doZ	$\overline{\text{SSx}}$ ↑ to SDOx Output High-Impedance$^{(4)}$	10	—	50	ns	—
SP52	TscH2ssH TscL2ssH	$\overline{\text{SSx}}$ after SCKx Edge	1.5 T_{CY} + 40	—	—	ns	See **Note 4**

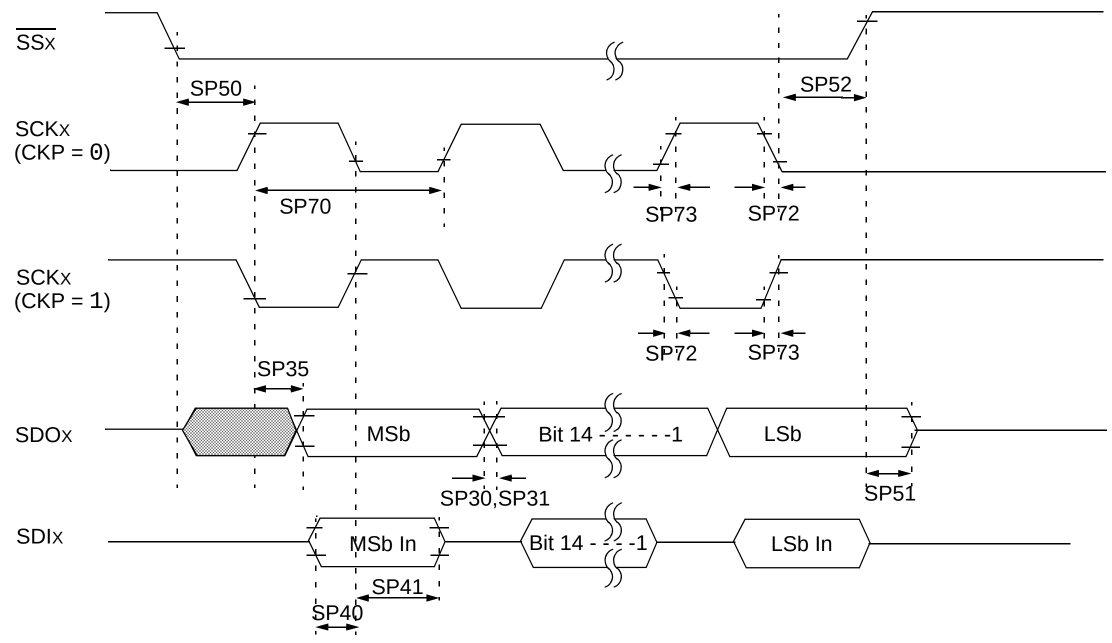
Note 1: These parameters are characterized, but are not tested in manufacturing.

2: Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.

3: The minimum clock period for SCKx is 66.7 ns. Therefore, the SCK clock generated by the Master must not violate this specification.

4: Assumes 50 pF load on all SPIx pins.

FIGURE 22-16: SPIx SLAVE MODE (FULL-DUPLEX, CKE = 0, CKP = 0, SMP = 0) TIMING CHARACTERISTICS



Note: Refer to Figure 22-1 for load conditions.

TABLE 22-35: 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[1]**	**Min**	**Typ[2]**	**Max**	**Units**	**Conditions**
SP70	TscP	Maximum SCK Input Frequency	—	—	11	MHz	See **Note 3**
SP72	TscF	SCKx Input Fall Time	—	—	—	ns	See parameter DO32 and **Note 4**
SP73	TscR	SCKx Input Rise Time	—	—	—	ns	See parameter DO31 and **Note 4**
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See parameter DO32 and **Note 4**
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See parameter DO31 and **Note 4**
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	6	20	ns	—
SP36	TdoV2scH, TdoV2scL	SDOx Data Output Setup to First SCKx Edge	30	—	—	ns	—
SP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	30	—	—	ns	—
SP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	30	—	—	ns	—
SP50	TssL2scH, TssL2scL	$\overline{\text{SSx}}$ ↓ to SCKx ↑ or SCKx Input	120	—	—	ns	—
SP51	TssH2doZ	$\overline{\text{SSx}}$ ↑ to SDOx Output High-Impedance[4]	10	—	50	ns	—
SP52	TscH2ssH TscL2ssH	$\overline{\text{SSx}}$ after SCKx Edge	1.5 TCY + 40	—	—	ns	See **Note 4**

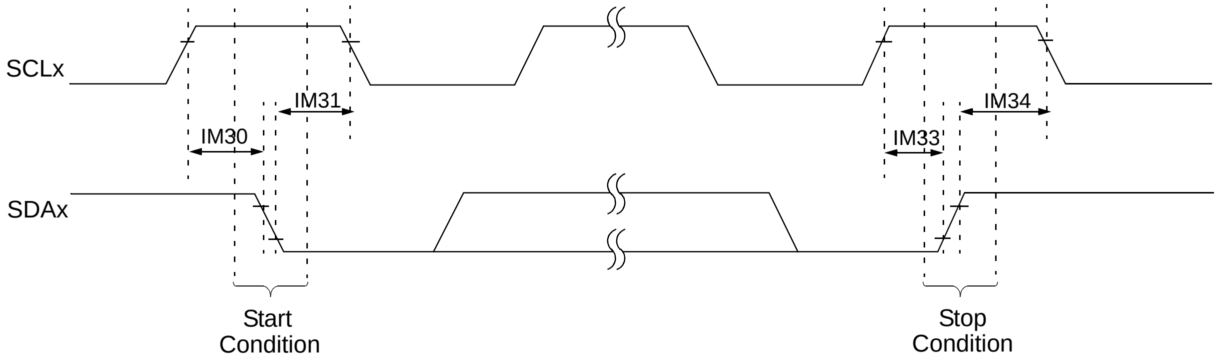
Note 1: These parameters are characterized, but are not tested in manufacturing.

2: Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.

3: The minimum clock period for SCKx is 91 ns. Therefore, the SCK clock generated by the Master must not violate this specification.

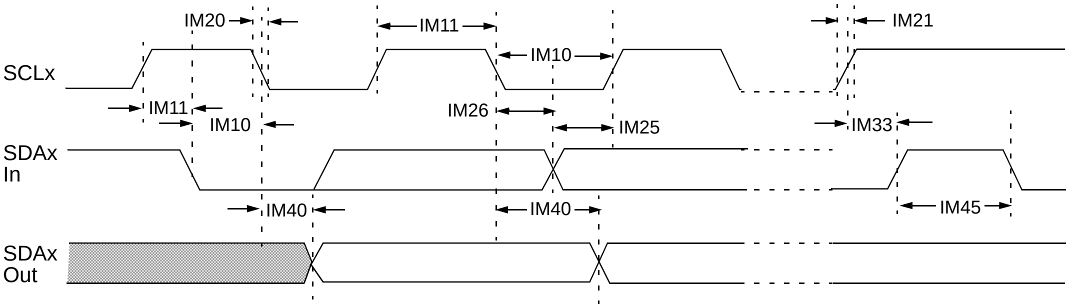
4: Assumes 50 pF load on all SPIx pins.

FIGURE 22-17: I2Cx BUS START/STOP BITS TIMING CHARACTERISTICS (MASTER MODE)



Note: Refer to Figure 22-1 for load conditions.

FIGURE 22-18: I2Cx BUS DATA TIMING CHARACTERISTICS (MASTER MODE)



Note: Refer to Figure 22-1 for load conditions.

TABLE 22-36: I2Cx BUS DATA TIMING REQUIREMENTS (MASTER MODE)

AC CHARACTERISTICS				**Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤ T_A ≤ +85°C for Industrial -40°C ≤ T_A ≤ +125°C for Extended			
Param No.	**Symbol**	**Characteristic[3]**		**Min[1]**	**Max**	**Units**	**Conditions**
IM10	$T_{LO:SCL}$	Clock Low Time	100 kHz mode	T_{CY}/2 (BRG + 1)	—	µs	—
			400 kHz mode	T_{CY}/2 (BRG + 1)	—	µs	—
			1 MHz mode[2]	T_{CY}/2 (BRG + 1)	—	µs	—
IM11	$T_{HI:SCL}$	Clock High Time	100 kHz mode	T_{CY}/2 (BRG + 1)	—	µs	—
			400 kHz mode	T_{CY}/2 (BRG + 1)	—	µs	—
			1 MHz mode[2]	T_{CY}/2 (BRG + 1)	—	µs	—
IM20	$T_{F:SCL}$	SDAx and SCLx Fall Time	100 kHz mode	—	300	ns	C_B is specified to be from 10 to 400 pF
			400 kHz mode	20 + 0.1 C_B	300	ns	
			1 MHz mode[2]	—	100	ns	
IM21	$T_{R:SCL}$	SDAx and SCLx Rise Time	100 kHz mode	—	1000	ns	C_B is specified to be from 10 to 400 pF
			400 kHz mode	20 + 0.1 C_B	300	ns	
			1 MHz mode[2]	—	300	ns	
IM25	$T_{SU:DAT}$	Data Input Setup Time	100 kHz mode	250	—	ns	—
			400 kHz mode	100	—	ns	
			1 MHz mode[2]	40	—	ns	
IM26	$T_{HD:DAT}$	Data Input Hold Time	100 kHz mode	0	—	µs	—
			400 kHz mode	0	0.9	µs	
			1 MHz mode[2]	0.2	—	µs	
IM30	$T_{SU:STA}$	Start Condition Setup Time	100 kHz mode	T_{CY}/2 (BRG + 1)	—	µs	Only relevant for Repeated Start condition
			400 kHz mode	T_{CY}/2 (BRG + 1)	—	µs	
			1 MHz mode[2]	T_{CY}/2 (BRG + 1)	—	µs	
IM31	$T_{HD:STA}$	Start Condition Hold Time	100 kHz mode	T_{CY}/2 (BRG + 1)	—	µs	After this period the first clock pulse is generated
			400 kHz mode	T_{CY}/2 (BRG + 1)	—	µs	
			1 MHz mode[2]	T_{CY}/2 (BRG + 1)	—	µs	
IM33	$T_{SU:STO}$	Stop Condition Setup Time	100 kHz mode	T_{CY}/2 (BRG + 1)	—	µs	—
			400 kHz mode	T_{CY}/2 (BRG + 1)	—	µs	
			1 MHz mode[2]	T_{CY}/2 (BRG + 1)	—	µs	
IM34	$T_{HD:STO}$	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[2]	T_{CY}/2 (BRG + 1)	—	ns	
IM40	$T_{AA:SCL}$	Output Valid From Clock	100 kHz mode	—	3500	ns	—
			400 kHz mode	—	1000	ns	—
			1 MHz mode[2]	—	400	ns	—
IM45	$T_{BF:SDA}$	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[2]	0.5	—	µs	
IM50	C_B	Bus Capacitive Loading		—	400	pF	—
IM51	T_{PGD}	Pulse Gobbler Delay		65	390	ns	**See Note 4**

Note 1: BRG is the value of the I^{2}C Baud Rate Generator. Refer to **Section 19. "Inter-Integrated Circuit (I^{2}C™)"** (DS70195) in the "*dsPIC33F/PIC24H Family Reference Manual".*

2: Maximum pin capacitance = 10 pF for all I2Cx pins (for 1 MHz mode only).

3: These parameters are characterized by similarity, but are not tested in manufacturing.

4: Typical value for this parameter is 130ns.

FIGURE 22-19: I2Cx BUS START/STOP BITS TIMING CHARACTERISTICS (SLAVE MODE)

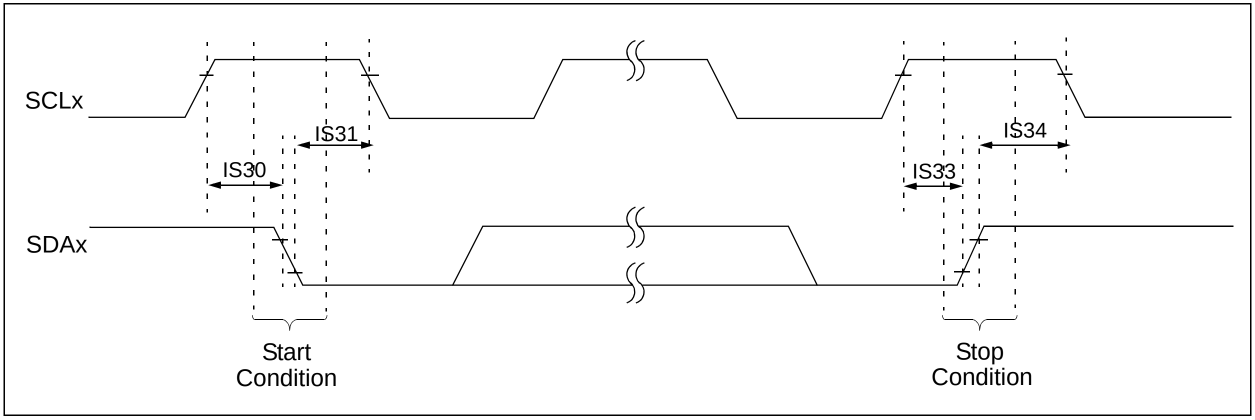


FIGURE 22-20: I2Cx BUS DATA TIMING CHARACTERISTICS (SLAVE MODE)

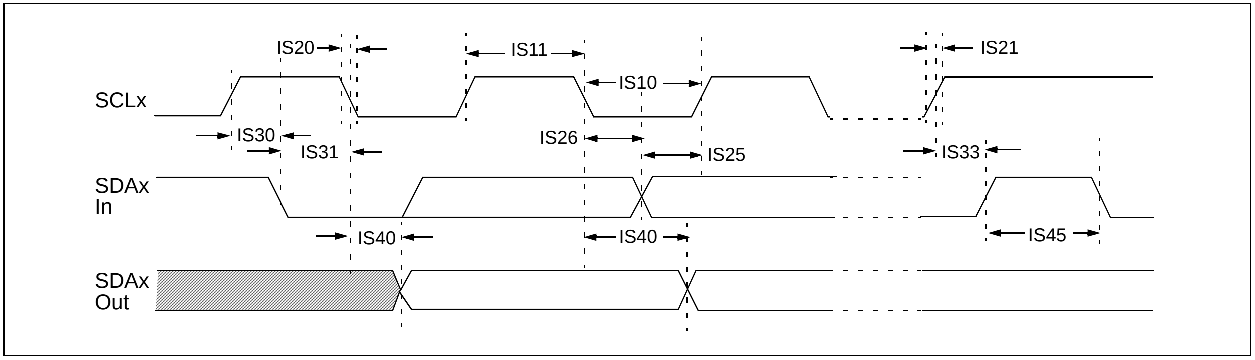


TABLE 22-37: I2Cx BUS DATA TIMING REQUIREMENTS (SLAVE MODE)

