



**MICROCHIP**

---

**PIC24FJ16MC101/102**  
**Data Sheet**

High-Performance, Ultra Low Cost  
16-bit Microcontrollers

---

**Note the following details of the code protection feature on Microchip devices:**

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

---

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE. Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights.

**Trademarks**

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

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

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

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

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

© 2011, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

 Printed on recycled paper.

ISBN: 978-1-61341-314-2

*Microchip received ISO/TS-16949:2002 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.*

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

---

---

## High-Performance, Ultra Low Cost 16-bit Microcontrollers

---

---

### Operating Range:

- Up to 16 MIPS operation (3.0V-3.6V):
  - Industrial temperature range (-40°C to +85°C)
  - Extended temperature range (-40°C to +125°C)

### On-Chip Flash and SRAM:

- Flash program memory (16 Kbytes)
- Data SRAM (1 Kbyte)
- Security for program Flash

### System Management:

- Flexible clock options:
  - External, crystal, resonator, internal FRC
  - Phase-Locked Loop (PLL)
- High-accuracy internal FRC
  - $\pm 0.25\%$  typical
- Power-on Reset (POR)
- Power-up Timer (PWRT)
- Oscillator Start-up Timer (OST)
- Brown-out Reset (BOR)
- Watchdog Timer with its own RC oscillator
- Fail-Safe Clock Monitor (FSCM)

### Motor Control PWM:

- 6-channel 16-bit Motor Control PWM:
  - Three duty cycle generators
  - Independent or Complementary mode
  - Programmable dead time and output polarity
  - Edge-aligned or center-aligned
  - Manual output override control
  - Up to two Fault inputs
  - Trigger for ADC conversions
  - PWM frequency for 16-bit resolution (@ 16 MIPS) = 488 Hz for Edge-Aligned mode, 244 Hz for Center-Aligned mode
  - PWM frequency for 11-bit resolution (@ 16 MIPS) = 15.63 kHz for Edge-Aligned mode, 7.81 kHz for Center-Aligned mode

### Power Management:

- Single supply on-chip voltage regulator
- Switch between clock sources in real time
- Idle, Sleep, and Doze modes with fast wake-up

### Analog Peripherals:

- 10-bit, 1.1 Msps Analog-to-Digital Converter (ADC):
  - Two and four simultaneous samples
  - Up to six input channels with auto-scanning
  - Conversion start can be manual or synchronized with one of four trigger sources
  - Sleep mode conversion for low-power applications
  - $\pm 2$  LSb max integral nonlinearity
  - $\pm 1$  LSb max differential nonlinearity
- Three Analog Comparators with programmable input/output configuration:
  - Up to four inputs per Comparator
  - Blanking function
  - Output digital filter
- Charge Time Measurement Unit (CTMU):
  - Supports capacitive touch sensing for touch screens and capacitive switches (mTouch™)
  - Provides high-resolution time measurement for advanced sensor applications
  - 200 ps resolution for time measurement and accurate temperature sensing
  - On-chip high-resolution temperature measurement capability

# PIC24FJ16MC101/102

---

## Timers/Capture/Compare/PWM:

- 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 three channels):
  - Capture on up, down, or both edges
  - 16-bit capture input functions
  - 4-deep FIFO on each capture
- Output Compare (up to two channels):
  - Single or Dual 16-bit Compare mode
  - 16-bit Glitchless PWM mode
- Hardware Real-Time Clock and Calendar (RTCC):
  - Provides clock, calendar and alarm function

## Digital I/O:

- Peripheral Pin Select functionality
- Up to 21 programmable digital I/O pins
- Wake-up/Interrupt-on-Change for up to 21 pins
- Output pins can drive from 3.0V to 3.6V
- Up to 5.5V output with open drain configuration on 5V tolerant pins
- All digital input pins are 5V tolerant
- Up to 8 mA sink on designated pins

## 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<sup>2</sup>C™:
  - 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 2.0 bus support
  - IrDA® encoding and decoding in hardware
  - High-Speed mode
  - Hardware Flow Control with CTS and RTS

## Interrupt Controller:

- 5-cycle latency
- Up to 23 available interrupt sources
- Up to three external interrupts
- Seven programmable priority levels
- Four processor exceptions

## High-Performance MCU CPU Features:

- 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
- 73 base instructions: mostly one word/one cycle
- Flexible and powerful indirect addressing mode
- Software stack
- 16 x 16 integer multiply operations
- 32/16 and 16/16 integer divide operations
- Up to ±16-bit shifts

## Packaging:

- 20-pin PDIP/SOIC/SSOP
- 28-pin SPDIP/SOIC/SSOP/QFN
- 28-pin QFN: 6x6 mm
- 36-pin TLA: 5x5 mm

**Note:** See [Table 1](#) for the list of peripheral features per device.

## PIC24FJ16MC101/102 PRODUCT FAMILIES

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

**TABLE 1: PIC24FJ16MC101/102 CONTROLLER FAMILIES**

Device	Pins	Program Flash (Kbyte)	RAM (Kbytes)	Remappable Peripherals							Motor Control PWM	PWM Faults	10-Bit, 1.1 Msps ADC	RTCC	I <sup>2</sup> C™	Comparators	CTMU	I/O Pins	Packages
				Remappable Pins	16-bit Timer <sup>(1)</sup>	Input Capture	Output Compare	UART	External Interrupts <sup>(2)</sup>	SPI									
PIC24FJ16MC101	20	16	1	10	3	3	2	1	3	1	6-ch	1	1 ADC, 4-ch	Y	1	3	Y	15	PDIP, SOIC, SSOP
PIC24FJ16MC102	28	16	1	16	3	3	2	1	3	1	6-ch	2	1 ADC, 6-ch	Y	1	3	Y	21	SPDIP, SOIC, SSOP, QFN
	36	16	1	16	3	3	2	1	3	1	6-ch	2	1 ADC, 6-ch	Y	1	3	Y	21	TLA

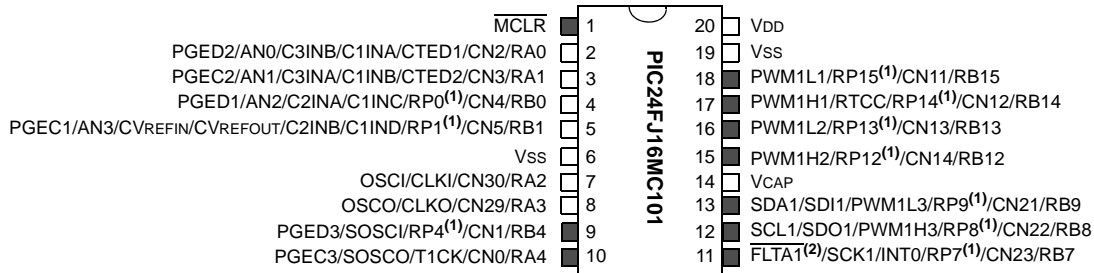
- Note 1:** Two out of three timers are remappable.  
**Note 2:** Two out of three interrupts are remappable.

# PIC24FJ16MC101/102

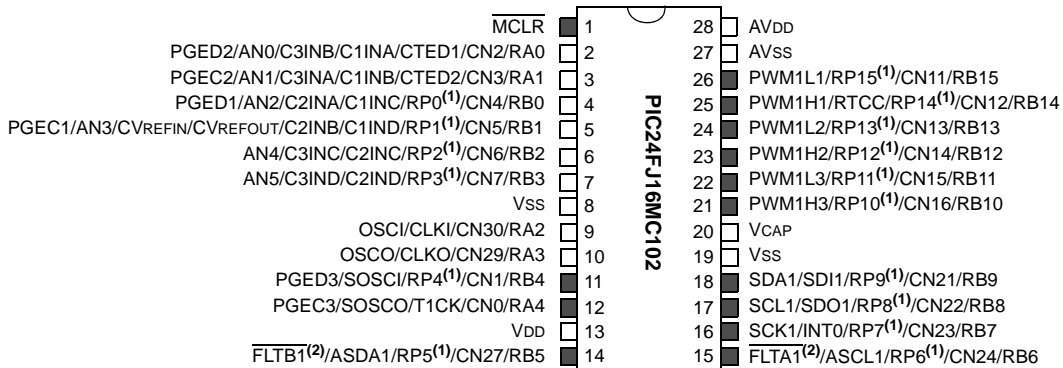
## Pin Diagrams

### 20-Pin PDIP/SOIC/SSOP

■ = Pins are up to 5V tolerant



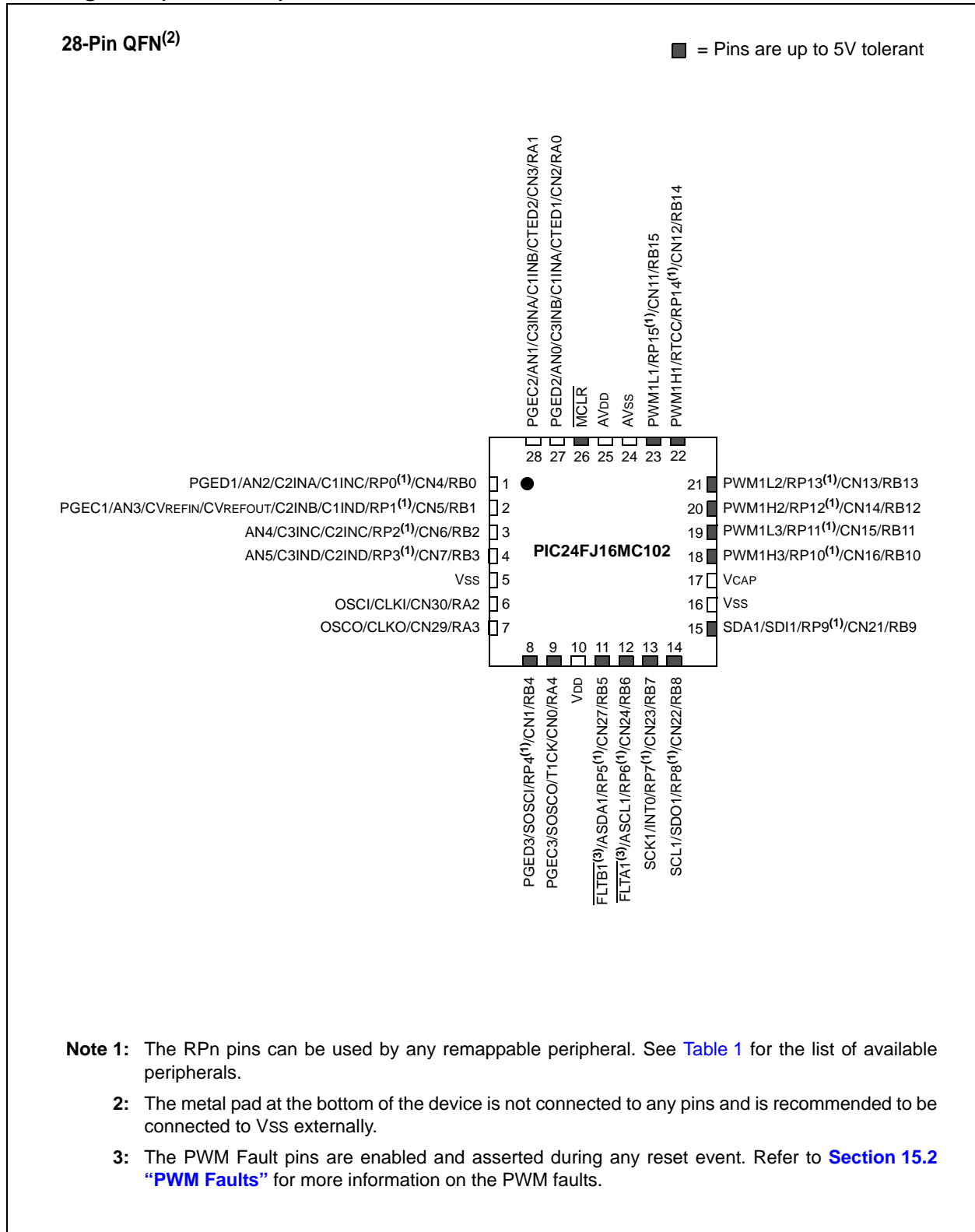
### 28-Pin SPDIP/SOIC/SSOP



**Note 1:** The RPN pins can be used by any remappable peripheral. See [Table 1](#) for the list of available peripherals.

**2:** The PWM Fault pins are enabled and asserted during any reset event. Refer to [Section 15.2 "PWM Faults"](#) for more information on the PWM faults.