AC CHARACTERISTICS				**Standard Operating Conditions: 3.0V to 3.6V** **(unless otherwise stated)** Operating temperature -40°C ≤ T_A ≤ +85°C for Industrial -40°C ≤ T_A ≤ +125°C for Extended			
Param	**Symbol**	**Characteristic(2)**		**Min**	**Max**	**Units**	**Conditions**
IS10	$T_{LO:SCL}$	Clock Low Time	100 kHz mode	4.7	—	µs	Device must operate at a minimum of 1.5 MHz
			400 kHz mode	1.3	—	µs	Device must operate at a minimum of 10 MHz
			1 MHz mode(1)	0.5	—	µs	—
IS11	$T_{HI:SCL}$	Clock High Time	100 kHz mode	4.0	—	µs	Device must operate at a minimum of 1.5 MHz
			400 kHz mode	0.6	—	µs	Device must operate at a minimum of 10 MHz
			1 MHz mode(1)	0.5	—	µs	—
IS20	$T_{F:SCL}$	SDAx and SCLx Fall Time	100 kHz mode	—	300	ns	C_B is specified to be from 10 to 400 pF
			400 kHz mode	20 + 0.1 C_B	300	ns	
			1 MHz mode(1)	—	100	ns	
IS21	$T_{R:SCL}$	SDAx and SCLx Rise Time	100 kHz mode	—	1000	ns	C_B is specified to be from 10 to 400 pF
			400 kHz mode	20 + 0.1 C_B	300	ns	
			1 MHz mode(1)	—	300	ns	
IS25	$T_{SU:DAT}$	Data Input Setup Time	100 kHz mode	250	—	ns	—
			400 kHz mode	100	—	ns	
			1 MHz mode(1)	100	—	ns	
IS26	$T_{HD:DAT}$	Data Input Hold Time	100 kHz mode	0	—	µs	—
			400 kHz mode	0	0.9	µs	
			1 MHz mode(1)	0	0.3	µs	
IS30	$T_{SU:STA}$	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(1)	0.25	—	µs	
IS31	$T_{HD:STA}$	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(1)	0.25	—	µs	
IS33	$T_{SU:STO}$	Stop Condition Setup Time	100 kHz mode	4.7	—	µs	—
			400 kHz mode	0.6	—	µs	
			1 MHz mode(1)	0.6	—	µs	
IS34	$T_{HD:STO}$	Stop Condition Hold Time	100 kHz mode	4000	—	ns	—
			400 kHz mode	600	—	ns	
			1 MHz mode(1)	250	—	ns	
IS40	$T_{AA:SCL}$	Output Valid From Clock	100 kHz mode	0	3500	ns	—
			400 kHz mode	0	1000	ns	
			1 MHz mode(1)	0	350	ns	
IS45	$T_{BF:SDA}$	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(1)	0.5	—	µs	
IS50	C_B	Bus Capacitive Loading		—	400	pF	—

Note 1: Maximum pin capacitance = 10 pF for all I2Cx pins (for 1 MHz mode only).

2: These parameters are characterized by similarity, but not tested in manufacturing.

TABLE 22-38: ADC MODULE SPECIFICATIONS

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**
Device Supply							
AD01	AVDD	Module VDD Supply$^{(2)}$	Greater of VDD – 0.3 or 3.0	—	Lesser of VDD + 0.3 or 3.6	V	—
AD02	AVSS	Module VSS Supply$^{(2)}$	VSS – 0.3	—	VSS + 0.3	V	—
Reference Inputs							
AD05	VREFH	Reference Voltage High	AVSS + 2.5	—	AVDD	V	See **Note 1**
AD05a			3.0	—	3.6	V	VREFH = AVDD VREFL = AVSS = 0, see **Note 2**
AD06	VREFL	Reference Voltage Low	AVSS	—	AVDD – 2.5	V	See **Note 1**
AD06a			0	—	0	V	VREFH = AVDD VREFL = AVSS = 0, see **Note 2**
AD07	VREF	Absolute Reference Voltage$^{(2)}$	2.5	—	3.6	V	VREF = VREFH - VREFL
AD08	IREF	Current Drain	— —	250 —	550 10	µA µA	ADC operating, See **Note 1** ADC off, See **Note 1**
AD08a	IAD	Operating Current	— —	7.0 2.7	9.0 3.2	mA mA	10-bit ADC mode, See **Note 2** 12-bit ADC mode, See **Note 2**
Analog Input							
AD12	VINH	Input Voltage Range VINH$^{(2)}$	VINL	—	VREFH	V	This voltage reflects Sample and Hold Channels 0, 1, 2, and 3 (CH0-CH3), positive input
AD13	VINL	Input Voltage Range VINL$^{(2)}$	VREFL	—	AVSS + 1V	V	This voltage reflects Sample and Hold Channels 0, 1, 2, and 3 (CH0-CH3), negative input
AD17	RIN	Recommended Impedance of Analog Voltage Source$^{(3)}$	— —	— —	200 200	Ω Ω	10-bit ADC 12-bit ADC

Note 1: These parameters are not characterized or tested in manufacturing.

2: These parameters are characterized, but are not tested in manufacturing.

3: These parameters are assured by design, but are not characterized or tested in manufacturing.

TABLE 22-39: ADC MODULE SPECIFICATIONS (12-BIT 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 No.	**Symbol**	**Characteristic**	**Min.**	**Typ**	**Max.**	**Units**	**Conditions**
ADC Accuracy (12-bit Mode) – Measurements with external VREF+/VREF-(3)							
AD20a	Nr	Resolution(4)	12 data bits			bits	—
AD21a	INL	Integral Nonlinearity	-2	—	+2	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V
AD22a	DNL	Differential Nonlinearity	>-1	—	<1	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V
AD23a	GERR	Gain Error	—	3.4	10	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V
AD24a	EOFF	Offset Error	—	0.9	5	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V
AD25a	—	Monotonicity	—	—	—	—	Guaranteed(1)
ADC Accuracy (12-bit Mode) – Measurements with internal VREF+/VREF-(3)							
AD20a	Nr	Resolution(4)	12 data bits			bits	—
AD21a	INL	Integral Nonlinearity	-2	—	+2	LSb	VINL = AVSS = 0V, AVDD = 3.6V
AD22a	DNL	Differential Nonlinearity	>-1	—	<1	LSb	VINL = AVSS = 0V, AVDD = 3.6V
AD23a	GERR	Gain Error	—	10.5	20	LSb	VINL = AVSS = 0V, AVDD = 3.6V
AD24a	EOFF	Offset Error	—	3.8	10	LSb	VINL = AVSS = 0V, AVDD = 3.6V
AD25a	—	Monotonicity	—	—	—	—	Guaranteed(1)
Dynamic Performance (12-bit Mode)(2)							
AD30a	THD	Total Harmonic Distortion	—	—	-75	dB	—
AD31a	SINAD	Signal to Noise and Distortion	68.5	69.5	—	dB	—
AD32a	SFDR	Spurious Free Dynamic Range	80	—	—	dB	—
AD33a	FNYQ	Input Signal Bandwidth	—	—	250	kHz	—
AD34a	ENOB	Effective Number of Bits	11.09	11.3	—	bits	—

Note 1: The A/D conversion result never decreases with an increase in the input voltage, and has no missing codes.

2: These parameters are characterized by similarity, but are not tested in manufacturing.

3: These parameters are characterized, but are tested at 20 ksps only.

4: Injection currents > | 0 | can affect the ADC results by approximately 4-6 counts.

TABLE 22-40: ADC MODULE SPECIFICATIONS (10-BIT 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 No.	**Symbol**	**Characteristic**	**Min.**	**Typ**	**Max.**	**Units**	**Conditions**
ADC Accuracy (10-bit Mode) – Measurements with external VREF+/VREF-(3)							
AD20b	Nr	Resolution(4)	10 data bits			bits	—
AD21b	INL	Integral Nonlinearity	-1.5	—	+1.5	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V
AD22b	DNL	Differential Nonlinearity	>-1	—	<1	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V
AD23b	GERR	Gain Error	—	3	6	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V
AD24b	EOFF	Offset Error	—	2	5	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V
AD25b	—	Monotonicity	—	—	—	—	Guaranteed(1)
ADC Accuracy (10-bit Mode) – Measurements with internal VREF+/VREF-(3)							
AD20b	Nr	Resolution(4)	10 data bits			bits	—
AD21b	INL	Integral Nonlinearity	-1	—	+1	LSb	VINL = AVSS = 0V, AVDD = 3.6V
AD22b	DNL	Differential Nonlinearity	>-1	—	<1	LSb	VINL = AVSS = 0V, AVDD = 3.6V
AD23b	GERR	Gain Error	—	7	15	LSb	VINL = AVSS = 0V, AVDD = 3.6V
AD24b	EOFF	Offset Error	—	3	7	LSb	VINL = AVSS = 0V, AVDD = 3.6V
AD25b	—	Monotonicity	—	—	—	—	Guaranteed(1)
Dynamic Performance (10-bit Mode)(2)							
AD30b	THD	Total Harmonic Distortion	—	—	-64	dB	—
AD31b	SINAD	Signal to Noise and Distortion	57	58.5	—	dB	—
AD32b	SFDR	Spurious Free Dynamic Range	72	—	—	dB	—
AD33b	FNYQ	Input Signal Bandwidth	—	—	550	kHz	—
AD34b	ENOB	Effective Number of Bits	9.16	9.4	—	bits	—

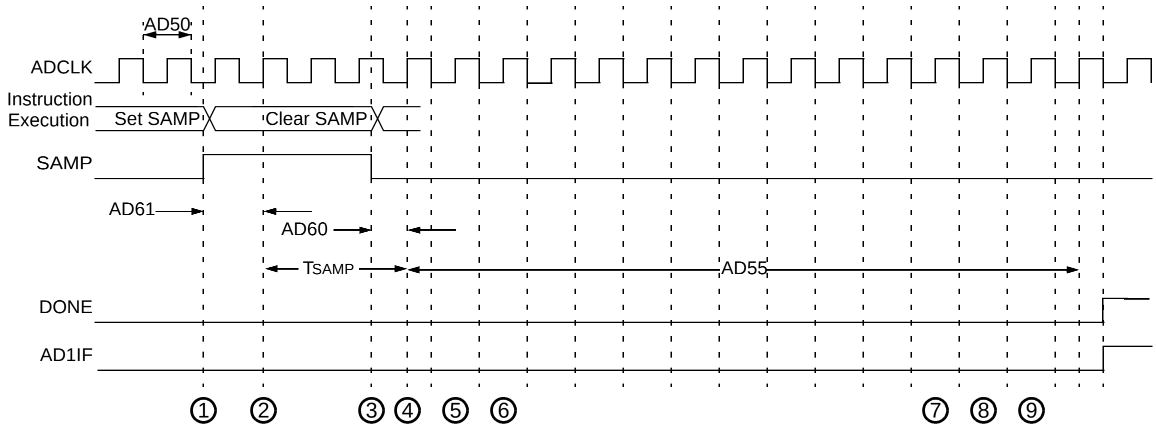
Note 1: The A/D conversion result never decreases with an increase in the input voltage, and has no missing codes.

2: These parameters are characterized by similarity, but are not tested in manufacturing.

3: These parameters are characterized, but are tested at 20 ksps only.

4: Injection currents > | 0 | can affect the ADC results by approximately 4-6 counts.

FIGURE 22-21: ADC CONVERSION (12-BIT MODE) TIMING CHARACTERISTICS (ASAM = `0`, SSRC<2:0> = `000`)



① – Software sets AD1CON. SAMP to start sampling.

② – Sampling starts after discharge period. TSAMP is described in **Section 16. “Analog-to-Digital Converter (ADC)”** (DS70183) in the *“dsPIC33F/PIC24H Family Reference Manual”*.

③ – Software clears AD1CON. SAMP to start conversion.

④ – Sampling ends, conversion sequence starts.

⑤ – Convert bit 11.

⑥ – Convert bit 10.

⑦ – Convert bit 1.

⑧ – Convert bit 0.

⑨ – One TAD for end of conversion.

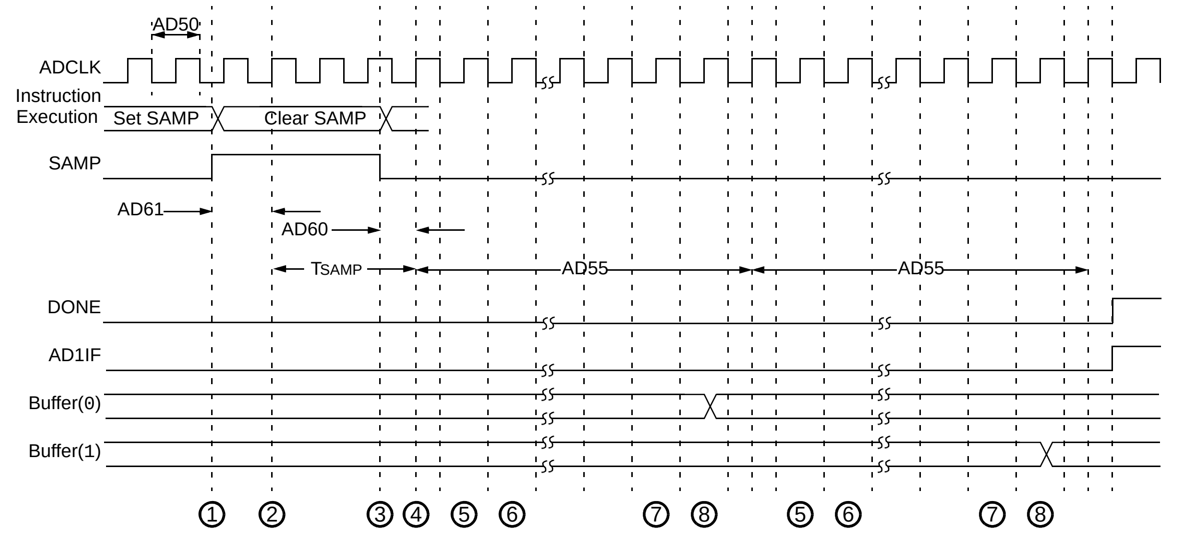
TABLE 22-41: ADC CONVERSION (12-BIT MODE) TIMING REQUIREMENTS

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**	**Min.**	**Typ**	**Max.**	**Units**	**Conditions**
Clock Parameters							
AD50	TAD	ADC Clock Period[(2)]	117.6	—	—	ns	—
AD51	tRC	ADC Internal RC Oscillator Period[(2)]	—	250	—	ns	—
Conversion Rate							
AD55	tCONV	Conversion Time[(2)]	—	14 TAD	—	ns	—
AD56	FCNV	Throughput Rate[(2)]	—	—	500	Ksps	—
AD57	TSAMP	Sample Time[(2)]	3.0 TAD	—	—	—	—
Timing Parameters							
AD60	tPCS	Conversion Start from Sample Trigger[(2)]	2.0 TAD	—	3.0 TAD	—	Auto Convert Trigger not selected
AD61	tPSS	Sample Start from Setting Sample (SAMP) bit[(2)]	2.0 TAD	—	3.0 TAD	—	—
AD62	tCSS	Conversion Completion to Sample Start (ASAM = `1`)[(2)]	—	0.5 TAD	—	—	—
AD63	tDPU	Time to Stabilize Analog Stage from ADC Off to ADC On[(2)]	—	—	20	µs	—

Note 1: Because the sample caps will eventually lose charge, clock rates below 10 kHz can affect linearity performance, especially at elevated temperatures.

2: These parameters are characterized but not tested in manufacturing.

FIGURE 22-22: ADC CONVERSION (10-BIT MODE) TIMING CHARACTERISTICS (CHPS<1:0> = `01`, SIMSAM = `0`, ASAM = `0`, SSRC<2:0> = `000`)



① – Software sets AD1CON. SAMP to start sampling.

② – Sampling starts after discharge period. TSAMP is described in **Section 16. "Analog-to-Digital Converter (ADC)"** (DS70183) in the *"dsPIC33F/PIC24H Family Reference Manual"*.

③ – Software clears AD1CON. SAMP to start conversion.

④ – Sampling ends, conversion sequence starts.

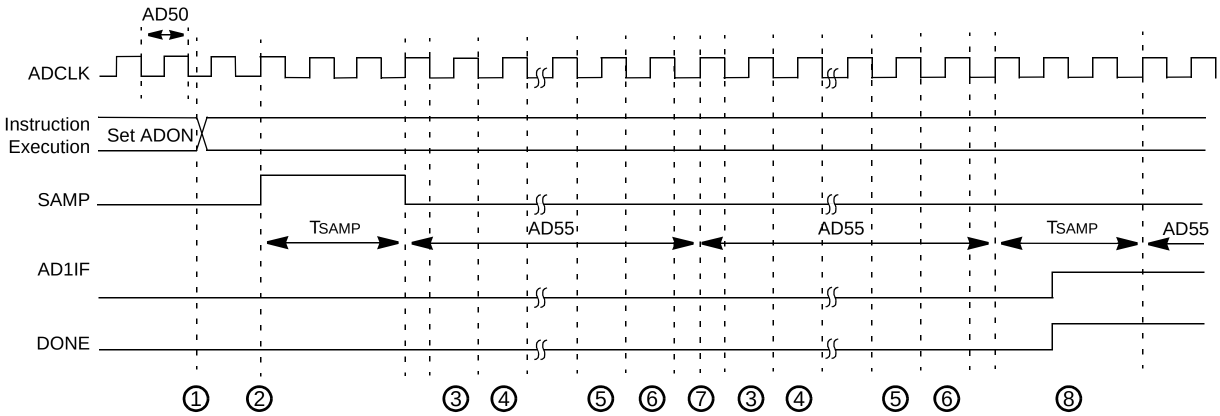
⑤ – Convert bit 9.

⑥ – Convert bit 8.

⑦ – Convert bit 0.

⑧ – One TAD for end of conversion.

FIGURE 22-23: ADC CONVERSION (10-BIT MODE) TIMING CHARACTERISTICS (CHPS<1:0> = `01`, SIMSAM = `0`, ASAM = `1`, SSRC<2:0> = `111`, SAMC<4:0> = `00001`)



① – Software sets AD1CON. ADON to start AD operation.

② – Sampling starts after discharge period. TSAMP is described in **Section 16. "Analog-to-Digital Converter (ADC)"** (DS70183) in the *"dsPIC33F/PIC24H Family Reference Manual".*

③ – Convert bit 9.

④ – Convert bit 8.

⑤ – Convert bit 0.

⑥ – One TAD for end of conversion.

⑦ – Begin conversion of next channel.

⑧ – Sample for time specified by SAMC<4:0>.

ADC CONVERSION (10-BIT MODE) 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**	**Min.**	**Typ[(1)]**	**Max.**	**Units**	**Conditions**
Clock Parameters							
AD50	TAD	ADC Clock Period[(1)]	76	—	—	ns	—
AD51	tRC	ADC Internal RC Oscillator Period[(1)]	—	250	—	ns	—
Conversion Rate							
AD55	tCONV	Conversion Time[(1)]	—	12 TAD	—	—	—
AD56	FCNV	Throughput Rate[(1)]	—	—	1.1	Msps	—
AD57	TSAMP	Sample Time[(1)]	2.0 TAD	—	—	—	—
Timing Parameters							
AD60	tPCS	Conversion Start from Sample Trigger[(1)]	2.0 TAD	—	3.0 TAD	—	Auto-Convert Trigger not selected
AD61	tPSS	Sample Start from Setting Sample (SAMP) bit[(1)]	2.0 TAD	—	3.0 TAD	—	—
AD62	tCSS	Conversion Completion to Sample Start (ASAM = 1)[(1)]	—	0.5 TAD	—	—	—
AD63	tDPU	Time to Stabilize Analog Stage from ADC Off to ADC On[(1)]	—	—	20	µs	—

Note 1: These parameters are characterized but not tested in manufacturing.

2: Because the sample caps will eventually lose charge, clock rates below 10 kHz can affect linearity performance, especially at elevated temperatures.

NOTES:

23.0 HIGH TEMPERATURE ELECTRICAL CHARACTERISTICS

This section provides an overview of dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 electrical characteristics for devices operating in an ambient temperature range of -40°C to +150°C.

Note: Programming of the Flash memory is not allowed above 125°C.

The specifications between -40°C to +150°C are identical to those shown in **Section 22.0 "Electrical Characteristics"** for operation between -40°C to +125°C, with the exception of the parameters listed in this section.

Parameters in this section begin with an H, which denotes High temperature. For example, parameter DC10 in **Section 22.0 "Electrical Characteristics"** is the Industrial and Extended temperature equivalent of HDC10.

Absolute maximum ratings for the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 high temperature devices are listed below. Exposure to these maximum rating conditions for extended periods can affect device reliability. Functional operation of the device at these or any other conditions above the parameters indicated in the operation listings of this specification is not implied.

Absolute Maximum Ratings(1)

Ambient temperature under bias(4)	-40°C to +150°C
Storage temperature	-65°C to +160°C
Voltage on VDD with respect to VSS	-0.3V to +4.0V
Voltage on any pin that is not 5V tolerant with respect to VSS(5)	-0.3V to (VDD + 0.3V)
Voltage on any 5V tolerant pin with respect to VSS when VDD < 3.0V(5)	-0.3V to 3.6V
Voltage on any 5V tolerant pin with respect to VSS when VDD ≥ 3.0V(5)	-0.3V to 5.6V
Maximum current out of VSS pin	60 mA
Maximum current into VDD pin(2)	60 mA
Maximum junction temperature	+155°C
Maximum output current sunk by any I/O pin(3)	1 mA
Maximum output current sourced by any I/O pin(3)	1 mA
Maximum current sunk by all ports combined	10 mA
Maximum current sourced by all ports combined(2)	10 mA

Note 1: Stresses above those listed under "Absolute Maximum Ratings" can cause permanent damage to the device. This is a stress rating only, and functional operation of the device at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure to maximum rating conditions for extended periods can affect device reliability.

2: Maximum allowable current is a function of device maximum power dissipation (see Table 23-2).

3: Unlike devices at 125°C and below, the specifications in this section also apply to the CLKOUT, VREF+, VREF-, SCLx, SDAx, PGCx, and PGDx pins.

4: AEC-Q100 reliability testing for devices intended to operate at 150°C is 1,000 hours. Any design in which the total operating time from 125°C to 150°C will be greater than 1,000 hours is not warranted without prior written approval from Microchip Technology Inc.

5: Refer to the **"Pin Diagrams"** section for 5V tolerant pins.

23.1 High Temperature DC Characteristics

TABLE 23-1: OPERATING MIPS VS. VOLTAGE

Characteristic	VDD Range (in Volts)	Temperature Range (in °C)	Max MIPS
			dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304
	3.0V to 3.6V	-40°C to +150°C	20

TABLE 23-2: THERMAL OPERATING CONDITIONS

Rating	Symbol	Min	Typ	Max	Unit
High Temperature Devices					
Operating Junction Temperature Range	TJ	-40	—	+155	°C
Operating Ambient Temperature Range	TA	-40	—	+150	°C
Power Dissipation: Internal chip power dissipation: PINT = VDD x (IDD - Σ IOH) I/O Pin Power Dissipation: I/O = Σ ({VDD - VOH} x IOH) + Σ (VOL x IOL)	PD	PINT + PI/O			W
Maximum Allowed Power Dissipation	PDMAX	(TJ - TA)/θJA			W

TABLE 23-3: DC TEMPERATURE AND VOLTAGE SPECIFICATIONS

DC CHARACTERISTICS			**Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤TA ≤+150°C for High Temperature				
Parameter No.	**Symbol**	**Characteristic**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
Operating Voltage							
HDC10	**Supply Voltage**						
	VDD	—	3.0	3.3	3.6	V	-40°C to +150°C

TABLE 23-4: DC CHARACTERISTICS: POWER-DOWN CURRENT (IPD)

DC CHARACTERISTICS			**Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤TA ≤+150°C for High Temperature			
Parameter No.	**Typical**	**Max**	**Units**	**Conditions**		
Power-Down Current (IPD)[3]						
HDC60e	250	2000	μA	+150°C	3.3V	Base Power-Down Current[1,3]
HDC61c	3	5	μA	+150°C	3.3V	Watchdog Timer Current: ΔIWDT[2,4]

Note 1: Base IPD is measured with all peripherals and clocks shut down. All I/Os are configured as inputs and pulled to VSS. WDT, etc., are all switched off, and VREGS (RCON<8>) = 1.

2: The Δ current is the additional current consumed when the module is enabled. This current should be added to the base IPD current.

3: These currents are measured on the device containing the most memory in this family.

4: These parameters are characterized, but are not tested in manufacturing.

TABLE 23-5: DC CHARACTERISTICS: DOZE CURRENT (I_{DD})

DC CHARACTERISTICS			**Standard Operating Conditions: 3.0V to 3.6V** **(unless otherwise stated)** Operating temperature -40°C ≤ T_A ≤ +150°C for High Temperature			
Parameter No.	**Typical[1]**	**Max**	**Units**	**Conditions**		
Operating Current (I_{PD})[1]						
HDC20	19	35	mA	+150°C	3.3V	10 MIPS
HDC21	27	45	mA	+150°C	3.3V	16 MIPS
HDC22	33	55	mA	+150°C	3.3V	20 MIPS

Note 1: These parameters are characterized, but are not tested in manufacturing.

TABLE 23-6: DC CHARACTERISTICS: DOZE CURRENT (I_{DOZE})

DC CHARACTERISTICS			**Standard Operating Conditions: 3.0V to 3.6V** **(unless otherwise stated)** Operating temperature -40°C ≤ T_A ≤ +150°C for High Temperature				
Parameter No.	**Typical[1]**	**Max**	**Doze Ratio**	**Units**	**Conditions**		
HDC72a	39	45	1:2	mA	+150°C	3.3V	20 MIPS
HDC72f	18	25	1:64	mA			
HDC72g	18	25	1:128	mA			

Note 1: Parameters with Doze ratios of 1:2 and 1:64 are characterized, but are not tested in manufacturing.

TABLE 23-7: DC CHARACTERISTICS: I/O PIN OUTPUT SPECIFICATIONS

DC CHARACTERISTICS			**Standard Operating Conditions: 3.0V to 3.6V** **(unless otherwise stated)** Operating temperature -40°C ≤ T_A ≤ +150°C for High Temperature				
Param No.	**Symbol**	**Characteristic**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
	V_{OL}	**Output Low Voltage**					
HDO10		I/O ports	—	—	0.4	V	I_{OL} = 1 mA, V_{DD} = 3.3V
HDO16		OSC2/CLKO	—	—	0.4	V	I_{OL} = 1 mA, V_{DD} = 3.3V
	V_{OH}	**Output High Voltage**					
HDO20		I/O ports	2.40	—	—	V	I_{OH} = -1 mA, V_{DD} = 3.3V
HDO26		OSC2/CLKO	2.41	—	—	V	I_{OH} = -1 mA, V_{DD} = 3.3V

TABLE 23-8: DC CHARACTERISTICS: PROGRAM MEMORY

DC CHARACTERISTICS			**Standard Operating Conditions: 3.0V to 3.6V** **(unless otherwise stated)** Operating temperature -40°C ≤ T_A ≤ +150°C for High Temperature				
Param No.	**Symbol**	**Characteristic**[1]	**Min**	**Typ**	**Max**	**Units**	**Conditions**
		Program Flash Memory					
HD130	E_P	Cell Endurance	10,000	—	—	E/W	-40° C to +150° C[2]
HD134	T_{RETD}	Characteristic Retention	20	—	—	Year	1000 E/W cycles or less and no other specifications are violated

Note 1: These parameters are assured by design, but are not characterized or tested in manufacturing.

2: Programming of the Flash memory is not allowed above 125°C.

23.2 AC Characteristics and Timing Parameters

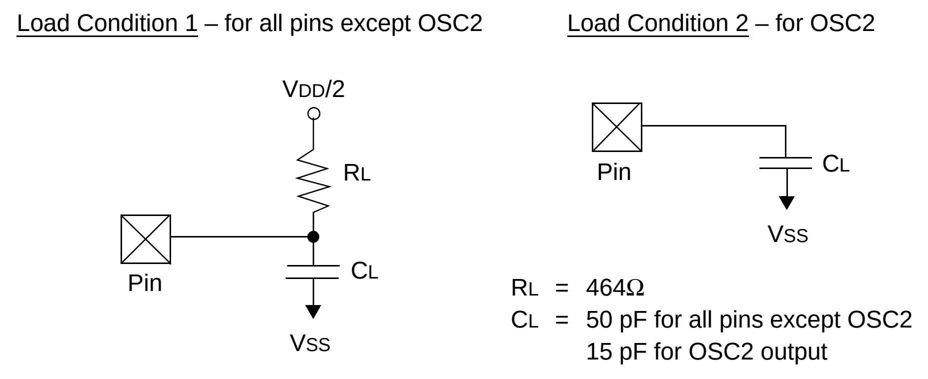
The information contained in this section defines dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 AC characteristics and timing parameters for high temperature devices. However, all AC timing specifications in this section are the same as those in **Section 22.2 “AC Characteristics and Timing Parameters”**, with the exception of the parameters listed in this section.

Parameters in this section begin with an H, which denotes High temperature. For example, parameter OS53 in **Section 22.2 “AC Characteristics and Timing Parameters”** is the Industrial and Extended temperature equivalent of HOS53.

TABLE 23-9: TEMPERATURE AND VOLTAGE SPECIFICATIONS – AC

AC CHARACTERISTICS	**Standard Operating Conditions: 3.0V to 3.6V** **(unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +150°C for High Temperature Operating voltage VDD range as described in Table 23-1.

FIGURE 23-1: LOAD CONDITIONS FOR DEVICE TIMING SPECIFICATIONS



RL = 464Ω
CL = 50 pF for all pins except OSC2
15 pF for OSC2 output

TABLE 23-10: PLL CLOCK TIMING SPECIFICATIONS

AC CHARACTERISTICS		**Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +150°C for High Temperature					
Param No.	**Symbol**	**Characteristic**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
HOS53	DCLK	CLKO Stability (Jitter)[(1)]	-5	0.5	5	%	Measured over 100 ms period

Note 1: 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:

$$\text{Peripheral Clock Jitter} = \frac{D_{CLK}}{\sqrt{\left(\frac{F_{OSC}}{\text{Peripheral Bit Rate Clock}}\right)}}$$

For example: Fosc = 32 MHz, DCLK = 5%, SPI bit rate clock, (i.e., SCK) is 2 MHz.

$$\text{SPI SCK Jitter} = \left[\frac{D_{CLK}}{\sqrt{\left(\frac{32\ MHz}{2\ MHz}\right)}}\right] = \left[\frac{5\%}{\sqrt{16}}\right] = \left[\frac{5\%}{4}\right] = 1.25\%$$

TABLE 23-11: SPIx MASTER MODE (CKE = 0) TIMING REQUIREMENTS

AC CHARACTERISTICS		**Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤TA ≤+150°C for High Temperature					
Param No.	**Symbol**	**Characteristic[1]**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
HSP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	10	25	ns	—
HSP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	28	—	—	ns	—
HSP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	35	—	—	ns	—

Note 1: These parameters are characterized but not tested in manufacturing.