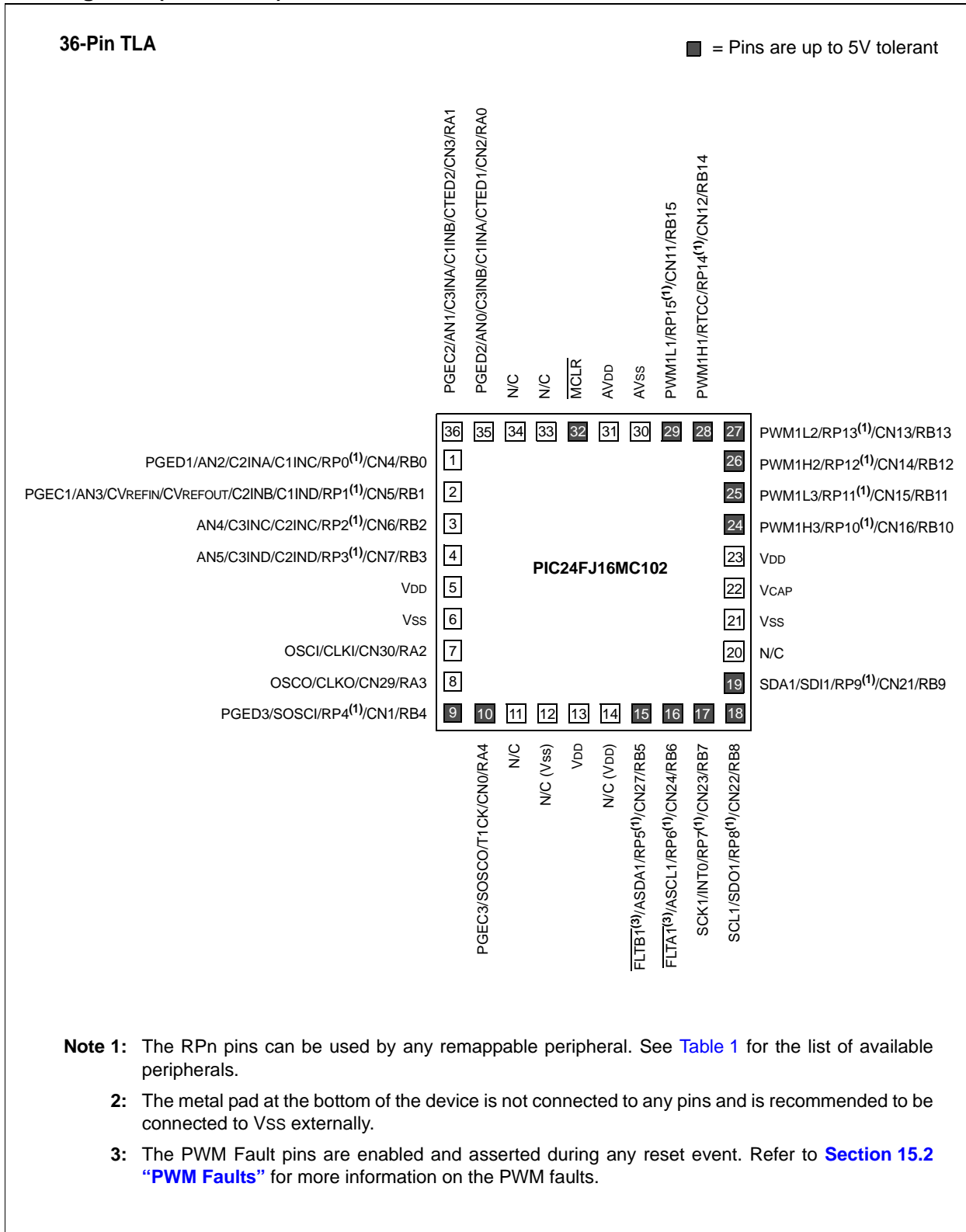
## Pin Diagrams (Continued)



- Note 1:** The RPN pins can be used by any remappable peripheral. See [Table 1](#) for the list of available peripherals.
- 2:** The metal pad at the bottom of the device is not connected to any pins and is recommended to be connected to VSS externally.
- 3:** The PWM Fault pins are enabled and asserted during any reset event. Refer to [Section 15.2 "PWM Faults"](#) for more information on the PWM faults.

# PIC24FJ16MC101/102

## Pin Diagrams (Continued)





## Table of Contents

PIC24FJ16MC101/102 Product Families.....	5
1.0 Device Overview .....	11
2.0 Guidelines for Getting Started with 16-bit Microcontrollers .....	17
3.0 CPU.....	21
4.0 Memory Organization.....	27
5.0 Flash Program Memory.....	51
6.0 Resets .....	55
7.0 Interrupt Controller .....	63
8.0 Oscillator Configuration.....	93
9.0 Power-Saving Features.....	101
10.0 I/O Ports .....	107
11.0 Timer1 .....	123
12.0 Timer2/3 Feature .....	125
13.0 Input Capture.....	131
14.0 Output Compare.....	133
15.0 Motor Control PWM Module.....	137
16.0 Serial Peripheral Interface (SPI).....	153
17.0 Inter-Integrated Circuit™ (I <sup>2</sup> C™).....	159
18.0 Universal Asynchronous Receiver Transmitter (UART) .....	167
19.0 10-bit Analog-to-Digital Converter (ADC) .....	173
20.0 Comparator Module.....	185
21.0 Real-Time Clock and Calendar (RTCC) .....	199
22.0 Charge Time Measurement Unit (CTMU) .....	209
23.0 Special Features .....	215
24.0 Instruction Set Summary .....	223
25.0 Development Support.....	231
26.0 Electrical Characteristics .....	235
27.0 Packaging Information.....	277
Appendix A: Revision History.....	295
Index .....	297
The Microchip Web Site.....	301
Customer Change Notification Service .....	301
Customer Support.....	301
Reader Response .....	302
Product Identification System .....	303

## TO OUR VALUED CUSTOMERS

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

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

### Most Current Data Sheet

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

<http://www.microchip.com>

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

### Errata

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

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

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

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

### Customer Notification System

Register on our web site at [www.microchip.com](http://www.microchip.com) to receive the most current information on all of our products.

## 1.0 DEVICE OVERVIEW

**Note 1:** This data sheet summarizes the features of the PIC24FJ16MC101/102 devices. However, it is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to the latest family reference sections of the “PIC24F Family Reference Manual”, which are available from the Microchip web site ([www.microchip.com](http://www.microchip.com)).

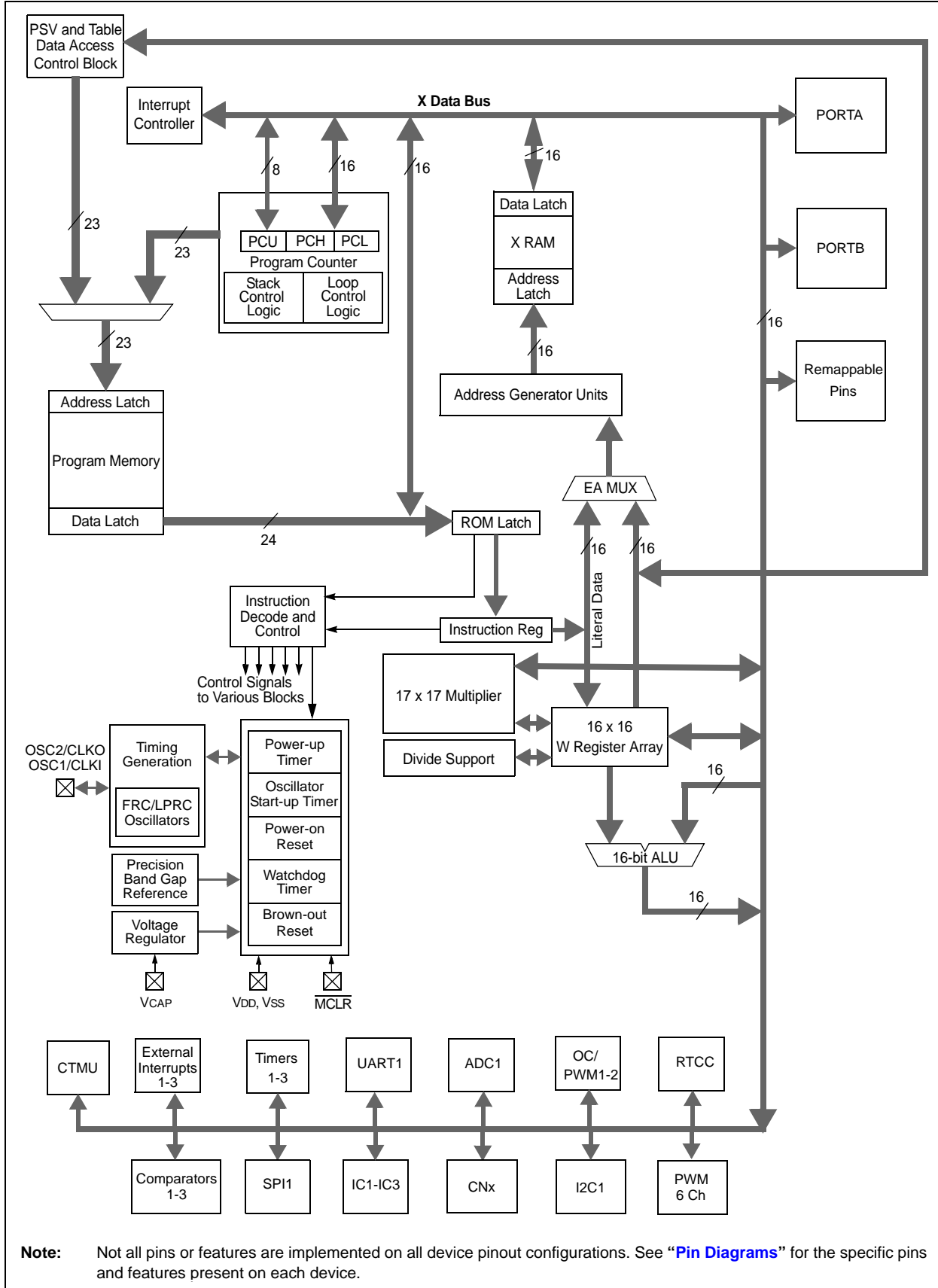
**2:** It is important to note that the specifications in **Section 26.0 “Electrical Characteristics”** of this data sheet, supercede any specifications that may be provided in PIC24F Family Reference Manual sections.

This document contains device specific information for the PIC24FJ16MC101/102 Microcontroller (MCU) devices. Central to all PIC24F devices is the 16-bit modified Harvard architecture, first introduced with Microchip’s dsPIC<sup>®</sup> digital signal controllers.

[Figure 1-1](#) shows a general block diagram of the core and peripheral modules in the PIC24FJ16MC101/102 family of devices. [Table 1-1](#) lists the functions of the various pins shown in the pinout diagrams.

# PIC24FJ16MC101/102

**FIGURE 1-1: PIC24FJ16MC101/102 BLOCK DIAGRAM**



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

Pin Name	Pin Type	Buffer Type	PPS	Description
AN0-AN5	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-CN7 CN11-CN16 CN21-CN24 CN27 CN29-CN30	I	ST ST ST ST ST	No No No No No	Change notification inputs. Can be software programmed for internal weak pull-ups on all inputs.
IC1-IC3	I	ST	Yes	Capture inputs 1/2/3.
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	I/O	ST	No	PORTA is a bidirectional I/O port.
RB0-RB15	I/O	ST	No	PORTB 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.
<u>U1CTS</u> <u>U1RTS</u> 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 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.

**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

- Note 1:** An external pull-down resistor is required for the FLTA1 pin on PIC24FJ16MC101 (20-pin) devices.  
**2:** The FLTB1 pin is not available on PIC24FJ16MC101 (20-pin) devices.  
**3:** The PWM Fault pins are enabled during any reset event. Refer to [Section 15.2 “PWM Faults”](#) for more information on the PWM faults.

# PIC24FJ16MC101/102

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

Pin Name	Pin Type	Buffer Type	PPS	Description
FLTA1 <sup>(1,3)</sup>	I	ST	No	PWM1 Fault A input.
FLTB1 <sup>(2,3)</sup>	I	ST	No	PWM1 Fault B input.
PWM1L1	O	—	No	PWM1 Low output 1.
PWM1H1	O	—	No	PWM1 High output 1.
PWM1L2	O	—	No	PWM1 Low output 2.
PWM1H2	O	—	No	PWM1 High output 2.
PWM1L3	O	—	No	PWM1 Low output 3.
PWM1H3	O	—	No	PWM1 High output 3.
RTCC	O	Digital	No	RTCC Alarm output.
CTPLS	O	Digital	Yes	CTMU Pulse Output.
CTED1	I	Digital	No	CTMU External Edge Input 1.
CTED2	I	Digital	No	CTMU External Edge Input 2.
CVREF	I	Analog	No	Comparator Voltage Positive Reference Input.
C1INA	I	Analog	No	Comparator 1 Positive Input A.
C1INB	I	Analog	No	Comparator 1 Negative Input B.
C1INC	I	Analog	No	Comparator 1 Negative Input C.
C1IND	I	Analog	No	Comparator 1 Negative Input D.
C1OUT	O	Digital	Yes	Comparator 1 Output.
C2INA	I	Analog	No	Comparator 2 Positive Input A.
C2INB	I	Analog	No	Comparator 2 Negative Input B.
C2INC	I	Analog	No	Comparator 2 Negative Input C.
C2IND	I	Analog	No	Comparator 2 Negative Input D.
C2OUT	O	Digital	Yes	Comparator 2 Output.
C3INA	I	Analog	No	Comparator 3 Positive Input A.
C3INB	I	Analog	No	Comparator 3 Negative Input B.
C3INC	I	Analog	No	Comparator 3 Negative Input C.
C3IND	I	Analog	No	Comparator 3 Negative Input D.
C3OUT	O	Digital	Yes	Comparator 3 Output.
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.
MCLR	I/P	ST	No	Master Clear (Reset) input. This pin is an active-low Reset to the device.
AVDD	P	P	No	Positive supply for analog modules. This pin must be connected at all times. For devices without this pin, this signal is connected to VDD internally.
AVSS	P	P	No	Ground reference for analog modules. For devices without this pin, this signal is connected to VSS internally.
VDD	P	—	No	Positive supply for peripheral logic and I/O pins.
VCAP	P	—	No	CPU logic filter capacitor connection.
VSS	P	—	No	Ground reference for 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

**Note 1:** An external pull-down resistor is required for the FLTA1 pin on PIC24FJ16MC101 (20-pin) devices.  
**Note 2:** The FLTB1 pin is not available on PIC24FJ16MC101 (20-pin) devices.  
**Note 3:** The PWM Fault pins are enabled during any reset event. Refer to [Section 15.2 “PWM Faults”](#) for more information on the PWM faults.