TABLE 23-12: SPIx MODULE MASTER MODE (CKE = 1) TIMING REQUIREMENTS

AC CHARACTERISTICS		**Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤TA ≤+150°C for High Temperature					
Param No.	**Symbol**	**Characteristic[1]**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
HSP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	10	25	ns	—
HSP36	TdoV2sc, TdoV2scL	SDOx Data Output Setup to First SCKx Edge	35	—	—	ns	—
HSP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	28	—	—	ns	—
HSP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	35	—	—	ns	—

Note 1: These parameters are characterized but not tested in manufacturing.

TABLE 23-13: SPIx MODULE SLAVE MODE (CKE = 0) TIMING REQUIREMENTS

AC CHARACTERISTICS		**Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤TA ≤+150°C for High Temperature					
Param No.	**Symbol**	**Characteristic[(1)]**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
HSP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	—	35	ns	—
HSP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	25	—	—	ns	—
HSP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	25	—	—	ns	—
HSP51	TssH2doZ	$\overline{\text{SSx}}$ ↑ to SDOx Output High-Impedance	15	—	55	ns	See **Note 2**

Note 1: These parameters are characterized but not tested in manufacturing.

2: Assumes 50 pF load on all SPIx pins.

TABLE 23-14: SPIx MODULE SLAVE MODE (CKE = 1) TIMING REQUIREMENTS

AC CHARACTERISTICS		**Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤TA ≤+150°C for High Temperature					
Param No.	**Symbol**	**Characteristic[(1)]**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
HSP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	—	35	ns	—
HSP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	25	—	—	ns	—
HSP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	25	—	—	ns	—
HSP51	TssH2doZ	$\overline{\text{SSx}}$ ↑ to SDOx Output High-Impedance	15	—	55	ns	See **Note 2**
HSP60	TssL2doV	SDOx Data Output Valid after $\overline{\text{SSx}}$ Edge	—	—	55	ns	—

Note 1: These parameters are characterized but not tested in manufacturing.

2: Assumes 50 pF load on all SPIx pins.

TABLE 23-15: INTERNAL RC ACCURACY

AC CHARACTERISTICS		**Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤TA ≤+150°C for Extended					
Param No.	**Characteristic**	**Min**	**Typ**	**Max**	**Units**	**Conditions**	
HF21	**LPRC @ 32.768 kHz[(1,2)]**						
	LPRC	-70	—	+70	%	-40°C ≤TA ≤+150°C	VDD = 3.0-3.6V

Note 1: Change of LPRC frequency as VDD changes.

2: LPRC accuracy impacts the Watchdog Timer Time-out Period (TWDT1). See **Section 19.4 “Watchdog Timer (WDT)”** for more information.

TABLE 23-16: ADC MODULE SPECIFICATIONS

AC CHARACTERISTICS		**Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)** Operating temperature $-40°C \le T_A \le +150°C$ for High Temperature					
Param No.	**Symbol**	**Characteristic**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
Reference Inputs							
HAD08	I_{REF}	Current Drain	— —	250 —	600 50	μA μA	ADC operating, See **Note 1** ADC off, See **Note 1**

Note 1: These parameters are not characterized or tested in manufacturing.

2: These parameters are characterized, but are not tested in manufacturing.

TABLE 23-17: ADC MODULE SPECIFICATIONS (12-BIT MODE)(3)

AC CHARACTERISTICS		**Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)** Operating temperature $-40°C \le T_A \le +150°C$ for High Temperature					
Param No.	**Symbol**	**Characteristic**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
ADC Accuracy (12-bit Mode) – Measurements with External V_{REF+}/V_{REF-}(1)							
HAD20a	Nr	Resolution(3)	12 data bits			bits	—
HAD21a	INL	Integral Nonlinearity	-2	—	+2	LSb	$V_{INL} = AV_{SS} = V_{REFL} = 0V$, $AV_{DD} = V_{REFH} = 3.6V$
HAD22a	DNL	Differential Nonlinearity	> -1	—	< 1	LSb	$V_{INL} = AV_{SS} = V_{REFL} = 0V$, $AV_{DD} = V_{REFH} = 3.6V$
HAD23a	G_{ERR}	Gain Error	-2	—	10	LSb	$V_{INL} = AV_{SS} = V_{REFL} = 0V$, $AV_{DD} = V_{REFH} = 3.6V$
HAD24a	E_{OFF}	Offset Error	-3	—	5	LSb	$V_{INL} = AV_{SS} = V_{REFL} = 0V$, $AV_{DD} = V_{REFH} = 3.6V$
ADC Accuracy (12-bit Mode) – Measurements with Internal V_{REF+}/V_{REF-}(1)							
HAD20a	Nr	Resolution(3)	12 data bits			bits	—
HAD21a	INL	Integral Nonlinearity	-2	—	+2	LSb	$V_{INL} = AV_{SS} = 0V$, $AV_{DD} = 3.6V$
HAD22a	DNL	Differential Nonlinearity	> -1	—	< 1	LSb	$V_{INL} = AV_{SS} = 0V$, $AV_{DD} = 3.6V$
HAD23a	G_{ERR}	Gain Error	2	—	20	LSb	$V_{INL} = AV_{SS} = 0V$, $AV_{DD} = 3.6V$
HAD24a	E_{OFF}	Offset Error	2	—	10	LSb	$V_{INL} = AV_{SS} = 0V$, $AV_{DD} = 3.6V$
Dynamic Performance (12-bit Mode)(2)							
HAD33a	F_{NYQ}	Input Signal Bandwidth	—	—	200	kHz	—

Note 1: These parameters are characterized, but are tested at 20 ksps only.

2: These parameters are characterized by similarity, but are not tested in manufacturing.

3: Injection currents > | 0 | can affect the ADC results by approximately 4-6 counts.

TABLE 23-18: ADC MODULE SPECIFICATIONS (10-BIT MODE)(3)

AC CHARACTERISTICS		**Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤TA ≤+150°C for High Temperature					
Param No.	**Symbol**	**Characteristic**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
ADC Accuracy (10-bit Mode) – Measurements with External VREF+/VREF-(1)							
HAD20b	Nr	Resolution(3)	10 data bits			bits	—
HAD21b	INL	Integral Nonlinearity	-3	—	3	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V
HAD22b	DNL	Differential Nonlinearity	> -1	—	< 1	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V
HAD23b	GERR	Gain Error	-5	—	6	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V
HAD24b	EOFF	Offset Error	-1	—	5	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V
ADC Accuracy (10-bit Mode) – Measurements with Internal VREF+/VREF-(1)							
HAD20b	Nr	Resolution(3)	10 data bits			bits	—
HAD21b	INL	Integral Nonlinearity	-2	—	2	LSb	VINL = AVSS = 0V, AVDD = 3.6V
HAD22b	DNL	Differential Nonlinearity	> -1	—	< 1	LSb	VINL = AVSS = 0V, AVDD = 3.6V
HAD23b	GERR	Gain Error	-5	—	15	LSb	VINL = AVSS = 0V, AVDD = 3.6V
HAD24b	EOFF	Offset Error	-1.5	—	7	LSb	VINL = AVSS = 0V, AVDD = 3.6V
Dynamic Performance (10-bit Mode)(2)							
HAD33b	FNYQ	Input Signal Bandwidth	—	—	400	kHz	—

Note 1: These parameters are characterized, but are tested at 20 ksps only.

2: These parameters are characterized by similarity, but are not tested in manufacturing.

3: Injection currents > | 0 | can affect the ADC results by approximately 4-6 counts.

TABLE 23-19: ADC CONVERSION (12-BIT MODE) TIMING REQUIREMENTS

AC CHARACTERISTICS		**Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤TA ≤+150°C for High Temperature					
Param No.	**Symbol**	**Characteristic**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
Clock Parameters							
HAD50	TAD	ADC Clock Period[1]	147	—	—	ns	—
Conversion Rate							
HAD56	FCNV	Throughput Rate[1]	—	—	400	Ksps	—

Note 1: These parameters are characterized but not tested in manufacturing.

TABLE 23-20: ADC CONVERSION (10-BIT MODE) TIMING REQUIREMENTS

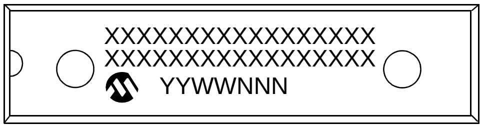
AC CHARACTERISTICS		**Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤TA ≤+150°C for High Temperature					
Param No.	**Symbol**	**Characteristic**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
Clock Parameters							
HAD50	TAD	ADC Clock Period[1]	104	—	—	ns	—
Conversion Rate							
HAD56	FCNV	Throughput Rate[1]	—	—	800	Ksps	—

Note 1: These parameters are characterized but not tested in manufacturing.

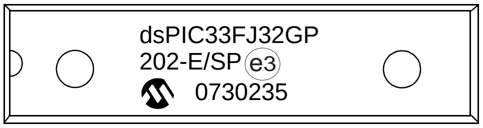
24.0 PACKAGING INFORMATION

24.1 Package Marking Information

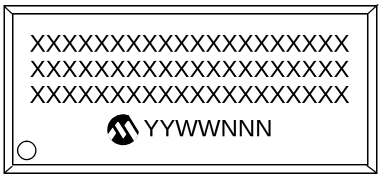
28-Lead SPDIP



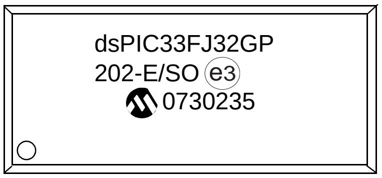
Example



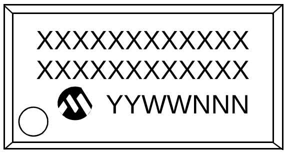
28-Lead SOIC



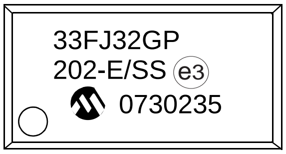
Example



28-Lead SSOP



Example



Legend:		
	XX...X	Customer-specific information
	Y	Year code (last digit of calendar year)
	YY	Year code (last 2 digits of calendar year)
	WW	Week code (week of January 1 is week ‘01’)
	NNN	Alphanumeric traceability code
	ⓔ3	Pb-free JEDEC designator for Matte Tin (Sn)
	*	This package is Pb-free. The Pb-free JEDEC designator (ⓔ3) can be found on the outer packaging for this package.

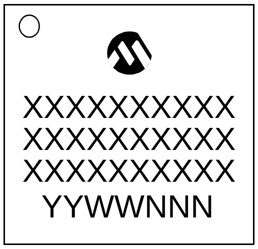
Note: If the full Microchip part number cannot be marked on one line, it is carried over to the next line, thus limiting the number of available characters for customer-specific information.

24.1 Package Marking Information (Continued)

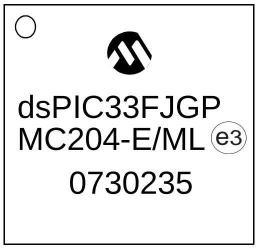
28-Lead QFN-S

Example

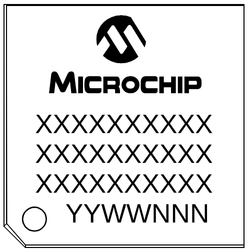
44-Lead QFN



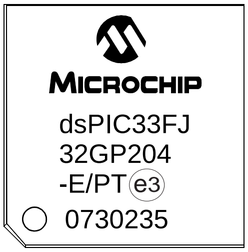
Example



44-Lead TQFP



Example



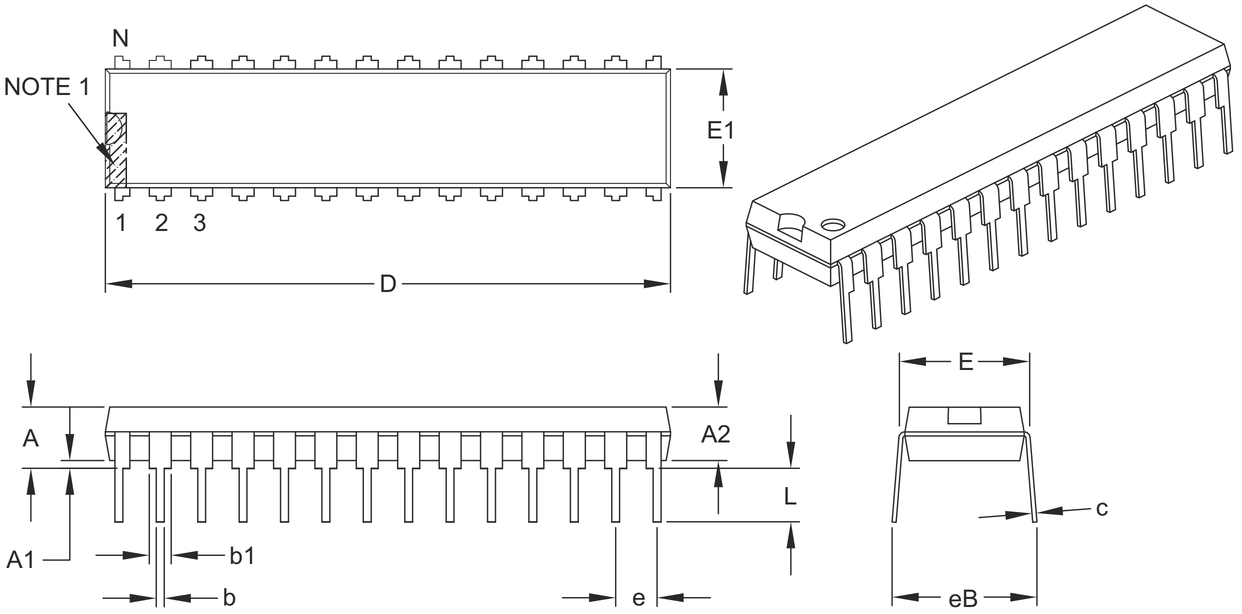
Legend:	XX...X	Customer-specific information
	Y	Year code (last digit of calendar year)
	YY	Year code (last 2 digits of calendar year)
	WW	Week code (week of January 1 is week ‘01’)
	NNN	Alphanumeric traceability code
	e3	Pb-free JEDEC designator for Matte Tin (Sn)
	*	This package is Pb-free. The Pb-free 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.