## 1.1 Referenced Sources

This device data sheet is based on the following individual chapters of the “PIC24F Family Reference Manual”. These documents should be considered as the primary reference for the operation of a particular module or device feature.

**Note:** To access the documents listed below, browse to the specific device product page of the Microchip web site ([www.microchip.com](http://www.microchip.com)).

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

- **Section 1. “Introduction”** (DS39718)
- **Section 2. “CPU”** (DS39703)
- **Section 3. “Data Memory”** (DS39717)
- **Section 4. “Program Memory”** (DS39715)
- **Section 6. “Oscillator”** (DS39700)
- **Section 7. “Reset”** (DS39712)
- **Section 8. “Interrupts”** (DS39707)
- **Section 9. “Watchdog Timer (WDT)”** (DS39697)
- **Section 10. “Power-Saving Features”** (DS39698)
- **Section 11. “Charge Time Measurement Unit (CTMU)”** (DS39724)
- **Section 12. “I/O Ports with Peripheral Pin Select (PPS)”** (DS39711)
- **Section 14. “Timers”** (DS39704)
- **Section 15. “Input Capture”** (DS39701)
- **Section 16. “Output Compare”** (DS39706)
- **Section 21. “UART”** (DS39708)
- **Section 23. “Serial Peripheral Interface (SPI)”** (DS39699)
- **Section 24. “Inter-Integrated Circuit™ (I<sup>2</sup>C™)”** (DS39702)
- **Section 29. “Real-Time Clock and Calendar (RTCC)”** (DS39696)
- **Section 32. “High-Level Device Integration”** (DS39719)
- **Section 33. “Programming and Diagnostics”** (DS39716)
- **Section 46. “10-bit Analog-to-Digital Converter (ADC) with 4 Simultaneous Conversions”** (DS39737)
- **Section 47. “Motor Control PWM”** (DS39735)
- **Section 48. “Comparator with Blanking”** (DS39741)

Notes:



## 2.0 GUIDELINES FOR GETTING STARTED WITH 16-BIT MICROCONTROLLERS

**Note 1:** This data sheet summarizes the features of the PIC24FJ16MC101/102 family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to the “PIC24F Family Reference Manual”. Please see the Microchip web site ([www.microchip.com](http://www.microchip.com)) for the latest PIC24F Family Reference Manual sections.

**2:** It is important to note that the specifications in [Section 26.0 “Electrical Characteristics”](#) of this data sheet, supercede any specifications that may be provided in PIC24F Family Reference Manual sections.

**3:** 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 PIC24FJ16MC101/102 family of 16-bit microcontrollers (MCUs) requires attention to a minimal set of device pin connections before proceeding with development. The following is a list of pin names, which must always be connected:

- All VDD and VSS pins  
(see [Section 2.2 “Decoupling Capacitors”](#))
- All AVDD and AVSS pins, if present on the device (regardless if ADC module is not used)  
(see [Section 2.2 “Decoupling Capacitors”](#))
- VCAP  
(see [Section 2.3 “CPU Logic Filter Capacitor Connection \(VCAP\)”](#))
- MCLR pin  
(see [Section 2.4 “Master Clear \(MCLR\) Pin”](#))
- PGECx/PGEDx pins used for In-Circuit Serial Programming™ (ICSP™) and debugging purposes  
(see [Section 2.5 “ICSP Pins”](#))
- OSC1 and OSC2 pins when external oscillator source is used  
(see [Section 2.6 “External Oscillator Pins”](#))

### 2.2 Decoupling Capacitors

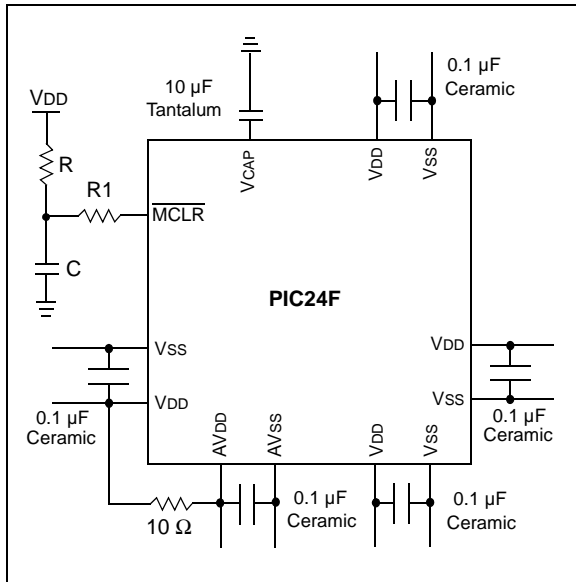
The use of decoupling capacitors on every pair of power supply pins, such as VDD, VSS, AVDD, and AVSS is required.

Consider the following criteria when using decoupling capacitors:

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

# PIC24FJ16MC101/102

**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 MCUs 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 26.0 “Electrical Characteristics”](#) for additional information.

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

## 2.4 Master Clear (MCLR) Pin

The MCLR pin provides two specific device functions:

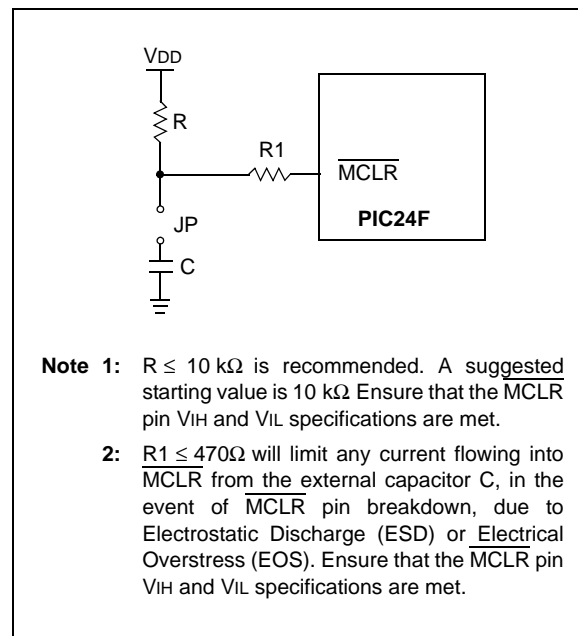
- Device Reset
- Device programming and debugging

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

For example, as shown in [Figure 2-2](#), it is recommended that the capacitor C, be isolated from the MCLR pin during programming and debugging operations.

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

**FIGURE 2-2: EXAMPLE OF MCLR PIN CONNECTIONS**



## 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. Alternately, refer to the AC/DC characteristics and timing requirements information in the “PIC24FJXXMCXXX Flash Programming Specification” for information on capacitive loading limits and pin input voltage high (V<sub>IH</sub>) and input low (V<sub>IL</sub>) requirements.

Ensure that the “Communication Channel Select” (i.e., PGECx/PGEDx pins) programmed into the device matches the physical connections for the ICSP to MPLAB® ICD 2, MPLAB ICD 3, or MPLAB REAL ICE™.

For more information on ICD 2, ICD 3, and REAL ICE connection requirements, refer to the following documents that are available on the Microchip web site.

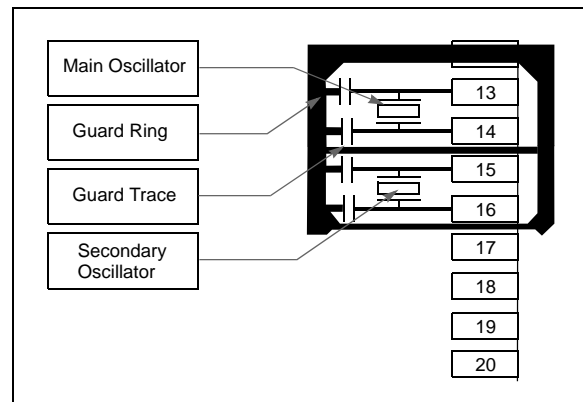
- “MPLAB® ICD 2 In-Circuit Debugger User’s Guide” (DS51331)
- “Using MPLAB® ICD 2” (poster) (DS51265)
- “MPLAB® ICD 2 Design Advisory” (DS51566)
- “Using MPLAB® ICD 3” (poster) (DS51765)
- “MPLAB® ICD 3 Design Advisory” (DS51764)
- “MPLAB® REAL ICE™ In-Circuit Debugger User’s Guide” (DS51616)
- “Using MPLAB® REAL ICE™” (poster) (DS51749)

## 2.6 External Oscillator Pins

Many MCUs 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  $4\text{ MHz} < F_{IN} < 8\text{ MHz}$  (for MSPLL mode) or  $3\text{ MHz} < F_{IN} < 8\text{ MHz}$  (for ECPLL mode) to comply with device PLL start-up conditions. HSPLL mode is not supported. This means that if the external oscillator frequency is outside this range, the application must start-up in the FRC mode first. The fixed PLL settings of 4x 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 enable the PLL, 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 analog-to-digital input pins (ANx) as “digital” pins, by setting all bits in the AD1PCFGL register.

The bits in the register that correspond to the analog-to-digital 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 analog-to-digital 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 analog-to-digital pins being recognized as analog input pins, resulting in the port value being read as a logic ‘0’, which may affect user application functionality.

## 2.9 Unused I/Os

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

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

## 3.0 CPU

- Note 1:** This data sheet summarizes the features of the PIC24FJ16MC101/102 family of devices. However, it is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 2. “CPU”** (DS39703) in the *“PIC24F Family Reference Manual”*, which is available from the Microchip web site ([www.microchip.com](http://www.microchip.com)).
- 2:** It is important to note that the specifications in **Section 26.0 “Electrical Characteristics”** of this data sheet, supercede any specifications that may be provided in PIC24F Family Reference Manual sections.
- 3:** 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 PIC24FJ16MC101/102 CPU module has a 16-bit (data) modified Harvard architecture with an enhanced instruction set and addressing modes. 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 by 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, single-cycle program loop constructs are supported using the REPEAT instruction, which is interruptible at any point.

The PIC24FJ16MC101/102 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 PIC24FJ16MC101/102 instruction set includes many addressing modes and is designed for optimum C compiler efficiency. For most instructions, PIC24FJ16MC101/102 devices are capable of executing a data (or program data) memory read, a working register (data) read, a data memory write, and a program (instruction) memory read per instruction cycle. As a result, three parameter instructions can be supported, allowing  $A + B = C$  operations to be executed in a single cycle.

A block diagram of the CPU is shown in [Figure 3-1](#), and the programmer’s model for the PIC24FJ16MC101/102 is shown in [Figure 3-2](#).

### 3.1 Data Addressing Overview

The data space can be linearly addressed as 32K words or 64 Kbytes using an Address Generation Unit (AGU). The upper 32 Kbytes of the data space memory map can optionally be mapped into program space at any 16K program word boundary defined by the 8-bit Program Space Visibility Page register (PSVPAG). The program to data space mapping feature lets any instruction access program space as if it were data space.

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

### 3.2 Special MCU Features

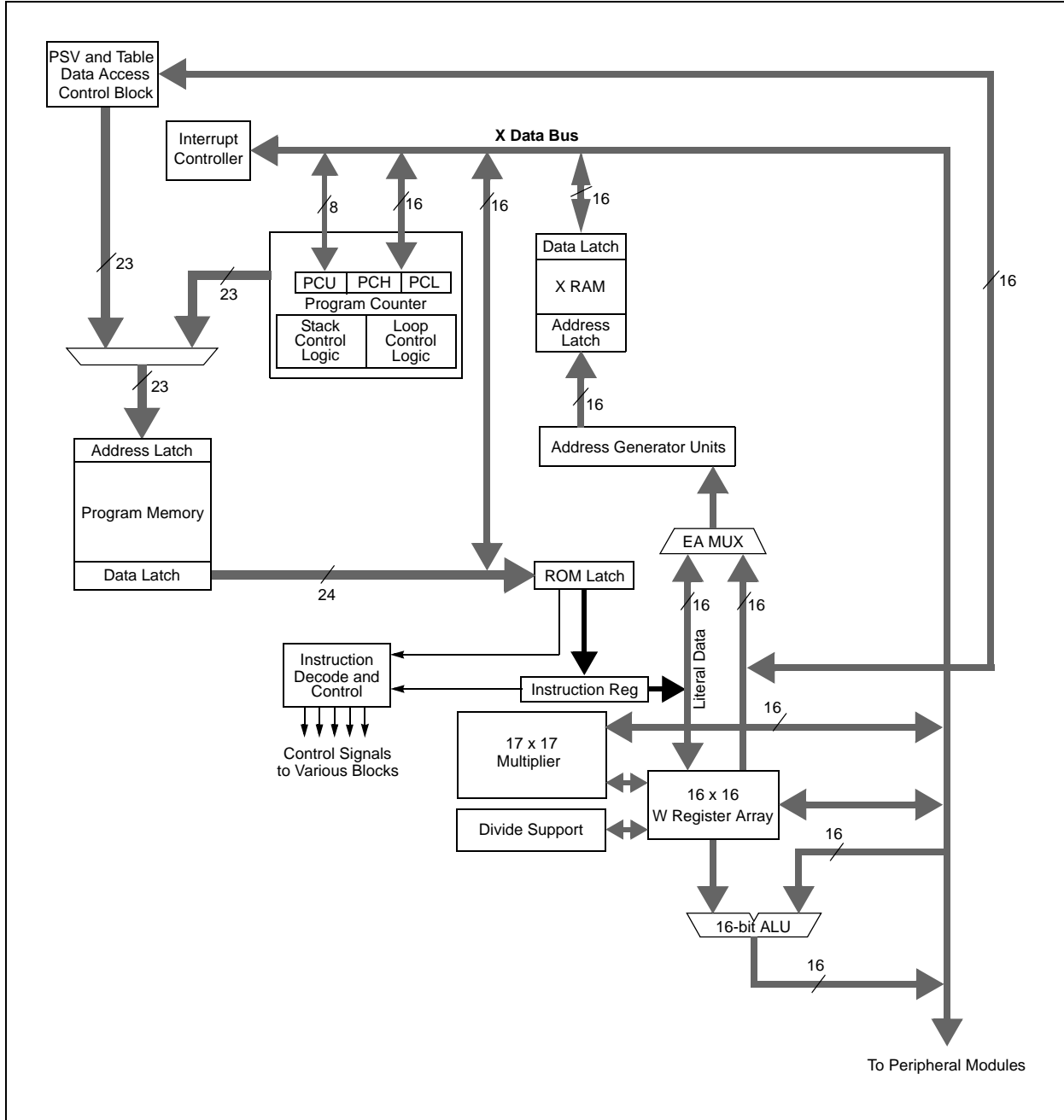
The PIC24FJ16MC101/102 features a 17-bit by 17-bit, single-cycle multiplier. The multiplier can perform signed, unsigned and mixed-sign multiplication. Using a 17-bit by 17-bit multiplier for 16-bit by 16-bit multiplication makes mixed-sign multiplication possible.

The PIC24FJ16MC101/102 supports 16/16 and 32/16 integer divide operations. 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.

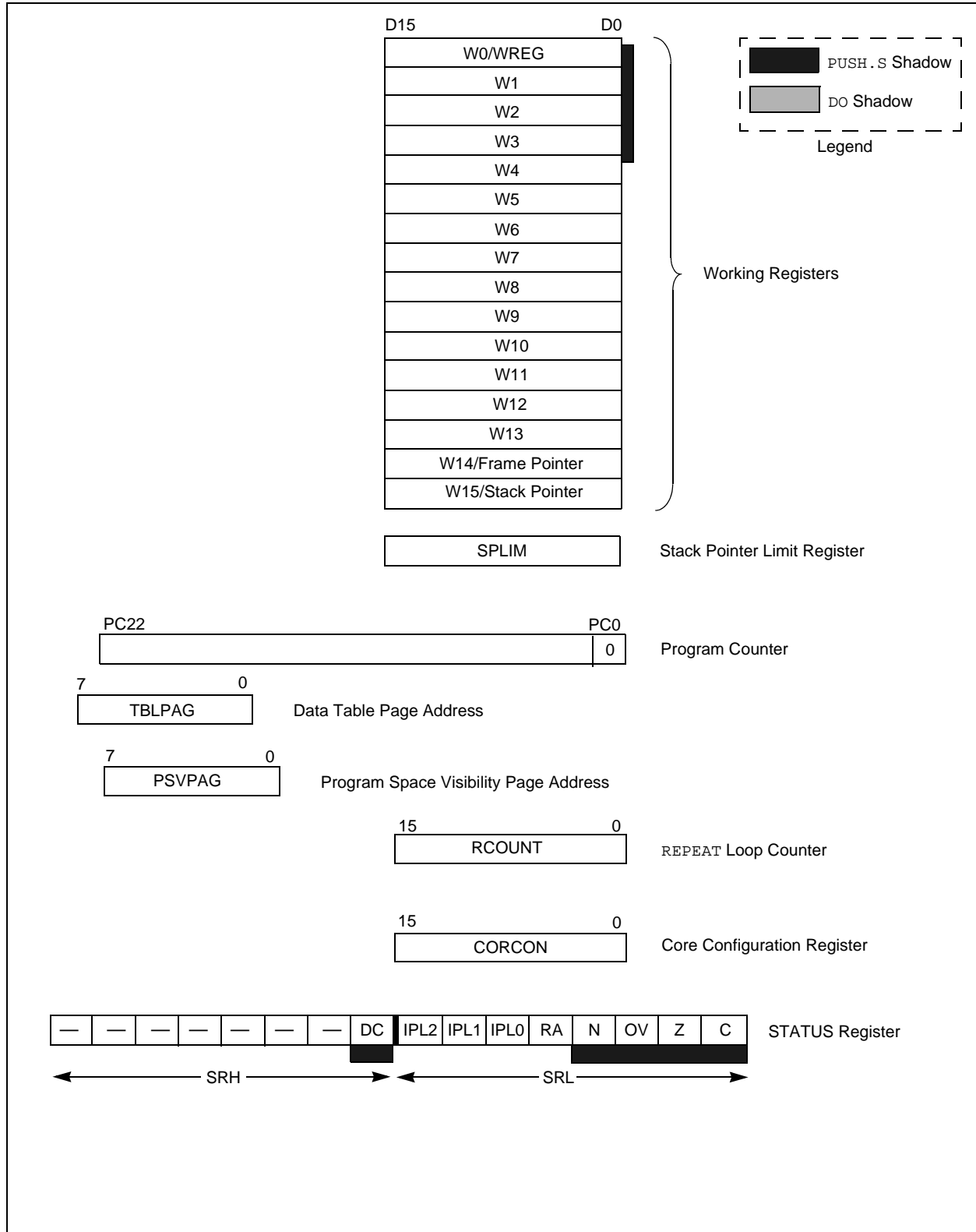
A multi-bit data shifter is used to perform up to a 16-bit, left or right shift in a single cycle.

# PIC24FJ16MC101/102

FIGURE 3-1: PIC24FJ16MC101/102 CPU CORE BLOCK DIAGRAM



**FIGURE 3-2: PIC24FJ16MC101/102 PROGRAMMER'S MODEL**



# PIC24FJ16MC101/102

## 3.3 CPU Control Registers

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

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

R/W-0 <sup>(1)</sup>	R/W-0 <sup>(2)</sup>	R/W-0 <sup>(2)</sup>	R-0	R/W-0	R/W-0	R/W-0	R/W-0
IPL<2:0> <sup>(2)</sup>			RA	N	OV	Z	C
bit 7							bit 0

#### Legend:

C = 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 15-9 **Unimplemented:** Read as '0'

bit 8 **DC:** MCU ALU Half Carry/Borrow bit

- 1 = A carry-out from the 4th low-order bit (for byte-sized data) or 8th low-order bit (for word-sized data) of the result occurred
- 0 = No carry-out from the 4th low-order bit (for byte-sized data) or 8th low-order bit (for word-sized data) of the result occurred

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

- 111 = CPU Interrupt Priority Level is 7 (15), user interrupts disabled
- 110 = CPU Interrupt Priority Level is 6 (14)
- 101 = CPU Interrupt Priority Level is 5 (13)
- 100 = CPU Interrupt Priority Level is 4 (12)
- 011 = CPU Interrupt Priority Level is 3 (11)
- 010 = CPU Interrupt Priority Level is 2 (10)
- 001 = CPU Interrupt Priority Level is 1 (9)
- 000 = CPU Interrupt Priority Level is 0 (8)

bit 4 **RA:** REPEAT Loop Active bit

- 1 = REPEAT loop in progress
- 0 = REPEAT loop not in progress

bit 3 **N:** MCU ALU Negative bit

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

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

- This bit is used for signed arithmetic (2's complement). It indicates an overflow of the magnitude which causes the sign bit to change state.
- 1 = Overflow occurred for signed arithmetic (in this arithmetic operation)
- 0 = No overflow occurred

bit 1 **Z:** MCU ALU Zero bit

- 1 = An operation which affects the Z bit has set it at some time in the past
- 0 = The most recent operation which affects the Z bit has cleared it (i.e., a non-zero result)

bit 0 **C:** MCU ALU Carry/Borrow bit

- 1 = A carry-out from the Most Significant bit (MSb) of the result occurred
- 0 = No carry-out from the Most Significant bit of the result occurred

**Note 1:** 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.

**2:** 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	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15						bit 8	

U-0	U-0	U-0	U-0	R/C-0	R/W-0	U-0	U-0
—	—	—	—	IPL3 <sup>(1)</sup>	PSV	—	—
bit 7						bit 0	

<b>Legend:</b>	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 15-4      **Unimplemented:** Read as '0'
- bit 3        **IPL3:** CPU Interrupt Priority Level Status bit 3<sup>(1)</sup>  
                  1 = CPU interrupt priority level is greater than 7  
                  0 = CPU interrupt priority level is 7 or less
- bit 2        **PSV:** Program Space Visibility in Data Space Enable bit  
                  1 = Program space visible in data space  
                  0 = Program space not visible in data space
- bit 1-0      **Unimplemented:** Read as '0'

**Note 1:** The IPL3 bit is concatenated with the IPL<2:0> bits (SR<7:5>) to form the CPU interrupt priority level.

## 3.4 Arithmetic Logic Unit (ALU)

The PIC24FJ16MC101/102 ALU is 16 bits wide and is capable of addition, subtraction, bit shifts, and logic operations. Unless otherwise mentioned, arithmetic operations are 2's complement in nature. Depending on the operation, the ALU may affect the values of the Carry (C), Zero (Z), Negative (N), Overflow (OV), and Digit Carry (DC) Status bits in the SR register. The C and DC Status bits operate as Borrow and Digit Borrow bits, respectively, for subtraction operations.

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

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

The PIC24FJ16MC101/102 CPU incorporates hardware support for both multiplication and division. This includes a dedicated hardware multiplier and support hardware for 16-bit divisor division.

### 3.4.1 MULTIPLIER

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

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

### 3.4.2 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. Sixteen-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.

### 3.4.3 MULTI-BIT DATA SHIFTER

The multi-bit data shifter is capable of performing up to 16-bit arithmetic or logic right shifts, or up to 16-bit left shifts in a single cycle. The source can be either a working register or a memory location.

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.

## 4.0 MEMORY ORGANIZATION

**Note 1:** This data sheet summarizes the features of the PIC24FJ16MC101/102 family of devices. However, it is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 3. “Data Memory”** (DS39717) and **Section 4. “Program Memory”** (DS39715) in the *“PIC24F Family Reference Manual”*, which are available from the Microchip web site ([www.microchip.com](http://www.microchip.com)).

**2:** It is important to note that the specifications in **Section 26.0 “Electrical Characteristics”** of this data sheet, supercede any specifications that may be provided in PIC24F Family Reference Manual sections.

The PIC24FJ16MC101/102 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.