24.2 Package Details

28-Lead Skinny Plastic Dual In-Line (SP) – 300 mil Body [SPDIP]

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



	Units	INCHES		
	Dimension Limits	MIN	NOM	MAX
Number of Pins	N	28		
Pitch	e	.100 BSC		
Top to Seating Plane	A	–	–	.200
Molded Package Thickness	A2	.120	.135	.150
Base to Seating Plane	A1	.015	–	–
Shoulder to Shoulder Width	E	.290	.310	.335
Molded Package Width	E1	.240	.285	.295
Overall Length	D	1.345	1.365	1.400
Tip to Seating Plane	L	.110	.130	.150
Lead Thickness	c	.008	.010	.015
Upper Lead Width	b1	.040	.050	.070
Lower Lead Width	b	.014	.018	.022
Overall Row Spacing §	eB	–	–	.430

Notes:

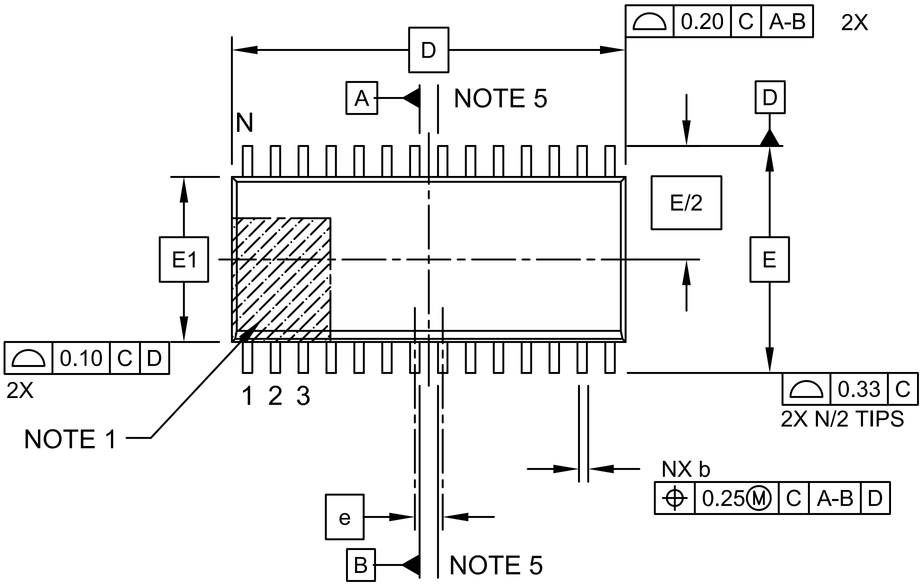
1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. § Significant Characteristic.
3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" per side.
4. Dimensioning and tolerancing per ASME Y14.5M.

   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

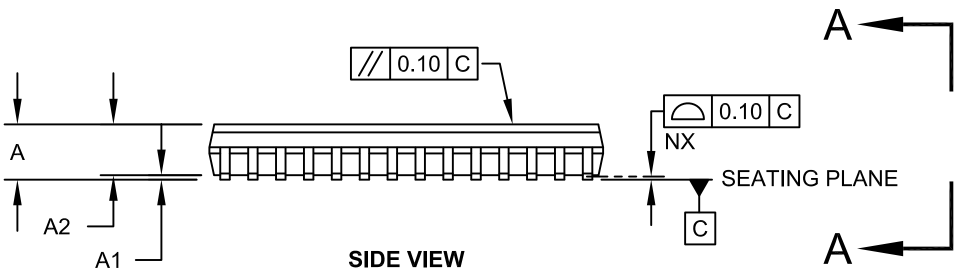
Microchip Technology Drawing C04-070B

28-Lead Plastic Small Outline (SO) - Wide, 7.50 mm Body [SOIC]

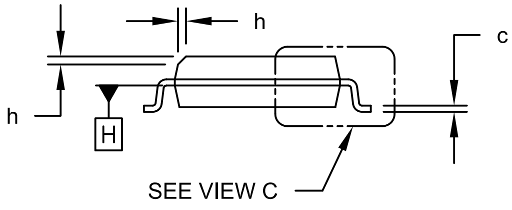
Note:	For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



TOP VIEW



SIDE VIEW

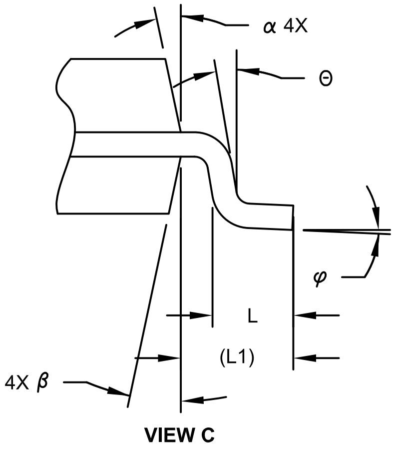


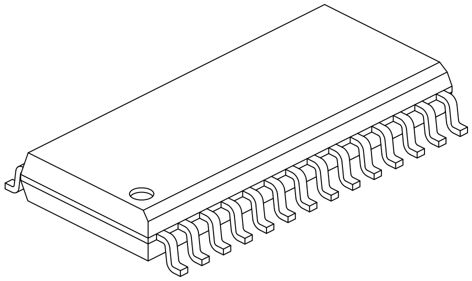
VIEW A-A

Microchip Technology Drawing C04-052C Sheet 1 of 2

28-Lead Plastic Small Outline (SO) - Wide, 7.50 mm Body [SOIC]

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





Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N	28		
Pitch	e	1.27 BSC		
Overall Height	A	-	-	2.65
Molded Package Thickness	A2	2.05	-	-
Standoff §	A1	0.10	-	0.30
Overall Width	E	10.30 BSC		
Molded Package Width	E1	7.50 BSC		
Overall Length	D	17.90 BSC		
Chamfer (Optional)	h	0.25	-	0.75
Foot Length	L	0.40	-	1.27
Footprint	L1	1.40 REF		
Lead Angle	Θ	0°	-	-
Foot Angle	φ	0°	-	8°
Lead Thickness	c	0.18	-	0.33
Lead Width	b	0.31	-	0.51
Mold Draft Angle Top	α	5°	-	15°
Mold Draft Angle Bottom	β	5°	-	15°

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. § Significant Characteristic
3. Dimension D does not include mold flash, protrusions or gate burrs, which shall not exceed 0.15 mm per end. Dimension E1 does not include interlead flash or protrusion, which shall not exceed 0.25 mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M

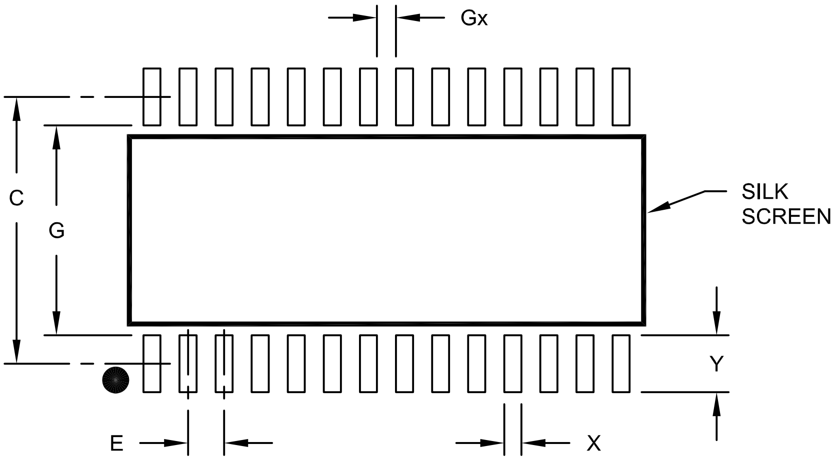
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

   REF: Reference Dimension, usually without tolerance, for information purposes only.
5. Datums A & B to be determined at Datum H.

Microchip Technology Drawing C04-052C Sheet 2 of 2

28-Lead Plastic Small Outline (SO) - Wide, 7.50 mm Body [SOIC]

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



RECOMMENDED LAND PATTERN

		Units	MILLIMETERS	
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E	1.27 BSC		
Contact Pad Spacing	C		9.40	
Contact Pad Width (X28)	X			0.60
Contact Pad Length (X28)	Y			2.00
Distance Between Pads	Gx	0.67		
Distance Between Pads	G	7.40		

Notes:

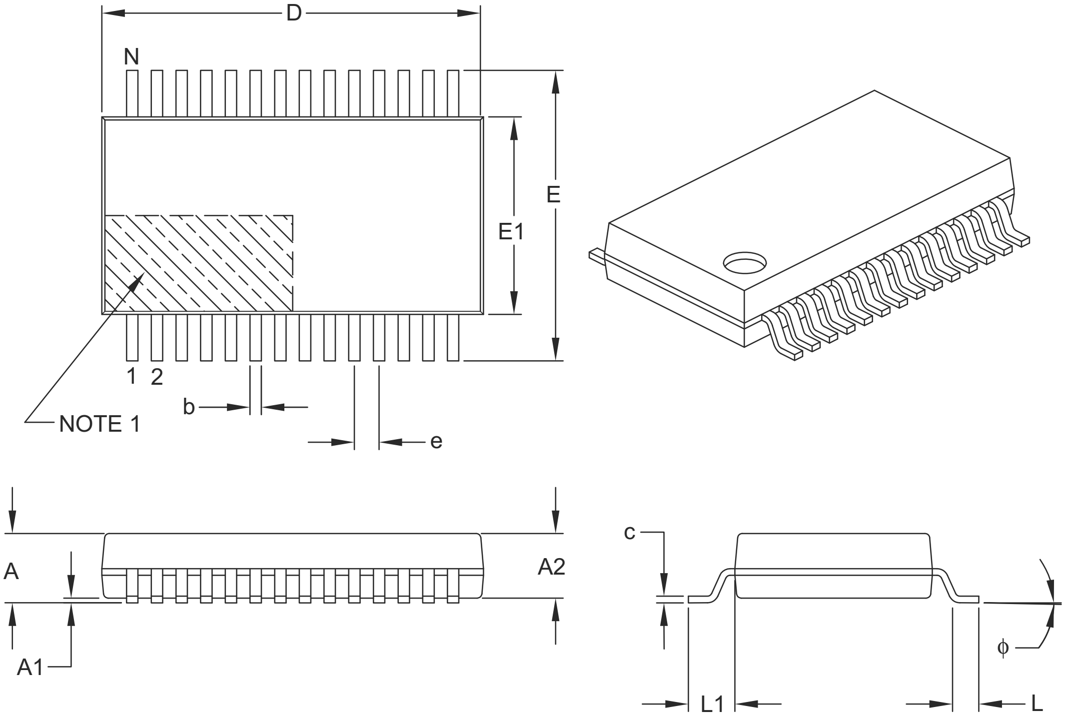
1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2052A

28-Lead Plastic Shrink Small Outline (SS) – 5.30 mm Body [SSOP]

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



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		28	
Pitch	e		0.65 BSC	
Overall Height	A	–	–	2.00
Molded Package Thickness	A2	1.65	1.75	1.85
Standoff	A1	0.05	–	–
Overall Width	E	7.40	7.80	8.20
Molded Package Width	E1	5.00	5.30	5.60
Overall Length	D	9.90	10.20	10.50
Foot Length	L	0.55	0.75	0.95
Footprint	L1		1.25 REF	
Lead Thickness	c	0.09	–	0.25
Foot Angle	ϕ	0°	4°	8°
Lead Width	b	0.22	–	0.38

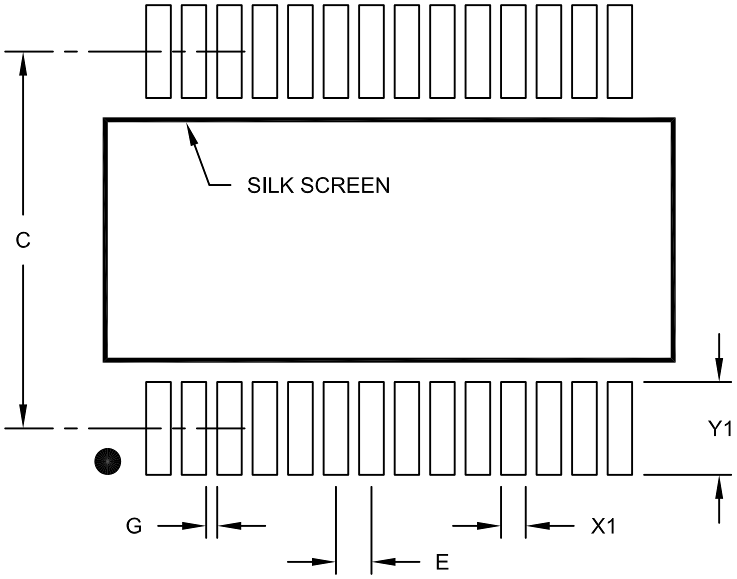
Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.20 mm per side.
3. Dimensioning and tolerancing per ASME Y14.5M.
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-073B

28-Lead Plastic Shrink Small Outline (SS) - 5.30 mm Body [SSOP]

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



RECOMMENDED LAND PATTERN

		Units	MILLIMETERS		
		Dimension Limits	MIN	NOM	MAX
Contact Pitch	E		0.65 BSC		
Contact Pad Spacing	C			7.20	
Contact Pad Width (X28)	X1				0.45
Contact Pad Length (X28)	Y1				1.75
Distance Between Pads	G		0.20		

Notes:

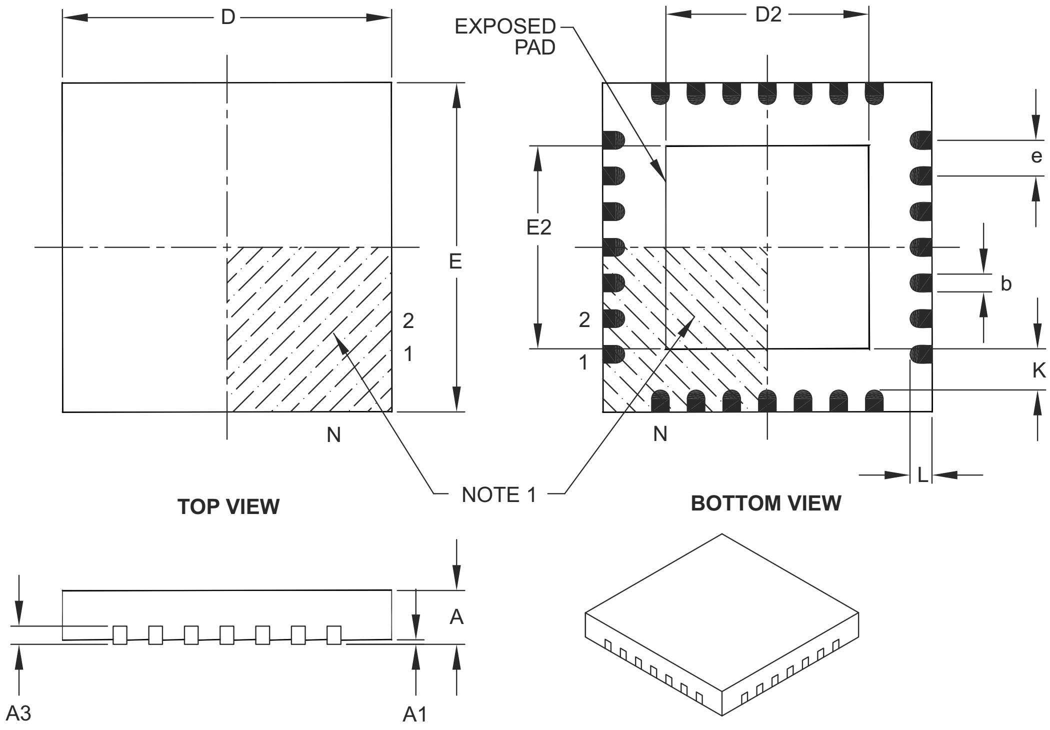
1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2073A