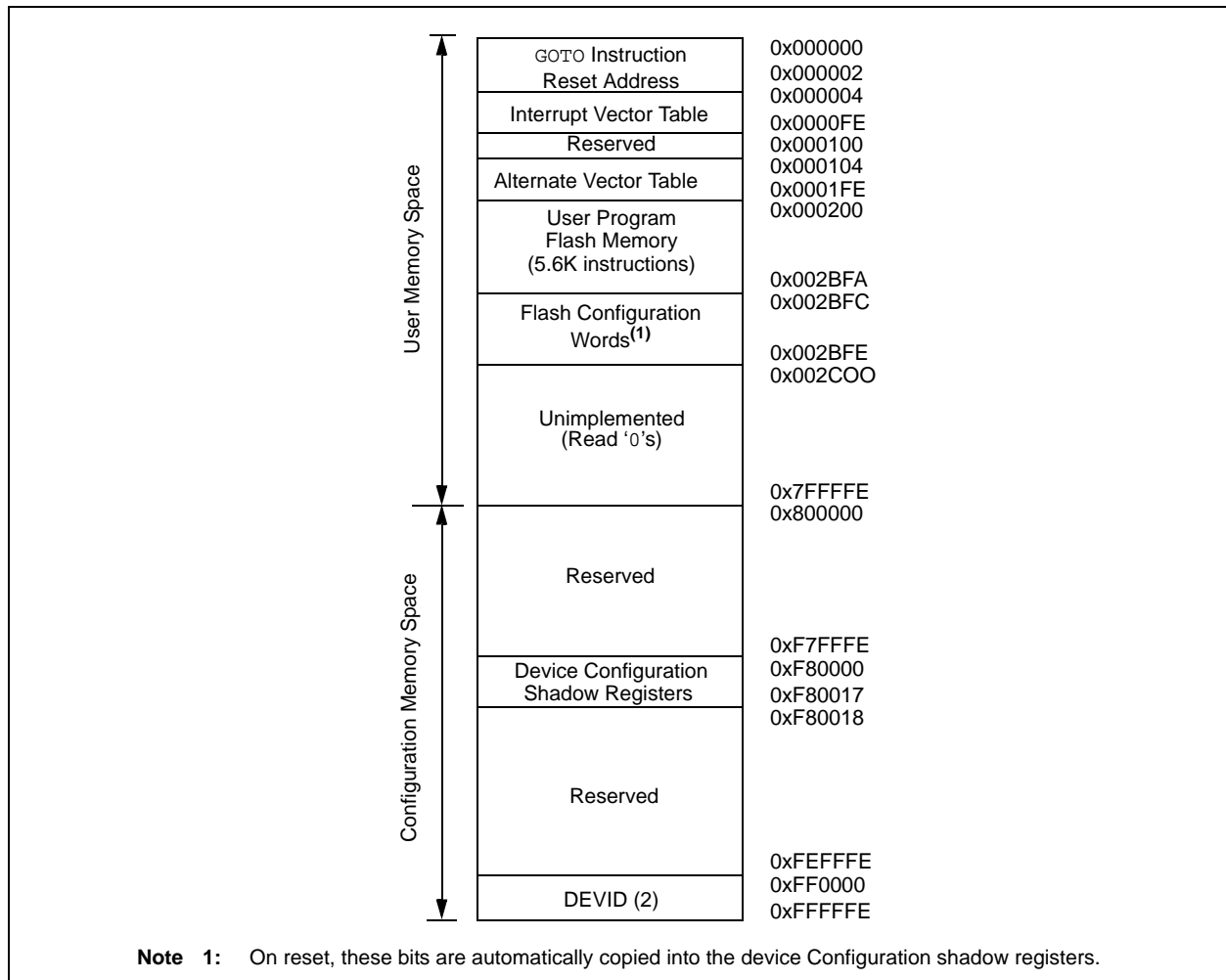
### 4.1 Program Address Space

The program address memory space of the PIC24FJ16MC101/102 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 4.4 “Interfacing Program and Data Memory Spaces”**.

User application access to the program memory space is restricted to the lower half of the address range (0x000000 to 0x7FFFFFFF). The exception is the use of TBLRD/TBLWT operations, which use TBLPAG<7> to permit access to the Configuration bits and Device ID sections of the configuration memory space.

The memory map for the PIC24FJ16MC101/102 family of devices is shown in **Figure 4-1**.

**FIGURE 4-1: PROGRAM MEMORY MAP FOR PIC24FJ16MC101/102 DEVICES**



# PIC24FJ16MC101/102

## 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).

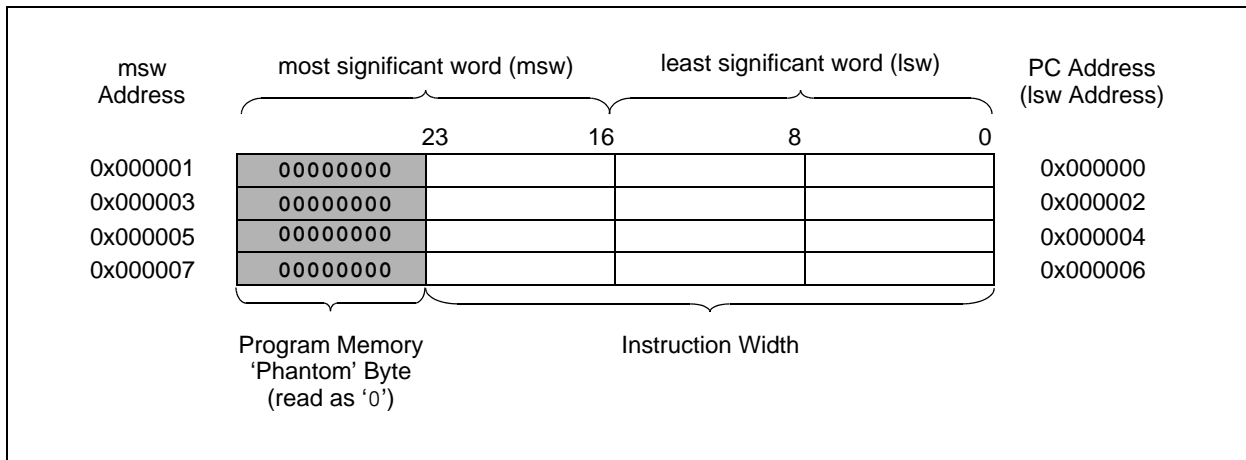
Program memory addresses are always word-aligned on the lower word, and addresses are incremented or decremented by two during code execution. This arrangement provides compatibility with data memory space addressing and makes data in the program memory space accessible.

## 4.1.2 INTERRUPT AND TRAP VECTORS

All PIC24FJ16MC101/102 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.

PIC24FJ16MC101/102 devices also have two interrupt vector tables, located from 0x000004 to 0x0000FF and 0x000100 to 0x0001FF. These vector tables allow each of the device interrupt sources to be handled by separate Interrupt Service Routines (ISRs). A more detailed discussion of the interrupt vector tables is provided in Section 7.1 "Interrupt Vector Table".

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



## 4.2 Data Address Space

The PIC24FJ16MC101/102 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  $EA_{\langle 15 \rangle} = 0$ ) is used for implemented memory addresses, while the upper half ( $EA_{\langle 15 \rangle} = 1$ ) is reserved for the Program Space Visibility area (see [Section 4.4.3 “Reading Data from Program Memory Using Program Space Visibility”](#)).

Microchip PIC24FJ16MC101/102 devices implement up to 1 Kbyte of data memory. Should an EA point to a location outside of this area, an all-zero word or byte will be returned.

### 4.2.1 DATA SPACE WIDTH

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

### 4.2.2 DATA MEMORY ORGANIZATION AND ALIGNMENT

To maintain backward compatibility with PIC<sup>®</sup> MCU devices and improve data space memory usage efficiency, the PIC24FJ16MC101/102 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 decoding 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 in progress is completed. If the error occurred on a write, the instruction is executed but the write does not occur. In either case, a trap is then executed, allowing the system and/or user application to examine the machine state prior to execution of the address Fault.

All byte loads into any W register are loaded into the LSB. The MSB is not modified.

A sign-extend instruction ( $SE$ ) is provided to allow user applications to translate 8-bit signed data to 16-bit signed values. Alternately, 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 PIC24FJ16MC101/102 core and peripheral modules for controlling the operation of the device.

SFRs are distributed among the modules that they control, and are generally grouped together by module. Much of the SFR space contains unused addresses; these are read as '0'.

**Note:** The actual set of peripheral features and interrupts varies by the device. 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  $MOV$  class of instructions, which support Memory Direct Addressing mode with a 16-bit address field, or by using Indirect Addressing mode with a working register as an address pointer.

# PIC24FJ16MC101/102

**FIGURE 4-3: DATA MEMORY MAP FOR PIC24FJ16MC101/102 DEVICES WITH 1 KB RAM**

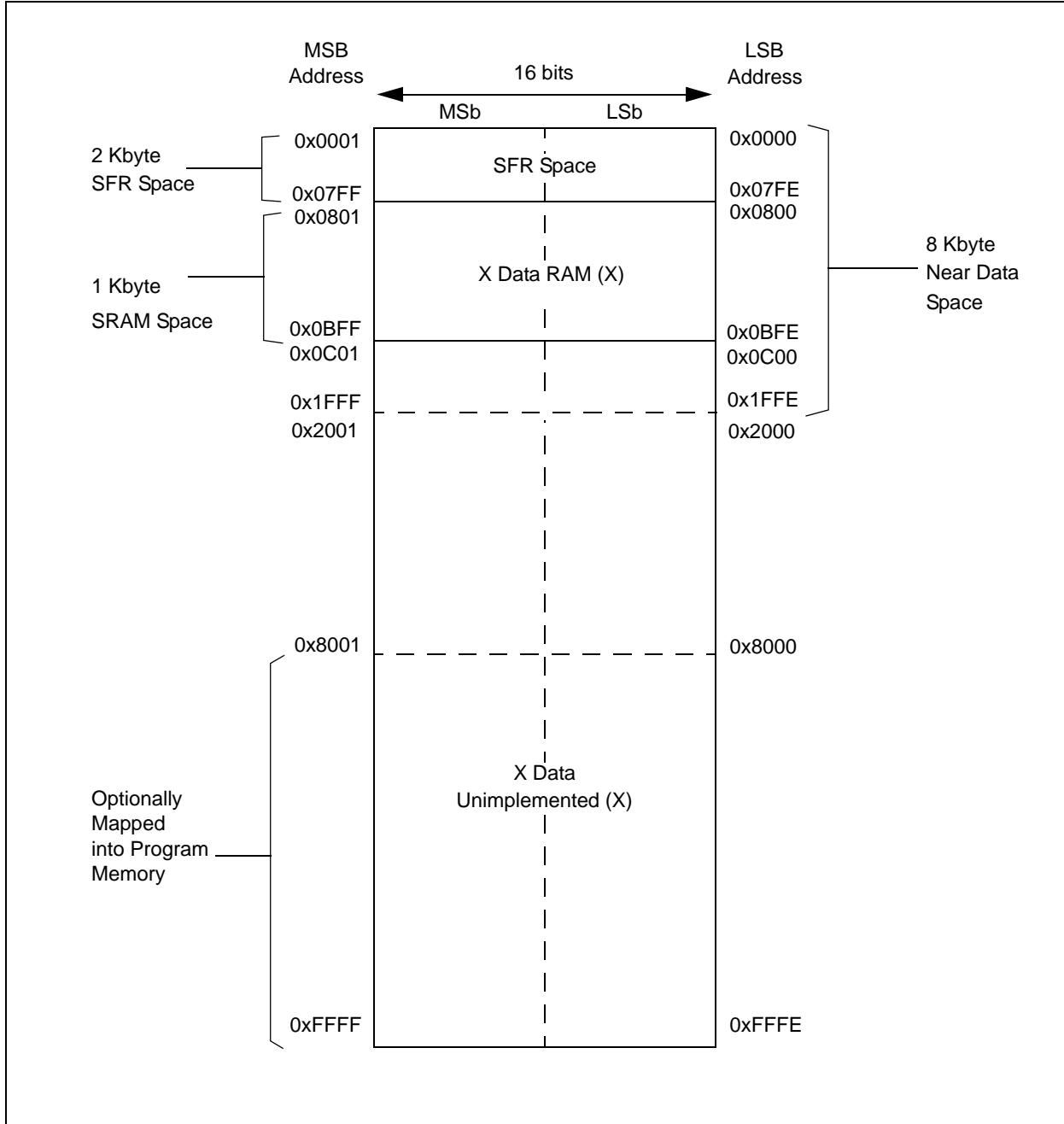


TABLE 4-1: CPU CORE REGISTERS MAP

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																xxxx
WREG1	0002	Working Register 1																xxxx
WREG2	0004	Working Register 2																xxxx
WREG3	0006	Working Register 3																xxxx
WREG4	0008	Working Register 4																xxxx
WREG5	000A	Working Register 5																xxxx
WREG6	000C	Working Register 6																xxxx
WREG7	000E	Working Register 7																xxxx
WREG8	0010	Working Register 8																xxxx
WREG9	0012	Working Register 9																xxxx
WREG10	0014	Working Register 10																xxxx
WREG11	0016	Working Register 11																xxxx
WREG12	0018	Working Register 12																xxxx
WREG13	001A	Working Register 13																xxxx
WREG14	001C	Working Register 14																xxxx
WREG15	001E	Working Register 15																0800
SPLIM	0020	Stack Pointer Limit Register																xxxx
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
SR	0042	—	—	—	—	—	—	—	DC	IPL2	IPL1	IPL0	RA	N	OV	Z	C	0000
CORCON	0044	—	—	—	—	—	—	—	—	—	—	—	—	IPL3	PSV	—	—	0020
DISICNT	0052	—	—	Disable Interrupts Counter Register														0000