28-Lead Plastic Quad Flat, No Lead Package (MM) – 6x6x0.9 mm Body [QFN-S] with 0.40 mm Contact Length

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



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N	28		
Pitch	e	0.65 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	6.00 BSC		
Exposed Pad Width	E2	3.65	3.70	4.70
Overall Length	D	6.00 BSC		
Exposed Pad Length	D2	3.65	3.70	4.70
Contact Width	b	0.23	0.38	0.43
Contact Length	L	0.30	0.40	0.50
Contact-to-Exposed Pad	K	0.20	–	–

Notes:

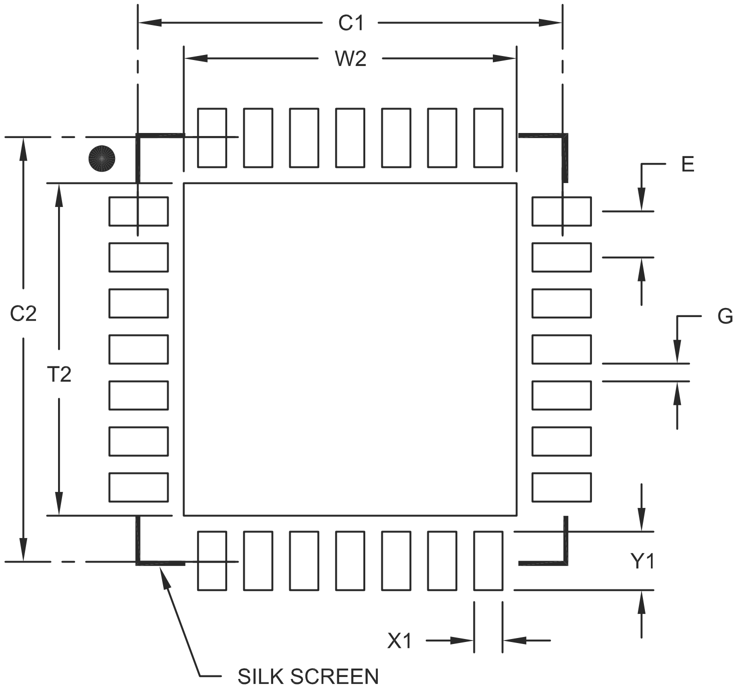
1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated.
3. Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-124B

28-Lead Plastic Quad Flat, No Lead Package (MM) – 6x6x0.9 mm Body [QFN-S] with 0.40 mm Contact Length

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



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E		0.65 BSC	
Optional Center Pad Width	W2			4.70
Optional Center Pad Length	T2			4.70
Contact Pad Spacing	C1		6.00	
Contact Pad Spacing	C2		6.00	
Contact Pad Width (X28)	X1			0.40
Contact Pad Length (X28)	Y1			0.85
Distance Between Pads	G	0.25		

Notes:

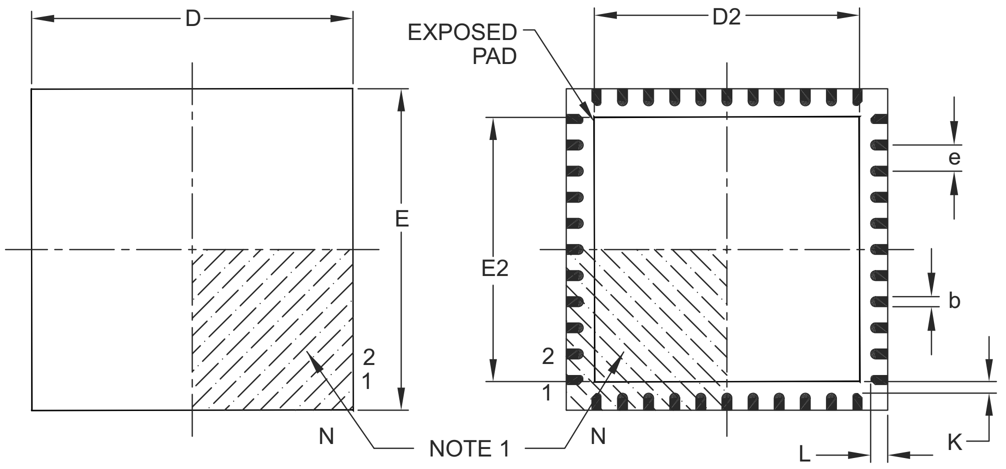
1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2124A

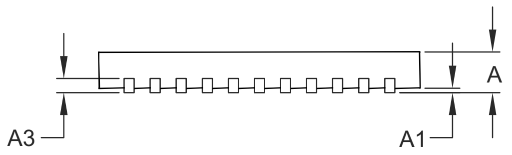
44-Lead Plastic Quad Flat, No Lead Package (ML) – 8x8 mm Body [QFN]

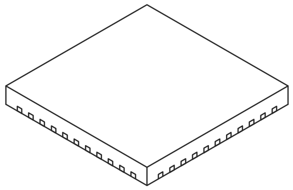
Note:	For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



TOP VIEW

BOTTOM VIEW





Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N	44		
Pitch	e	0.65 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	8.00 BSC		
Exposed Pad Width	E2	6.30	6.45	6.80
Overall Length	D	8.00 BSC		
Exposed Pad Length	D2	6.30	6.45	6.80
Contact Width	b	0.25	0.30	0.38
Contact Length	L	0.30	0.40	0.50
Contact-to-Exposed Pad	K	0.20	–	–

Notes:

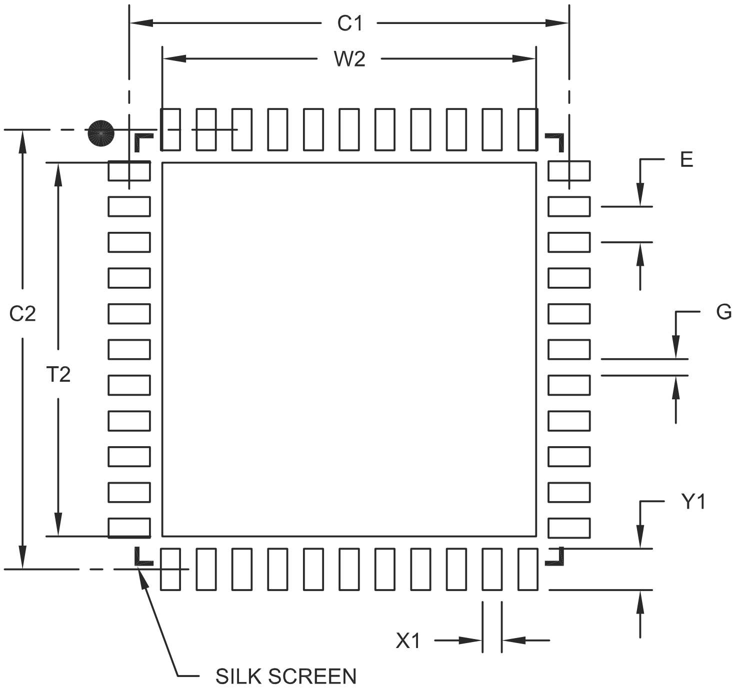
1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated.
3. Dimensioning and tolerancing per ASME Y14.5M.

   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-103B

44-Lead Plastic Quad Flat, No Lead Package (ML) – 8x8 mm Body [QFN]

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



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E	0.65 BSC		
Optional Center Pad Width	W2			6.80
Optional Center Pad Length	T2			6.80
Contact Pad Spacing	C1		8.00	
Contact Pad Spacing	C2		8.00	
Contact Pad Width (X44)	X1			0.35
Contact Pad Length (X44)	Y1			0.80
Distance Between Pads	G	0.25		

Notes:

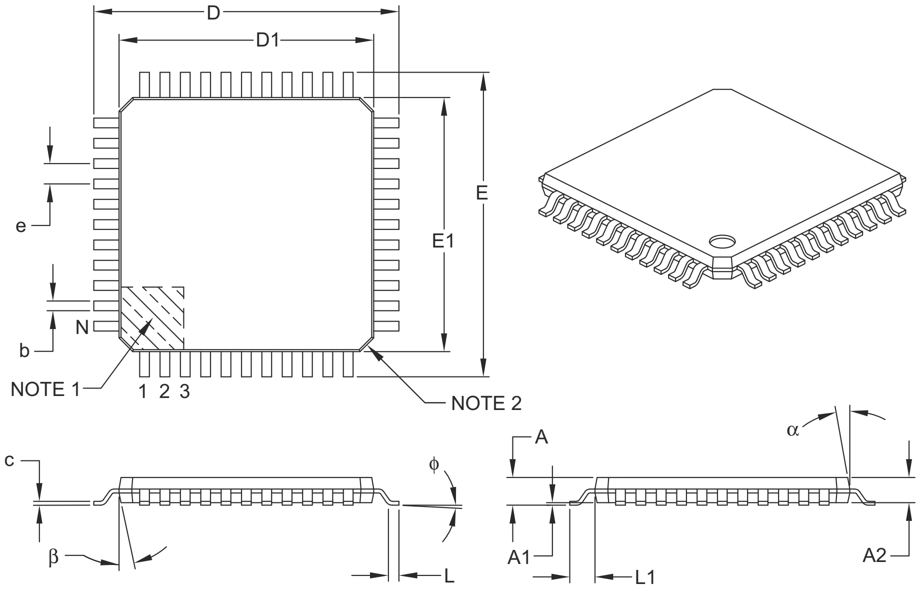
1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2103A

44-Lead Plastic Thin Quad Flatpack (PT) – 10x10x1 mm Body, 2.00 mm Footprint [TQFP]

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



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Leads	N		44	
Lead Pitch	e		0.80 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		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	0.30	0.37	0.45
Mold Draft Angle Top	α	11°	12°	13°
Mold Draft Angle Bottom	β	11°	12°	13°

Notes:

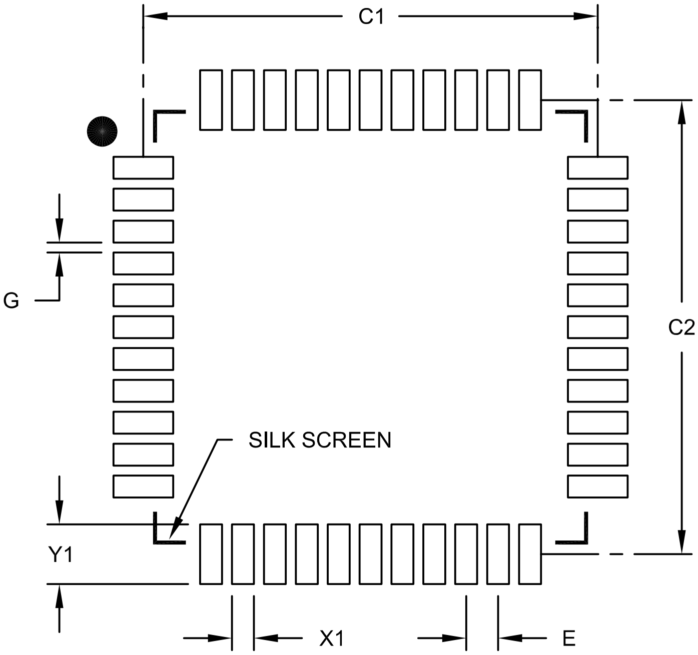
1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Chamfers at corners are optional; size may vary.
3. Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.25 mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-076B

44-Lead Plastic Thin Quad Flatpack (PT) 10X10X1 mm Body, 2.00 mm Footprint [TQFP]

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



RECOMMENDED LAND PATTERN

		Units	MILLIMETERS	
	Dimension Limits	MIN	NOM	MAX
Contact Pitch	E	0.80 BSC		
Contact Pad Spacing	C1		11.40	
Contact Pad Spacing	C2		11.40	
Contact Pad Width (X44)	X1			0.55
Contact Pad Length (X44)	Y1			1.50
Distance Between Pads	G	0.25		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2076B

APPENDIX A: REVISION HISTORY

Revision A (July 2007)

This is the initial released version of this document.

Revision B (June 2008)

This revision includes minor typographical and formatting changes throughout the data sheet text.

The major changes are referenced by their respective section in the following table.

TABLE A-1: MAJOR SECTION UPDATES

Section Name	Update Description
“High-Performance, 16-bit Digital Signal Controllers”	Added Extended Interrupts column to Remappable Peripherals in the Controller Families table and Note 2 (see Table 1). Added Note 1 to all pin diagrams, which references RPn pin usage by remappable peripherals (see **“Pin Diagrams”**).
Section 1.0 “Device Overview”	Changed PORTA pin name from RA15 to RA10 (see Table 1-1).
Section 3.0 “Memory Organization”	Added SFR definitions (ACCAL, ACCAH, ACCAU, ACCBL, ACCBH, and ACCBU) to the CPU Core Register Map (see Table 3-1). Updated Reset value for CORCON (see Table 3-1). Updated Reset values for the following SFRs: IPC1, IPC3-IPC5, IPC7, IPC16 and INTTREG (see Table 3-4). Updated the Reset value for CLKDIV in the System Control Register Map (see Table 3-20).
Section 6.0 “Resets”	Entire section was replaced to maintain consistency with other dsPIC33F data sheets.
Section 7.0 “Oscillator Configuration”	Removed the first sentence of the third clock source item (External Clock) in **Section 7.1.1.2 “Primary”**. Updated the default bit values for DOZE and FRCDIV in the Clock Divisor Register (see Register 7-2). Added the center frequency in the OSCTUN register for the FRC Tuning bits (TUN<5:0>) value `011111` and updated the center frequency for bits value `011110` (see Register 7-4).
Section 8.0 “Power-Saving Features”	Added the following two registers: • PMD1: Peripheral Module Disable Control Register 1 • PMD2: Peripheral Module Disable Control Register 2
Section 9.0 “I/O Ports”	Added paragraph and Table 9-1 to **Section 9.1.1 “Open-Drain Configuration”**, which provides details on I/O pins and their functionality. Removed the following sections, which are now available in the related section of the dsPIC33F/PIC24H Family Reference Manual: • 9.4.2 “Available Peripherals” • 9.4.3.3 “Mapping” • 9.4.5 “Considerations for Peripheral Pin Selection”
Section 13.0 “Output Compare”	Replaced sections 13.1, 13.2 and 13.3 and related figures and tables with entirely new content.