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

**TABLE 4-2: CHANGE NOTIFICATION REGISTER MAP FOR PIC24FJ16MC101 DEVICES**

SFR Name	SFR Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
CNEN1	0060	—	CN14IE	CN13IE	CN12IE	CN11IE	—	—	—	—	—	CN5IE	CN4IE	CN3IE	CN2IE	CN1IE	CN0IE	0000
CNEN2	0062	—	CN30IE	CN29IE	—	—	—	—	—	CN23IE	CN22IE	CN21IE	—	—	—	—	—	0000
CNPU1	0068	—	CN14PUE	CN13PUE	CN12PUE	CN11PUE	—	—	—	—	—	CN5PUE	CN4PUE	CN3PUE	CN2PUE	CN1PUE	CN0PUE	0000
CNPU2	006A	—	CN30PUE	CN29PUE	—	—	—	—	—	CN23PUE	CN22PUE	CN21PUE	—	—	—	—	—	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 PIC24FJ16MC102 DEVICES**

SFR Name	SFR Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
CNEN1	0060	CN15IE	CN14IE	CN13IE	CN12IE	CN11IE	—	—	—	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 4-4: 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	—	—	—	—	—	—	—	—	—	—	MATHERR	ADDRERR	STKERR	OSCFAIL	—	0000
INTCON2	0082	ALTIPT	DISI	—	—	—	—	—	—	—	—	—	—	—	INT2EP	INT1EP	INT0EP	0000
IFS0	0084	—	—	AD1IF	U1TXIF	U1RXIF	SPI1IF	SPI1EIF	T3IF	T2IF	OC2IF	IC2IF	—	T1IF	OC1IF	IC1IF	INT0IF	0000
IFS1	0086	—	—	INT2IF	—	—	—	—	—	—	—	—	INT1IF	CNIF	CMIF	MI2C1IF	SI2C1IF	0000
IFS2	0088	—	—	—	—	—	—	—	—	—	—	IC3IF	—	—	—	—	—	0000
IFS3	008A	FLTA1IF	RTCIF	—	—	—	—	PWM1IF	—	—	—	—	—	—	—	—	—	0000
IFS4	008C	—	—	CTMUIF	—	—	—	—	—	—	—	—	—	—	—	U1EIF	FLT1BIF	0000
IEC0	0094	—	—	AD1IE	U1TXIE	U1RXIE	SPI1IE	SPI1EIE	T3IE	T2IE	OC2IE	IC2IE	—	T1IE	OC1IE	IC1IE	INT0IE	0000
IEC1	0096	—	—	INT2IE	—	—	—	—	—	—	—	—	INT1IE	CNIE	CMIE	MI2C1IE	SI2C1IE	0000
IEC2	0098	—	—	—	—	—	—	—	—	—	—	IC3IE	—	—	—	—	—	0000
IEC3	009A	FLTA1IE	RTCIE	—	—	—	—	PWM1IE	—	—	—	—	—	—	—	—	—	0000
IEC4	009C	—	—	CTMUIE	—	—	—	—	—	—	—	—	—	—	—	U1EIE	FLT1BIE	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>			—	CMIP<2:0>			—	MI2C1IP<2:0>		—	SI2C1IP<2:0>			4444
IPC5	00AE	—	—	—	—	—	—	—	—	—	—	—	—	—	INT1IP<2:0>			0004
IPC7	00B2	—	—	—	—	—	—	—	—	—	—	INT2IP<2:0>		—	—	—	—	0040
IPC9	00B6	—	—	—	—	—	—	—	—	—	—	IC3IP<2:0>		—	—	—	—	0040
IPC14	00C0	—	—	—	—	—	—	—	—	—	—	PWM1IP<2:0>		—	—	—	—	0040
IPC15	00C2	—	—	FLTA1IP<2:0>			—	RTCIP<2:0>			—	—	—	—	—	—	—	4400
IPC16	00C4	—	—	—	—	—	—	—	—	—	—	U1EIP<2:0>		—	FLT1BIP<2:0>			0040
IPC19	00CA	—	—	—	—	—	—	—	—	—	—	CTMUIP<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	
IC3BUF	0148	Input 3 Capture Register																xxxx
IC3CON	014A	—	—	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: 6-OUTPUT PWM1 REGISTER MAP**

SFR Name	Addr.	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset State	
P1TCON	01C0	PTEN	—	PTSIDL	—	—	—	—	—	PTOPS<3:0>			PTCKPS<1:0>		PTMOD<1:0>			0000 0000 0000 0000	
P1TMR	01C2	PTDIR	PWM Timer Count Value Register															0000 0000 0000 0000	
P1TPER	01C4	—	PWM Time Base Period Register															0111 1111 1111 1111	
P1SECOMP	01C6	SEVTDIR	PWM Special Event Compare Register															0000 0000 0000 0000	
PWM1CON1	01C8	—	—	—	—	—	—	PMOD3	PMOD2	PMOD1	—	PEN3H	PEN2H	PEN1H	—	PEN3L	PEN2L	PEN1L	0000 0000 0000 0000
PWM1CON2	01CA	—	—	—	—	—	SEVOPS<3:0>			—	—	—	—	—	IUE	OSYNC	UDIS	0000 0000 0000 0000	
P1DTCON1	01CC	DTBPS<1:0>		DTB<5:0>				DTAPS<1:0>			DTA<5:0>					0000 0000 0000 0000			
P1DTCON2	01CE	—	—	—	—	—	—	—	—	—	—	DTS3A	DTS3I	DTS2A	DTS2I	DTS1A	DTS1I	0000 0000 0000 0000	
P1FLTAON	01D0	—	—	FAOV3H	FAOV3L	FAOV2H	FAOV2L	FAOV1H	FAOV1L	FLTAM	—	—	—	—	FAEN3	FAEN2	FAEN1	0000 0000 0000 0111	
P1FLTBON	01D2	—	—	FBOV3H	FBOV3L	FBOV2H	FBOV2L	FBOV1H	FBOV1L	FLTBM	—	—	—	—	FBEN3	FBEN2	FBEN1	0000 0000 0000 0111	
P1OVDCON	01D4	—	—	POVD3H	POVD3L	POVD2H	POVD2L	POVD1H	POVD1L	—	—	POUT3H	POUT3L	POUT2H	POUT2L	POUT1H	POUT1L	0011 1111 0000 0000	
P1DC1	01D6	PWM Duty Cycle 1 Register															0000 0000 0000 0000		
P1DC2	01D8	PWM Duty Cycle 2 Register															0000 0000 0000 0000		
P1DC3	01DA	PWM Duty Cycle 3 Register															0000 0000 0000 0000		
PWM1KEY	01DE	PWMKEY<15:0>															0000 0000 0000 0000		

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

**TABLE 4-9: 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 4-10: 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	RTSMO	—	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	RIDL	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 4-11: 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	—	SPIIDL	—	—	—	—	—	—	SPIROV	—	—	—	—	SPIBF	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 4-12: ADC1 REGISTER MAP FOR PIC24FJ16MC101 DEVICES

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
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	—	—	—	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	—	—	—	—	—	—	—	—	—	—	—	PCFG3	PCFG2	PCFG1	PCFG0	0000	
AD1CSSL	0330	—	—	—	—	—	—	—	—	—	—	—	CSS3	CSS2	CSS1	CSS0	0000	

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

TABLE 4-13: ADC1 REGISTER MAP FOR PIC24FJ16MC102 DEVICES

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
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	—	—	—	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	—	—	—	—	—	—	—	—	—	PCFG5	PCFG4	PCFG3	PCFG2	PCFG1	PCFG0	0000	
AD1CSSL	0330	—	—	—	—	—	—	—	—	—	CSS5	CSS4	CSS3	CSS2	CSS1	CSS0	0000	

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

**TABLE 4-14: CTMU REGISTER MAP**

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets	
CTMUCON1	033A	CTMUEN	—	CTMUSIDL	TGEN	EDGEN	EDGSEQEN	IDISSEN	CTTRIG	—	—	—	—	—	—	—	—	0000	
CTMUCON2	033C	EDG1MOD	EDG1POL	EDG1SEL<3:0>				EDG2STAT	EDG1STAT	EDG2MOD	EDG2POL	EDG2SEL<3:0>				—	—	0000	
CTMUICON	033E	ITRIM<5:0>					IRNG<1:0>					—	—	—	—	—	—	—	0000

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

**TABLE 4-15: REAL-TIME CLOCK AND CALENDAR REGISTER MAP**

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets	
ALRMVAL	0620	Alarm Value Register Window based on APTR<1:0>																	xxxx
ALCFGRPT	0622	ALRMEN	CHIME	AMASK<3:0>				ALRMPTR<1:0>			ARPT<7:0>							0000	
RTCVAL	0624	RTCC Value Register Window based on RTCPTR<1:0>																	xxxx
RCFGCAL	0626	RTCEN	—	RTCWREN	RTCSYNC	HALFSEC	RTCOE	RTCPTR<1:0>				CAL<7:0>							0000

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

**TABLE 4-16: PAD CONFIGURATION REGISTER MAP**

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
PADCFG1	02FC	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RTSECSSEL	—	0000

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

TABLE 4-17: COMPARATOR REGISTER MAP

File Name	Addr.	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
CMSTAT	0650	CMSIDL	—	—	—	—	C3EVT	C2EVT	C1EVT	—	—	—	—	—	C3OUT	C2OUT	C1OUT	0000
CVRCON	0652	—	—	—	—	—	VREFSEL	BGSEL<1:0>	CVREN	CVROE	CVRR	—	—	—	CVR<3:0>	—	—	0000
CM1CON	0654	CON	COE	CPOL	—	—	—	CEVT	COUT	—	—	—	—	—	—	—	CCH<1:0>	0000
CM1MSKSR	0656	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
CM1MSKCON	0658	HLMS	—	OCEN	OCNEN	OBEN	OBNEN	OAEN	OANEN	NAGS	PAGS	ACEN	ACNEN	ABEN	ABNEN	AAEN	AANEN	0000
CM1FLTR	065A	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
CM2CON	065C	CON	COE	CPOL	—	—	—	CEVT	COUT	—	—	—	—	—	—	—	CCH<1:0>	0000
CM2MSKSR	065E	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
CM2MSKCON	0660	HLMS	—	OCEN	OCNEN	OBEN	OBNEN	OAEN	OANEN	NAGS	PAGS	ACEN	ACNEN	ABEN	ABNEN	AAEN	AANEN	0000
CM2FLTR	0662	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
CM3CON	0664	CON	COE	CPOL	—	—	—	CEVT	COUT	—	—	—	—	—	—	—	CCH<1:0>	0000
CM3MSKSR	0666	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
CM3MSKCON	0668	HLMS	—	OCEN	OCNEN	OBEN	OBNEN	OAEN	OANEN	NAGS	PAGS	ACEN	ACNEN	ABEN	ABNEN	AAEN	AANEN	0000
CM3FLTR	066A	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000

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

TABLE 4-18: 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	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	1F00
RPINR1	0682	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	001F
RPINR3	0686	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	1F1F
RPINR7	068E	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	1F1F
RPINR8	0690	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	001F
RPINR11	0696	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	001F
RPINR18	06A4	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	1F1F
RPINR21	06AA	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	001F

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



**TABLE 4-19: PERIPHERAL PIN SELECT OUTPUT REGISTER MAP FOR PIC24FJ16MC101 DEVICES**

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
RPOR0	06C0	—	—	—	RP1R<4:0>				—	—	—	RP0R<4:0>				0000		
RPOR2	06C4	—	—	—	—	—	—	—	—	—	—	RP4R<4:0>				0000		
RPOR3	06C6	—	—	—	RP7R<4:0>				—	—	—	—	—	—	—	—	—	0000
RPOR4	06C8	—	—	—	RP9R<4:0>				—	—	—	RP8R<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 4-20: PERIPHERAL PIN SELECT OUTPUT REGISTER MAP FOR PIC24FJ16MC102 DEVICES**

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
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 4-21: PORTA REGISTER MAP**

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
TRISA	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 4-22: PORTB REGISTER MAP FOR PIC24FJ16MC101 DEVICES**

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
TRISB	02C8	TRISB15	TRISB14	TRISB13	TRISB12	—	—	TRISB9	TRISB8	TRISB7	—	—	TRISB4	—	—	TRISB1	TRISB0	F393
PORTB	02CA	RB15	RB14	RB13	RB12	—	—	RB9	RB8	RB7	—	—	RB4	—	—	RB1	RB0	xxxx
LATB	02CC	LATB15	LATB14	LATB13	LATB12	—	—	LATB9	LATB8	LATB7	—	—	LATB4	—	—	LATB1	LATB0	xxxx
ODCB	02CE	ODCB15	ODCB14	ODCB13	ODCB12	—	—	ODCB9	ODCB8	ODCB7	—	—	ODCB4	—	—	ODCB1	ODCB0	0000

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

**TABLE 4-23: PORTB REGISTER MAP FOR PIC24FJ16MC102 DEVICES**

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
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 4-24: 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 <sup>(1)</sup>
OSCCON	0742	—	COSC<2:0>			—	NOSC<2:0>			CLKLOCK	IOLOCK	LOCK	—	CF	—	LPOSCEN	OSWEN	0300 <sup>(2)</sup>
CLKDIV	0744	ROI	DOZE<2:0>			DOZEN			FRCDIV<2:0>			—	—	—	—	—	—	3040
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.  
**Note 2:** OSCCON register Reset values dependent on the FOSC Configuration bits and by type of Reset.