TABLE A-1: MAJOR SECTION UPDATES (CONTINUED)

Section Name	Update Description
Section 14.0 “Serial Peripheral Interface (SPI)”	Removed the following sections, which are now available in the related section of the dsPIC33F/PIC24H Family Reference Manual: • 14.1 “Interrupts” • 14.2 “Receive Operations” • 14.3 “Transmit Operations” • 14.4 “SPI Setup” (retained Figure 14-1: SPI Module Block Diagram)
Section 15.0 “Inter-Integrated Circuit (I^{2}C™)”	Removed the following sections, which are now available in the related section of the dsPIC33F/PIC24H Family Reference Manual: • 15.3 “I^{2}C Interrupts” • 15.4 “Baud Rate Generator” (retained Figure 15-1: I^{2}C Block Diagram) • 15.5 “I^{2}C Module Addresses” • 15.6 “Slave Address Masking” • 15.7 “IPMI Support” • 15.8 “General Call Address Support” • 15.9 “Automatic Clock Stretch” • 15.10 “Software Controlled Clock Stretching (STREN = 1)” • 15.11 “Slope Control” • 15.12 “Clock Arbitration” • 15.13 “Multi-Master Communication, Bus Collision, and Bus Arbitration” • 15.14 “Peripheral Pin Select Limitations”
Section 16.0 “Universal Asynchronous Receiver Transmitter (UART)”	Removed the following sections, which are now available in the related section of the dsPIC33F/PIC24H Family Reference Manual: • 16.1 “UART Baud Rate Generator” • 16.2 “Transmitting in 8-bit Data Mode” • 16.3 “Transmitting in 9-bit Data Mode” • 16.4 “Break and Sync Transmit Sequence” • 16.5 “Receiving in 8-bit or 9-bit Data Mode” • 16.6 “Flow Control Using $\overline{\text{UxCTS}}$ and $\overline{\text{UxRTS}}$ Pins” • 16.7 “Infrared Support” Removed IrDA references and Note 1, and updated the bit and bit value descriptions for UTXINV (UxSTA<14>) in the UARTx Status and Control Register (see Register 16-2).
Section 17.0 “10-bit/12-bit Analog-to-Digital Converter (ADC)”	Removed Equation 17-1: ADC Conversion Clock Period and Figure 17-2: ADC Transfer Function (10-Bit Example). Added ADC1 Module Block Diagram for dsPIC33FJ16GP304 and dsPIC33FJ32GP204 Devices (Figure 18-1) and ADC1 Module Block Diagram FOR dsPIC33FJ32GP202 Devices (Figure 17-2). Added Note 2 to Figure 17-3: ADC Conversion Clock Period Block Diagram. Added device-specific information to Note 1 in the ADC1 Input Scan Select Register Low (see Register 17-6), and updated the default bit value for bits 12-10 (CSS12-CSS10) from U-0 to R/W-0. Added device-specific information to Note 1 in the ADC1 Port Configuration Register Low (see Register 17-7), and updated the default bit value for bits 12-10 (PCFG12-PCFG10) from U-0 to R/W-0.

TABLE A-1: MAJOR SECTION UPDATES (CONTINUED)

Section Name	Update Description
Section 18.0 "Special Features"	Added FICD register information for address 0xF8000E in the Device Configuration Register Map (see Table 18-1). Added FICD register content (BKBUG, COE, JTAGEN, and ICS<1:0> to the dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304 Configuration Bits Description (see Table 18-2). Added a note regarding the placement of low-ESR capacitors, after the second paragraph of **Section 18.2 "On-Chip Voltage Regulator"** and to Figure 18-1. Removed the words "if enabled" from the second sentence in the fifth paragraph of **Section 18.3 "BOR: Brown-out Reset"**.
Section 21.0 "Electrical Characteristics"	Updated Max MIPS value for -40ºC to +125ºC temperature range in Operating MIPS vs. Voltage (see Table 21-1). Removed Typ value for parameter DC12 (see Table 22-4). Updated MIPS conditions for parameters DC24c, DC44c, DC72a, DC72f and DC72g (see Table 21-5, Table 21-6 and Table 21-8). Added Note 4 (reference to new table containing digital-only and analog pin information to I/O Pin Input Specifications (see Table 21-9). Updated Typ, Min, and Max values for Program Memory parameters D136, D137, and D138 (see Table 21-12). Updated Max value for Internal RC Accuracy parameter F21 for -40°C $\leq T_A \leq$ +125°C condition and added Note 2 (see Table 21-19). Removed all values for Reset, Watchdog Timer, Oscillator Start-up Timer, and Power-up Timer parameter SY20 and updated conditions, which now refers to **Section 18.4 "Watchdog Timer (WDT)"** and LPRC parameter F21a (see Table 21-21). Updated Min and Typ values for parameters AD60, AD61, AD62 and AD63 and removed Note 3 (see Table 21-37). Updated Min and Typ values for parameters AD60, AD61, AD62 and AD63 and removed Note 3 (see Table 21-38).

Revision C (December 2008)

This revision includes minor typographical and formatting changes throughout the data sheet text.

The major changes are referenced by their respective section in the following table.

TABLE A-2: MAJOR SECTION UPDATES

Section Name	Update Description
"High-Performance, 16-bit Digital Signal Controllers"	Updated all pin diagrams to denote the pin voltage tolerance (see **"Pin Diagrams"**).
Section 2.0 "Guidelines for Getting Started with 16-bit Digital Signal Controllers"	Added new section to the data sheet that provides guidelines on getting started with 16-bit Digital Signal Controllers.
Section 10.0 "I/O Ports"	Updated 5V tolerant status for I/O pin RB4 from Yes to No (see Table 10-1).
Section 22.0 "Electrical Characteristics"	Removed the maximum value for parameter DC12 (RAM Data Retention Voltage) in Table 22-4. Updated typical values for Operating Current (I_{DD}) and added Note 3 in Table 22-5. Updated typical and maximum values for Idle Current (I_{IDLE}): Core OFF Clock ON Base Current and added Note 3 in Table 22-6. Updated typical and maximum values for Power Down Current (I_{PD}) and added Note 5 in Table 22-7. Updated typical and maximum values for Doze Current (I_{DOZE}) and added Note 2 in Table 22-8. Added Note 3 to Table 22-12. Updated minimum value for Internal Voltage Regulator Specifications in Table 22-13. Added parameter OS42 (G_M) and Notes 4, 5, and 6 to Table 22-16. Added Notes 2 and 3 to Table 22-17. Added Note 2 to Table 22-20. Added Note 2 to Table 22-21. Added Note 2 to Table 22-22. Added Note 1 to Table 22-23. Added Note 1 to Table 22-24. Added Note 3 to Table 22-32. Added Note 2 to Table 22-33. Updated typical value for parameter AD08 (ADC in operation) and added Notes 2 and 3 in Table 22-34. Updated minimum, typical, and maximum values for parameters AD23a, AD24a, AD30a, AD32a, AD32a, and AD34a, and added Notes 2 and 3 in Table 22-35. Updated minimum, typical, and maximum values for parameters AD23b, AD24b, AD30b, AD32b, AD32b, and AD34b, and added Notes 2 and 3 in Table 22-36.

Revision D (October 2009)

This revision includes minor typographical and formatting changes throughout the data sheet text.

Global changes include:

- Changed all instances of OSCI to OSC1 and OSCO to OSC2.
- Changed all instances of PGCx/EMUCx and PGDx/EMUDx (where x = 1, 2 or 3) to PGECx and PGEDx.

Changed all instances of VDDCORE and VDDCORE/VCAP to VCAP/VDDCORE

All other major changes are referenced by their respective section in the following table.

TABLE A-3: MAJOR SECTION UPDATES

Section Name	Update Description
“High-Performance, 16-bit Digital Signal Controllers”	Added Note 2 to the 28-Pin QFN-S and 44-Pin QFN pin diagrams, which references pin connections to VSS.
Section 8.0 “Oscillator Configuration”	Updated the Oscillator System Diagram (see Figure 8-1). Added Note 1 to the Oscillator Tuning (OSCTUN) register (see Register 8-4).
Section 10.0 “I/O Ports”	Removed Table 10-1 and added reference to pin diagrams for I/O pin availability and functionality.
Section 15.0 “Serial Peripheral Interface (SPI)”	Added Note 2 to the SPIx Control Register 1 (see Register 15-2).
Section 17.0 “Universal Asynchronous Receiver Transmitter (UART)”	Updated the UTXINV bit settings in the UxSTA register and added Note 1 (see Register 17-2).
Section 22.0 “Electrical Characteristics”	Updated the Min value for parameter DC12 (RAM Retention Voltage) and added Note 4 to the DC Temperature and Voltage Specifications (see Table 22-4). Updated the Min value for parameter DI35 (see Table 22-20). Updated AD08 and added reference to Note 2 for parameters AD05a, AD06a and AD08a (see Table 22-34).

Revision E (November 2009)

The revision includes the following global update:

- Added Note 2 to the shaded table that appears at the beginning of each chapter. This new note provides information regarding the availability of registers and their associated bits

This revision also includes minor typographical and formatting changes throughout the data sheet text.

All other major changes are referenced by their respective section in the following table.

TABLE A-4: MAJOR SECTION UPDATES

Section Name	Update Description
“High-Performance, 16-bit Digital Signal Controllers”	Added information on high temperature operation (see **“Operating Range:”**).
Section 10.0 “I/O Ports”	Changed the reference to digital-only pins to 5V tolerant pins in the second paragraph of **Section 10.2 “Open-Drain Configuration”**.
Section 17.0 “Universal Asynchronous Receiver Transmitter (UART)”	Updated the two baud rate range features to: 10 Mbps to 38 bps at 40 MIPS.
Section 18.0 “10-bit/12-bit Analog-to-Digital Converter (ADC)”	Updated the ADC1 block diagrams (see Figure 18-1 and Figure 18-2).
Section 19.0 “Special Features”	Updated the second paragraph and removed the fourth paragraph in **Section 19.1 “Configuration Bits”**. Updated the Device Configuration Register Map (see Table 19-1).
Section 22.0 “Electrical Characteristics”	Updated the Absolute Maximum Ratings for high temperature and added Note 4. Updated the SPIx Module Slave Mode (CKE = 1) Timing Characteristics (see Figure 22-12). Updated the Internal RC Accuracy parameter numbers (see Table 22-18 and Table 22-19).
Section 23.0 “High Temperature Electrical Characteristics”	Added new chapter with high temperature specifications.
“Product Identification System”	Added the “H” definition for high temperature.

Revision F (November 2009)

This revision includes minor typographical and formatting changes throughout the data sheet text.

All other major changes are referenced by their respective section in the following table.

TABLE A-5: MAJOR SECTION UPDATES

Section Name	Update Description
“High-Performance, 16-bit Digital Signal Controllers”	Updated MIPS rating from 16 to 20 for high temperature devices in **“Operating Range:”** and in TABLE 22-1: “Operating MIPS vs. Voltage”.

Revision G (January 2011)

This revision includes typographical and formatting changes throughout the data sheet text. In addition, all instances of VDDCORE have been removed.

All other major changes are referenced by their respective section in the following table.

TABLE A-6: MAJOR SECTION UPDATES

Section Name	Update Description
High-Performance, 16-bit Digital Signal Controllers	Added the SSOP package information (see **“Packaging:”**, Table 1, and **“Pin Diagrams”**).
Section 2.0 “Guidelines for Getting Started with 16-bit Digital Signal Controllers”	Updated the title of **Section 2.3 “CPU Logic Filter Capacitor Connection (VCAP)”**. The frequency limitation for device PLL start-up conditions was updated in **Section 2.7 “Oscillator Value Conditions on Device Start-up”**. The second paragraph in **Section 2.9 “Unused I/Os”** was updated.
Section 3.0 “CPU”	Removed references to DMA in the CPU Core Block Diagram (see Figure 3-1).
Section 4.0 “Memory Organization”	Updated the data memory reference in the third paragraph in **Section 4.2 “Data Address Space”**. The All Resets values for the following SFRs in the Timer Register Map were changed (see Table 4-5): • TMR1 • TMR2 • TMR3
Section 8.0 “Oscillator Configuration”	Added Note 3 to the OSCCON: Oscillator Control Register (see Register 8-1). Added Note 2 to the CLKDIV: Clock Divisor Register (see Register 8-2). Added Note 1 to the PLLFBD: PLL Feedback Divisor Register (see Register 8-3). Added Note 2 to the OSCTUN: FRC Oscillator Tuning Register (see Register 8-4).
Section 18.0 “10-bit/12-bit Analog-to-Digital Converter (ADC)”	Updated the VREFL references in the ADC1 module block diagrams (see Figure 18-1 and Figure 18-2).
Section 19.0 “Special Features”	Added a new paragraph and removed the third paragraph in **Section 19.1 “Configuration Bits”**. Added the column “RTSP Effects” to the Configuration Bits Descriptions (see Table 19-2).
Section 24.0 “Packaging Information”	Added the 28-Lead SSOP package information (see **Section 24.1 “Package Marking Information”** and **Section 24.2 “Package Details”**).

TABLE A-6: MAJOR SECTION UPDATES (CONTINUED)

Section Name	Update Description
Section 22.0 “Electrical Characteristics”	Added the 28-pin SSOP Thermal Packaging Characteristics (see Table 22-3). Removed Note 4 from the DC Temperature and Voltage Specifications (see Table 22-4). Updated the maximum value for parameters DI18 and DI19 and added parameters DI28, DI29, DI60a, DI60b, and DI60c to the I/O Pin Input Specifications (see Table 22-9). Updated Note 3 in the PLL Clock Timing Specifications (see Table 22-17). Removed Note 2 from the AC Characteristics: Internal RC Accuracy (see Table 22-18). Updated the characteristic description for parameter DI35 in the I/O Timing Requirements (see Table 22-20). Updated *all* SPI specifications (see Table 22-28 through Table 22-35 and Figure 22-10 through Figure 22-16). Added Note 4 to the 12-bit mode ADC Module Specifications (see Table 22-39). Added Note 4 to the 10-bit mode ADC Module Specifications (see Table 22-40).
Section 23.0 “High Temperature Electrical Characteristics”	Updated all ambient temperature end range values to +150ºC throughout the chapter. Updated the storage temperature end range to +160ºC. Updated the maximum junction temperature from +145ºC to +155ºC. Updated Note 1 in the PLL Clock Timing Specifications (see Table 23-10). Added Note 3 to the 12-bit Mode ADC Module Specifications (see Table 23-17). Added Note 3 to the 10-bit Mode ADC Module Specifications (see Table 23-18).
“Product Identification System”	Added the “SS” definition for the SSOP package.

Revision H (July 2011)

This revision includes typographical and formatting changes throughout the data sheet text.

All other major changes are referenced by their respective section in the following table.

TABLE A-7: MAJOR SECTION UPDATES

Section Name	Update Description
Section 19.0 “Special Features”	Added Note 3 to the Connections for the On-chip Voltage Regulator diagram (see Figure 19-1).
Section 22.0 “Electrical Characteristics”	Removed Note 3 and parameter DC10 (V_{CORE}) from the DC Temperature and Voltage Specifications (see Table 22-4). Updated the Characteristics definition and Conditions for parameter BO10 in the Electrical Characteristics: BOR (see Table 22-11). Added Note 1 to the Internal Voltage Regulator Specifications (see Table 22-13).