**TABLE 4-25: 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 <sup>(1)</sup>	
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 4-26: 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	—	PWM1MD	—	I2C1MD	—	U1MD	—	SP1MD	—	—	AD1MD	0000
PMD2	0772	—	—	—	—	—	IC3MD	IC2MD	IC1MD	—	—	—	—	—	—	OC2MD	OC1MD	0000
PMD3	0774	—	—	—	—	—	CMPMD	RTCCMD	—	—	—	—	—	—	—	—	—	0000
PMD4	0776	—	—	—	—	—	—	—	—	—	—	—	—	—	CTMUMD	—	—	0000

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

# PIC24FJ16MC101/102

## 4.2.5 SOFTWARE STACK

In addition to its use as a working register, the W15 register in the PIC24FJ16MC101/102 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.

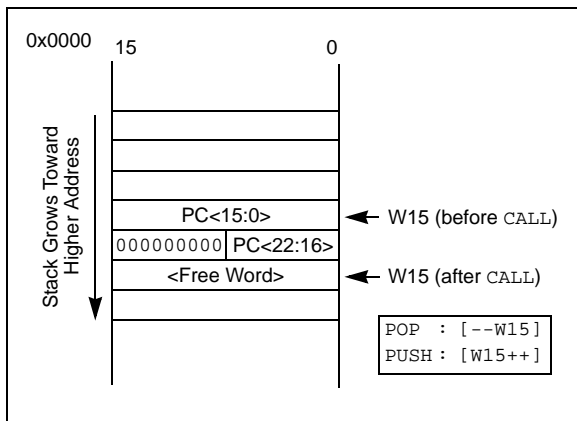
The Stack Pointer Limit register (SPLIM) associated with the Stack Pointer sets an upper address boundary for the stack. SPLIM is uninitialized at Reset. As is the case for the Stack Pointer, SPLIM<0> is forced to '0' because all stack operations must be word aligned.

Whenever an EA is generated using W15 as a source or destination pointer, the resulting address is compared with the value in SPLIM. If the contents of the Stack Pointer (W15) and the SPLIM register are equal and a push operation is performed, a stack error trap will not occur. However, 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 0x0C00 in RAM, initialize the SPLIM with the value 0x0BFE.

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 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**



## 4.2.6 DATA RAM PROTECTION FEATURE

The PIC24FXXXX 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 BS) is accessible only from the Boot Segment Flash code, when enabled. SSRAM (Secure RAM segment for RAM) is accessible only from the Secure Segment Flash code, when enabled. See Table 4-1 for an overview of the BSRAM and SSRAM SFRs.

## 4.3 Instruction Addressing Modes

The addressing modes shown in Table 4-27 form the basis of the addressing modes that are optimized to support the specific features of individual instructions. The addressing modes provided in the MAC class of instructions differ from those provided in 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 of the addressing modes given above. Individual instructions can support different subsets of these addressing modes.

**TABLE 4-27: 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 INSTRUCTIONS

Move instructions provide a greater degree of addressing flexibility than other instructions. In addition to the addressing modes supported by most MCU instructions, move 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 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 OTHER INSTRUCTIONS

In addition to 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 Interfacing Program and Data Memory Spaces

The PIC24FJ16MC101/102 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 PIC24FJ16MC101/102 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 lookups from a large table of static data. The application can only access the lsw of the program word.

# PIC24FJ16MC101/102

## 4.4.1 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 MSb 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 MSb 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.

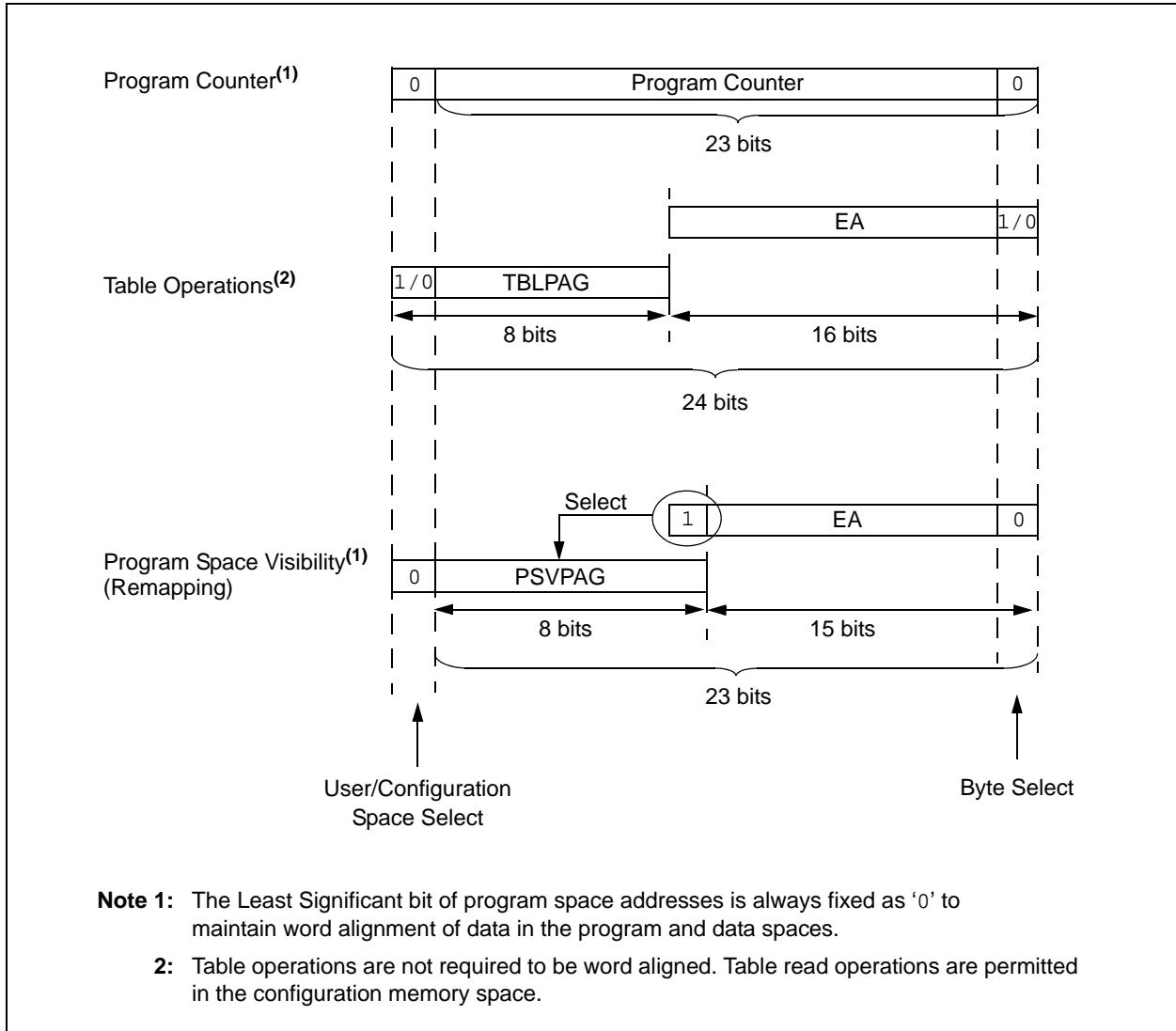
Table 4-28 and Figure 4-5 show how the program EA is created for table operations and remapping accesses from the data EA.

**TABLE 4-28: 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 xxx				
	Configuration	TBLPAG<7:0>		Data EA<15:0>		
		1xxx xxxx xxxx xxxx xxxx xxx				
Program Space Visibility (Block Remap/Read)	User	0	PSVPAG<7:0>		Data EA<14:0> <sup>(1)</sup>	
		0	xxxx xxxx xxx xxxx xxxx xxx			

**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 4-5: DATA ACCESS FROM PROGRAM SPACE ADDRESS GENERATION**



# PIC24FJ16MC101/102

## 4.4.2 DATA ACCESS FROM PROGRAM MEMORY USING TABLE INSTRUCTIONS

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

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

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

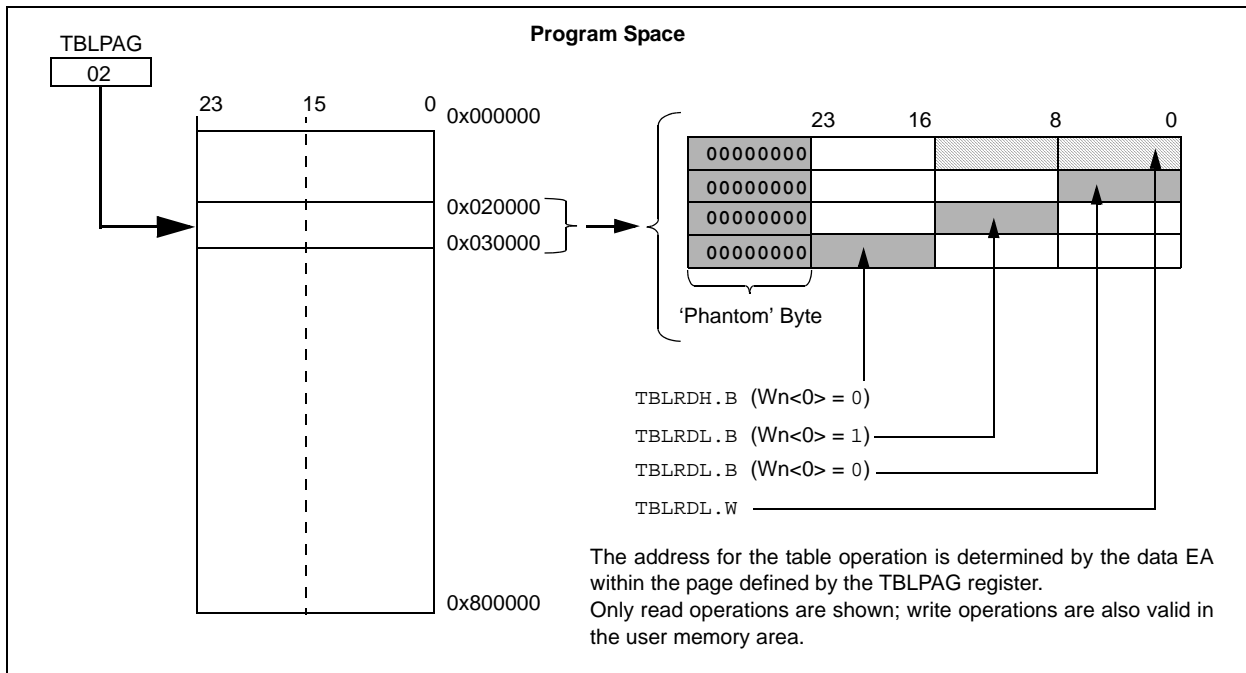
- `TBLRDL` (Table Read Low):
  - In Word mode, this instruction maps the lower word of the program space location ( $P<15:0>$ ) to a data address ( $D<15:0>$ ).
  - In Byte mode, either the upper or lower byte of the lower program word is mapped to the lower byte of a data address. The upper byte is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

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

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 4-6: ACCESSING PROGRAM MEMORY WITH TABLE INSTRUCTIONS**





## 4.4.3 READING DATA FROM PROGRAM MEMORY USING PROGRAM SPACE VISIBILITY

The upper 32 Kbytes of data space may optionally be mapped into any 16K word page of the program space. This option provides transparent access to stored constant data from the data space without the need to use special instructions (such as TBLRD and TBLRDH).

Program space access through the data space occurs if the MSb 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 0x8000 and higher maps directly into a corresponding program memory address (see Figure 4-7), 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.