NOTES:

INDEX

A

- A/D Converter 171
  - Initialization 171
  - Key Features 171
- AC Characteristics 215, 253
  - ADC Module 256
  - ADC Module (10-bit Mode) 257
  - ADC Module (12-bit Mode) 256
  - Internal RC Accuracy 217
  - Load Conditions 215, 253
- ADC Module
  - ADC11 Register Map 43, 45, 46
- Alternate 73
- Alternate Interrupt Vector Table 73
- Alternate Interrupt Vector Table (AIVT) 73
- Arithmetic Logic Unit (ALU) 27
- Assembler
  - MPASM Assembler 202

B

- Barrel Shifter 31
- Bit-Reversed Addressing 52
  - Example 53
  - Implementation 52
  - Sequence Table (16-Entry) 53
- Block Diagrams
  - 16-bit Timer1 Module 137
  - A/D Module 172, 173
  - Connections for On-Chip Voltage Regulator 188
  - Device Clock 101, 103
  - DSP Engine 28
  - dsPIC33F 12
  - dsPIC33F CPU Core 22
  - Input Capture 145
  - Output Compare 147
  - PLL 103
  - Reset System 65
  - Shared Port Structure 115
  - SPI 151
  - Timer2 (16-bit) 141
  - Timer2/3 (32-bit) 140
  - UART 165
  - Watchdog Timer (WDT) 189

C

- C Compilers
  - MPLAB C18 202
- Clock Switching 110
  - Enabling 110
  - Sequence 110
- Code Examples
  - Erasing a Program Memory Page 63
  - Initiating a Programming Sequence 64
  - Loading Write Buffers 64
  - Port Write/Read 116
  - PWRSAV Instruction Syntax 111
- Code Protection 185, 190
- Configuration Bits 185
  - Description (Table) 186
- Configuration Register Map 185
- Configuring Analog Port Pins 116
- CPU
  - Control Register 24
- CPU Clocking System 102
  - Options 102
  - Selection 102
- Customer Change Notification Service 287
- Customer Notification Service 287
- Customer Support 287

D

- Data Accumulators and Adder/Subtractor 29
  - Data Space Write Saturation 31
  - Overflow and Saturation 29
  - Round Logic 30
  - Write Back 30
- Data Address Space 35
  - Alignment 35
  - Memory Map for dsPIC33F Devices with 8 KBs RAM 36
  - Near Data Space 35
  - Software Stack 49
  - Width 35
- DC Characteristics 206
  - Doze Current (I_{DOZE}) 251
  - High Temperature 250
  - I/O Pin Input Specifications 211
  - I/O Pin Output 251
  - I/O Pin Output Specifications 213
  - Idle Current (I_{DOZE}) 210
  - Idle Current (I_{IDLE}) 209
  - Operating Current (I_{DD}) 208
  - Operating MIPS vs. Voltage 250
  - Power-Down Current (I_{PD}) 210
  - Power-down Current (I_{PD}) 250
  - Program Memory 214, 252
  - Temperature and Voltage 250
  - Temperature and Voltage Specifications 207
  - Thermal Operating Conditions 250
- Development Support 201
- DSP Engine 27
  - Multiplier 29

E

- Electrical Characteristics 205
  - AC 215, 253
- Equations
  - Device Operating Frequency 102
- Errata 9

F

- Flash Program Memory 59
  - Control Registers 60
  - Operations 60
  - Programming Algorithm 63
  - RTSP Operation 60
  - Table Instructions 59
- Flexible Configuration 185

H

- High Temperature Electrical Characteristics 249

I

- I/O Ports 115
  - Parallel I/O (PIO) 115
  - Write/Read Timing 116
- I^2C
  - Operating Modes 157
  - Registers 157

- I^2C Module
  - I2C1 Register Map 42
- In-Circuit Debugger 191
- In-Circuit Emulation 185
- In-Circuit Serial Programming (ICSP) 185, 191
- Input Capture
  - Registers 146
- Input Change Notification 116
- Instruction Addressing Modes 49
  - File Register Instructions 49
  - Fundamental Modes Supported 50
  - MAC Instructions 50
  - MCU Instructions 49
  - Move and Accumulator Instructions 50
  - Other Instructions 50
- Instruction Set
  - Overview 196
  - Summary 193
- Instruction-Based Power-Saving Modes 111
  - Idle 112
  - Sleep 111
- Internal RC Oscillator
  - Use with WDT 189
- Internet Address 287
- Interrupt Control and Status Registers 77
  - IECx 77
  - IFSx 77
  - INTCON1 77
  - INTCON2 77
  - IPCx 77
- Interrupt Setup Procedures 99
  - Initialization 99
  - Interrupt Disable 99
  - Interrupt Service Routine 99
  - Trap Service Routine 99
- Interrupt Vector Table (IVT) 73
- Interrupts Coincident with Power Save Instructions 112

J

- JTAG Boundary Scan Interface 185

M

- Memory Organization 33
- Microchip Internet Web Site 287
- Modulo Addressing 51
  - Applicability 52
  - Operation Example 51
  - Start and End Address 51
  - W Address Register Selection 51
- MPLAB ASM30 Assembler, Linker, Librarian 202
- MPLAB Integrated Development Environment Software 201
- MPLAB PM3 Device Programmer 204
- MPLAB REAL ICE In-Circuit Emulator System 203
- MPLINK Object Linker/MPLIB Object Librarian 202

N

- NVM Module
  - Register Map 48

O

- Open-Drain Configuration 116
- Output Compare 147
  - Registers 149

P

- Packaging 259
  - Details 261
  - Marking 259, 260
- Peripheral Module Disable (PMD) 112
- Pinout I/O Descriptions (table) 13
- PMD Module
  - Register Map 48
- PORTA
  - Register Map 47
- PORTB
  - Register Map 47
- Power-on Reset (POR) 70
- Power-Saving Features 111
  - Clock Frequency and Switching 111
- Program Address Space 33
  - Construction 54
  - Data Access from Program Memory Using
    - Program Space Visibility 57
  - Data Access from Program Memory Using
    - Table Instructions 56
  - Data Access from, Address Generation 55
  - Memory Map 33
  - Table Read Instructions
    - TBLRDH 56
    - TBLRDL 56
  - Visibility Operation 57
- Program Memory
  - Interrupt Vector 34
  - Organization 34
  - Reset Vector 34

R

- Reader Response 288
- Registers
  - AD1CHS0 (ADC1 Input Channel 0 Select 181
  - AD1CHS123 (ADC1 Input Channel 1, 2, 3 Select) 179
  - AD1CON1 (ADC1 Control 1) 175
  - AD1CON2 (ADC1 Control 2) 177
  - AD1CON3 (ADC1 Control 3) 178
  - AD1CSSL (ADC1 Input Scan Select Low) 183
  - AD1PCFGL (ADC1 Port Configuration Low) 183
  - CLKDIV (Clock Divisor) 106
  - CORCON (Core Control) 26, 78
  - I2CxCON (I2Cx Control) 159
  - I2CxMSK (I2Cx Slave Mode Address Mask) 163
  - I2CxSTAT (I2Cx Status) 161
  - ICxCON (Input Capture x Control) 146
  - IEC0 (Interrupt Enable Control 0) 86, 88, 89
  - IFS0 (Interrupt Flag Status 0) 82
  - IFS1 (Interrupt Flag Status 1) 84
  - IFS4 (Interrupt Flag Status 4) 85
  - INTCON1 (Interrupt Control 1) 79
  - INTCON2 (Interrupt Control 2) 81
  - INTTREG Interrupt Control and Status Register 98
  - IPC0 (Interrupt Priority Control 0) 90
  - IPC1 (Interrupt Priority Control 1) 91
  - IPC16 (Interrupt Priority Control 16) 97
  - IPC2 (Interrupt Priority Control 2) 92
  - IPC3 (Interrupt Priority Control 3) 93
  - IPC4 (Interrupt Priority Control 4) 94
  - IPC5 (Interrupt Priority Control 5) 95
  - IPC7 (Interrupt Priority Control 7) 96
  - NVMCOM (Flash Memory Control) 61, 62
  - OCxCON (Output Compare x Control) 149
  - OSCCON (Oscillator Control) 104
  - OSCTUN (FRC Oscillator Tuning) 109
  - PLLFBD (PLL Feedback Divisor) 108

  - PMD1 (Peripheral Module Disable Control Register 1) 113
  - PMD2 (Peripheral Module Disable Control Register 2) 114
  - RCON (Reset Control) 66
  - SPIxCON1 (SPIx Control 1) 153
  - SPIxCON2 (SPIx Control 2) 155
  - SPIxSTAT (SPIx Status and Control) 152
  - SR (CPU Status) 24, 78
  - T1CON (Timer1 Control) 138
  - TxCON (T2CON, T4CON, T6CON or T8CON Control) 142
  - TyCON (T3CON, T5CON, T7CON or T9CON Control) 143
  - UxMODE (UARTx Mode) 166
  - UxSTA (UARTx Status and Control) 168

- Reset
  - Illegal Opcode 65, 72
  - Trap Conflict 71, 72
  - Uninitialized W Register 65, 72
- Reset Sequence 73
- Resets 65

S

- Serial Peripheral Interface (SPI) 151
- Software Reset Instruction (SWR) 71
- Software Simulator (MPLAB SIM) 203
- Software Stack Pointer, Frame Pointer
  - CALL Stack Frame 49
- Special Features of the CPU 185
- SPI Module
  - SPI1 Register Map 42
- Symbols Used in Opcode Descriptions 194
- System Control
  - Register Map 48

T

- Temperature and Voltage Specifications
  - AC 215, 253
- Timer1 137
- Timer2/3, Timer4/5, Timer6/7 and Timer8/9 139
- Timing Characteristics
  - CLKO and I/O 219
- Timing Diagrams
  - 10-bit A/D Conversion 246
  - 10-bit A/D Conversion (CHPS = 01, SIMSAM = 0, ASAM = 0, SSRC = 000) 246
  - 12-bit A/D Conversion (ASAM = 0, SSRC = 000) 245
  - Brown-out Situations 71
  - External Clock 216
  - I2Cx Bus Data (Master Mode) 238
  - I2Cx Bus Data (Slave Mode) 240
  - I2Cx Bus Start/Stop Bits (Master Mode) 238
  - I2Cx Bus Start/Stop Bits (Slave Mode) 240
  - Input Capture (CAPx) 224
  - OC/PWM 225
  - Output Compare (OCx) 224
  - Reset, Watchdog Timer, Oscillator Start-up Timer and Power-up Timer 220
  - Timer1, 2, 3, 4, 5, 6, 7, 8, 9 External Clock 222
- Timing Requirements
  - ADC Conversion (10-bit mode) 258
  - ADC Conversion (12-bit Mode) 258
  - CLKO and I/O 219
  - External Clock 216
  - Input Capture 224
  - SPIx Master Mode (CKE = 0) 254
  - SPIx Module Master Mode (CKE = 1) 254
  - SPIx Module Slave Mode (CKE = 0) 255
  - SPIx Module Slave Mode (CKE = 1) 255
- Timing Specifications
  - 10-bit A/D Conversion Requirements 247
  - 12-bit A/D Conversion Requirements 245
  - I2Cx Bus Data Requirements (Master Mode) 239
  - I2Cx Bus Data Requirements (Slave Mode) 241
  - Output Compare Requirements 224
  - PLL Clock 217, 253
  - Reset, Watchdog Timer, Oscillator Start-up Timer, Power-up Timer and Brown-out Reset Requirements 221
  - Simple OC/PWM Mode Requirements 225
  - Timer1 External Clock Requirements 222
  - Timer2 External Clock Requirements 223
  - Timer3 External Clock Requirements 223

U

- UART Module
  - UART1 Register Map 42
- Using the RCON Status Bits 72

V

- Voltage Regulator (On-Chip) 188

W

- Watchdog Time-out Reset (WDTR) 71
- Watchdog Timer (WDT) 185, 189
  - Programming Considerations 189
- WWW Address 287
- WWW, On-Line Support 9

NOTES:

THE MICROCHIP WEB SITE

Microchip provides online support via our WWW site at www.microchip.com. This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQs), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at www.microchip.com. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support
- Development Systems Information Line

Customers should contact their distributor, representative or field application engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the web site at: http://microchip.com/support

READER RESPONSE

It is our intention to provide you with the best documentation possible to ensure successful use of your Microchip product. If you wish to provide your comments on organization, clarity, subject matter, and ways in which our documentation can better serve you, please FAX your comments to the Technical Publications Manager at (480) 792-4150.

Please list the following information, and use this outline to provide us with your comments about this document.

TO: Technical Publications Manager
RE: Reader Response

Total Pages Sent ________

From: Name ____
Company ____
Address ____
City / State / ZIP / Country ________________________________
Telephone: (_______) _________ - _________
FAX: (______) _________ - _________

Application (optional):

Would you like a reply? ___Y ___N

Device: dsPIC33FJ32GP202/204 and dsPIC33FJ16GP304

Literature Number: DS70290H

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)?

____

____

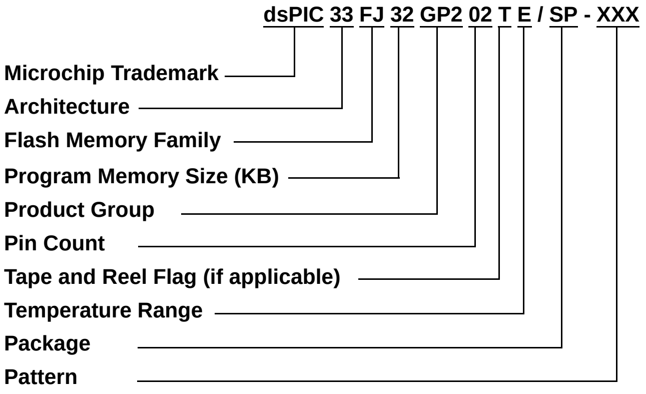
7. How would you improve this document?

____

____

PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.



Architecture:	33	=	16-bit Digital Signal Controller
Flash Memory Family:	FJ	=	Flash program memory, 3.3V
Product Group:	GP2	=	General purpose family
	GP3	=	General purpose family
Pin Count:	02	=	28-pin
	03	=	44-pin
Temperature Range:	I	=	-40° C to +85° C (Industrial)
	E	=	-40° C to +125° C (Extended)
	H	=	-40° C to +150° C (High)
Package:	SP	=	Skinny Plastic Dual In-Line - 300 mil body (SPDIP)
	SO	=	Plastic Small Outline - Wide - 7.5 mm body (SOIC)
	SS	=	Plastic Shrink Small Outline - 5.3 mm body (SSOP)
	ML	=	Plastic Quad, No Lead Package - 8x8 mm body (QFN)
	PT	=	Plastic Thin Quad Flatpack - 10x10x1 mm body (TQFP)
	MM	=	Plastic Quad, No Lead Package - 6x6 mm body (QFN-S)

Examples:

a) dsPIC33FJ32GP202-E/SP:
General-purpose dsPIC33, 32 KB program memory, 28-pin, Extended temp., SPDIP package.

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://www.microchip.com/support
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Cleveland
Independence, OH
Tel: 216-447-0464
Fax: 216-447-0643

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Farmington Hills, MI
Tel: 248-538-2250
Fax: 248-538-2260

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608

Santa Clara
Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

Toronto
Mississauga, Ontario,
Canada
Tel: 905-673-0699
Fax: 905-673-6509

ASIA/PACIFIC

Asia Pacific Office
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

China - Beijing
Tel: 86-10-8569-7000
Fax: 86-10-8528-2104

China - Chengdu
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

China - Chongqing
Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

China - Hangzhou
Tel: 86-571-2819-3180
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 - 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: 886-3-6578-300
Fax: 886-3-6578-370

Taiwan - Kaohsiung
Tel: 886-7-213-7830
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

05/02/11