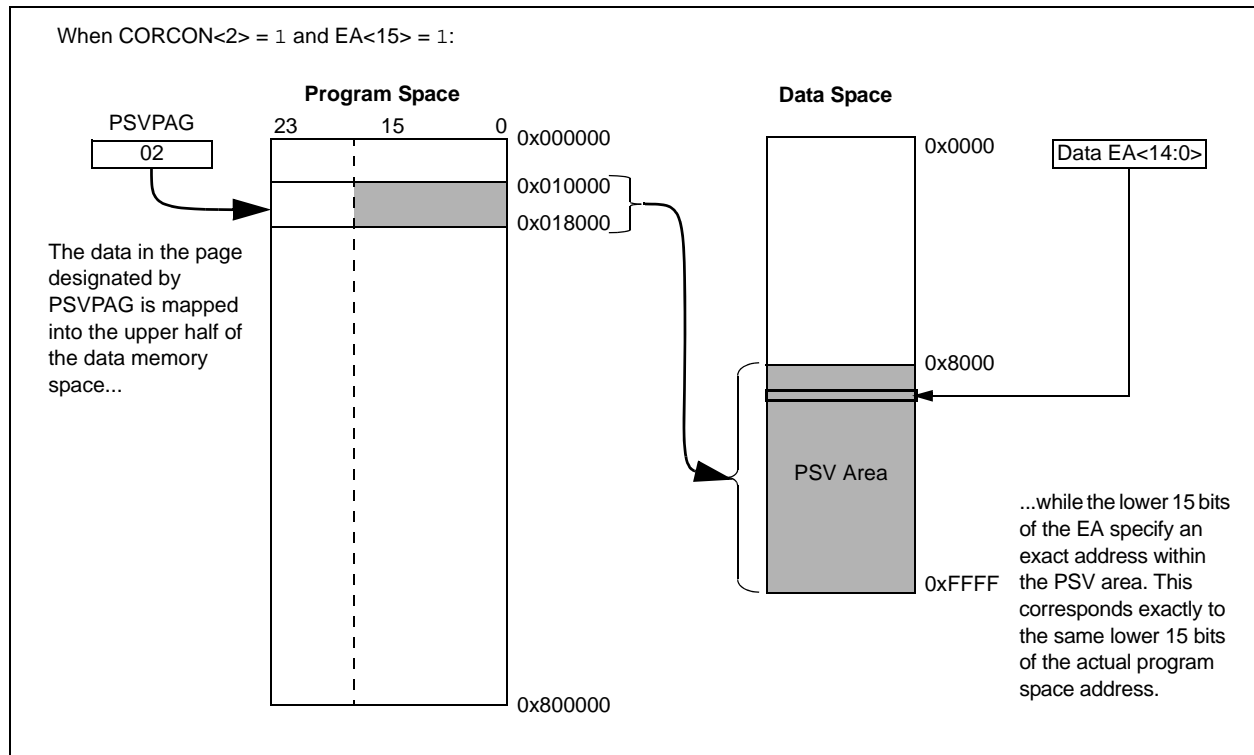
For operations that use PSV and are executed outside a REPEAT loop, the MOV and MOV.D instructions require one instruction cycle in addition to the specified execution time. All other instructions require two instruction cycles in addition to the specified execution time.

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 4-7: PROGRAM SPACE VISIBILITY OPERATION**



# PIC24FJ16MC101/102

---

---

NOTES:

## 5.0 FLASH PROGRAM MEMORY

**Note 1:** This data sheet summarizes the features of the PIC24FJ16MC101/102 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”** (DS39715) in the *“PIC24F Family Reference Manual”*, which is available from the Microchip web site ([www.microchip.com](http://www.microchip.com)).

- 2: It is important to note that the specifications in **Section 26.0 “Electrical Characteristics”** of this data sheet, supercede any specifications that may be provided in PIC24F Family Reference Manual sections.
- 3: 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 PIC24FJ16MC101/102 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 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 (MCLR). This allows users to manufacture boards with unprogrammed devices, and then program the microcontroller just before shipping the product. This also allows the most recent firmware or a custom firmware to be programmed.

RTSP is accomplished using TBLRD (table read) and TBLWT (table write) instructions. With RTSP, the user application can write program memory data in a single program memory word, and erase program memory in blocks or ‘pages’ of 512 instructions (1536 bytes).

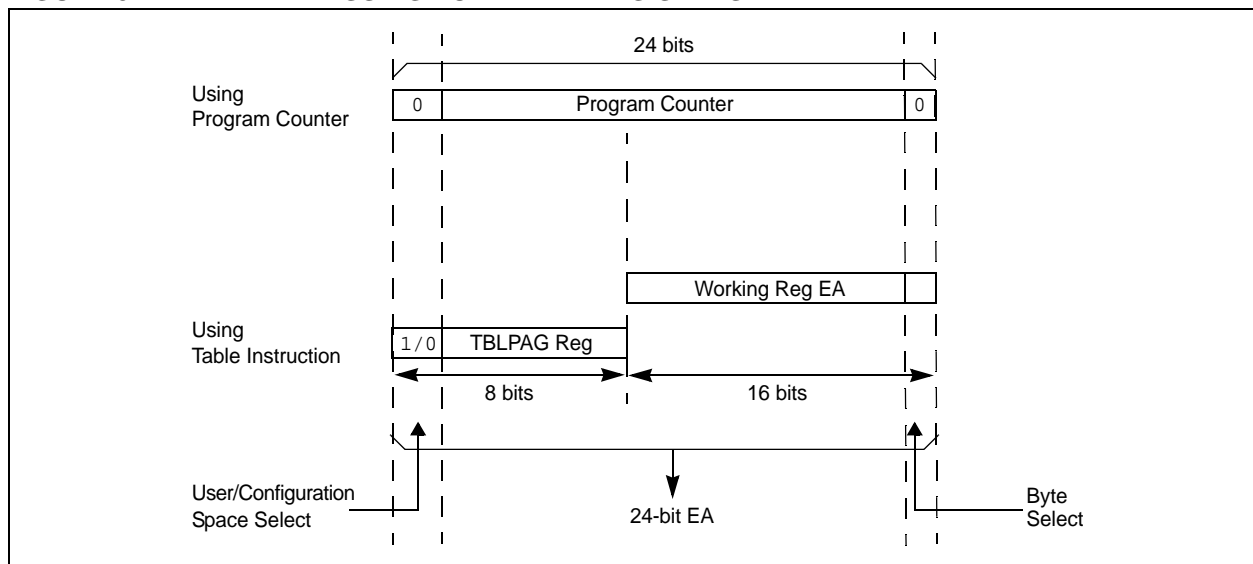
### 5.1 Table Instructions and Flash Programming

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 PIC24FJ16MC101/102 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); and to program one word. [Table 26-12](#) shows typical erase and programming times. The 8-row erase pages are edge-aligned from the beginning of program memory, on boundaries of 1536 bytes.

## 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 operation is finished.

For erase and program times, refer to parameters DI37a and DI37b (Page Erase Time), and DI38a and DI38b (Word Write Cycle Time), in [Table 26-12: “DC Characteristics: Program Memory”](#).

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

### 5.3.1 PROGRAMMING ALGORITHM FOR FLASH PROGRAM MEMORY

Programmers can program one word (24 bits) of program Flash memory at a time. To do this, it is necessary to erase the 8-row erase page that contains the desired address of the location the user wants to change.

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

<p><b>Note:</b> Performing a page erase operation on the last page of program memory will clear the Flash Configuration words, thereby enabling code protection as a result. Therefore, users should avoid performing page erase operations on the last page of program memory.</p>
---

Refer to [Section 4. “Program Memory” \(DS39715\)](#) in the *“PIC24F Family Reference Manual”* for details and codes examples on programming using RTSP.

## 5.4 Control Registers

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

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

NVMKEY is a write-only register that is used for write protection. To start a programming or erase sequence, the user application must consecutively write 0x55 and 0xAA to the NVMKEY register. Refer to [Section 5.3 “Programming Operations”](#) for further details.

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

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

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

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

- bit 15      **WR:** Write Control bit  
 1 = Initiates a Flash memory program or erase operation. The operation is self-timed and the bit is cleared by hardware once operation is complete  
 0 = Program or erase operation is complete and inactive
- bit 14      **WREN:** Write Enable bit  
 1 = Enable Flash program/erase operations  
 0 = Inhibit Flash program/erase operations
- bit 13      **WRERR:** Write Sequence Error Flag bit  
 1 = An improper program or erase sequence attempt or termination has occurred (bit is set automatically on any set attempt of the WR bit)  
 0 = The program or erase operation completed normally
- bit 12-7    **Unimplemented:** Read as '0'
- bit 6        **ERASE:** Erase/Program Enable bit  
 1 = Perform the erase operation specified by NVMOP<3:0> on the next WR command  
 0 = Perform the program operation specified by NVMOP<3:0> on the next WR command
- bit 5-4     **Unimplemented:** Read as '0'
- bit 3-0     **NVMOP<3:0>:** NVM Operation Select bits<sup>(2)</sup>  
If ERASE = 1:  
 1111 = No operation  
 1101 = Erase General Segment  
 1100 = No operation  
 0011 = No operation  
 0010 = Memory page erase operation  
 0001 = No operation  
 0000 = No operation  
  
If ERASE = 0:  
 1111 = No operation  
 1101 = No operation  
 1100 = No operation  
 0011 = Memory word program operation  
 0010 = No operation  
 0001 = No operation  
 0000 = No operation

- Note 1:** These bits can only be reset on POR.  
**Note 2:** All other combinations of NVMOP<3:0> are unimplemented.

# PIC24FJ16MC101/102

## 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

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

bit 15-8      **Unimplemented:** Read as '0'  
bit 7-0      **NVMKEY<7:0>:** Key Register (write-only) bits

## 6.0 RESETS

**Note 1:** This data sheet summarizes the features of the PIC24FJ16MC101/102 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. “Reset”** (DS39712) in the “PIC24F Family Reference Manual”, which is available from the Microchip web site ([www.microchip.com](http://www.microchip.com)).

**2:** It is important to note that the specifications in **Section 26.0 “Electrical Characteristics”** of this data sheet, supercede any specifications that may be provided in PIC24F Family Reference Manual sections.

**3:** 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 data sheet for register Reset states.

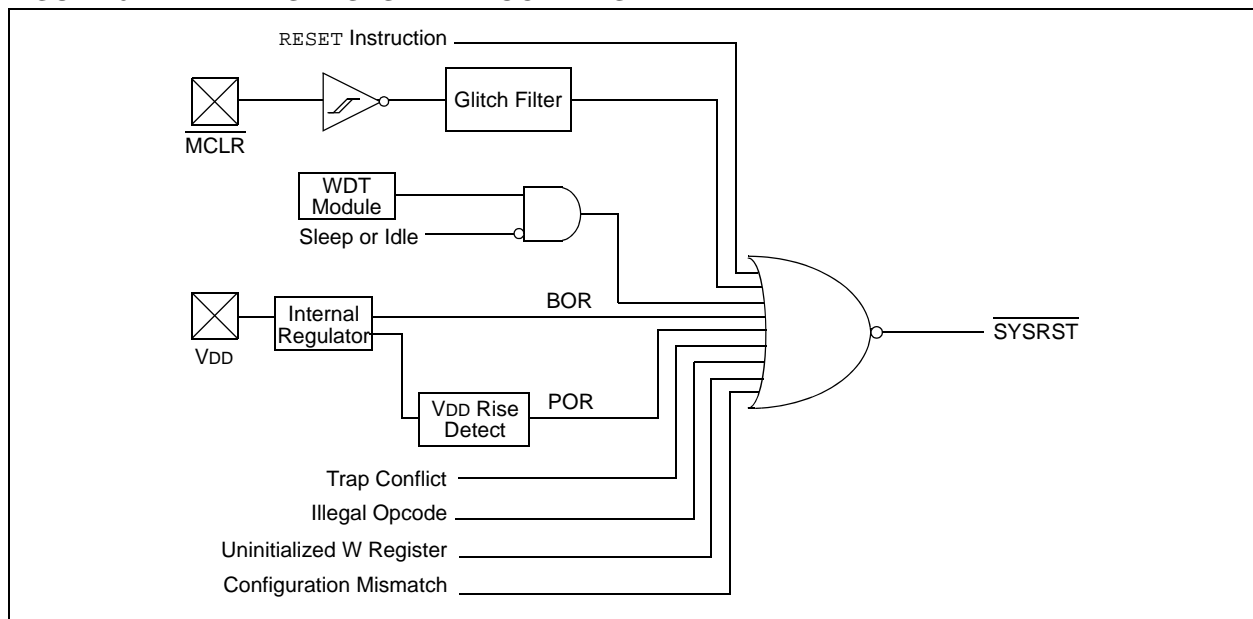
All types of device Reset set a corresponding status bit in the RCON register to indicate the type of Reset (see **Register 6-1**).

All bits that are set, with the exception of the POR bit (RCON<0>), are cleared during a POR event. 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 data sheet.

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

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



# PIC24FJ16MC101/102

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

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 <sup>(2)</sup>	WDTO	SLEEP	IDLE	BOR	POR
bit 7						bit 0	

### Legend:

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

- bit 15            **TRAPR:** Trap Reset Flag bit  
                   1 = A Trap Conflict Reset has occurred  
                   0 = A Trap Conflict Reset has not occurred
- bit 14            **IOPUWR:** Illegal Opcode or Uninitialized W Access Reset Flag bit  
                   1 = An illegal opcode detection, an illegal address mode or uninitialized W register used as an Address Pointer caused a Reset  
                   0 = An illegal opcode or uninitialized W Reset has not occurred
- bit 13-10       **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 Stand-by During Sleep bit  
                   1 = Voltage regulator is active during Sleep  
                   0 = Voltage regulator goes into Stand-by 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<sup>(2)</sup>  
                   1 = WDT is enabled  
                   0 = WDT is disabled
- bit 4             **WDTO:** Watchdog Timer Time-out Flag bit  
                   1 = WDT time-out has occurred  
                   0 = WDT time-out has not occurred
- bit 3             **SLEEP:** Wake-up from Sleep Flag bit  
                   1 = Device has been in Sleep mode  
                   0 = Device has not been in Sleep mode
- bit 2             **IDLE:** Wake-up from Idle Flag bit  
                   1 = Device was in Idle mode  
                   0 = Device was not in Idle mode
- bit 1             **BOR:** Brown-out Reset Flag bit  
                   1 = A Brown-out Reset has occurred  
                   0 = A Brown-out Reset has not occurred
- bit 0             **POR:** Power-on Reset Flag bit  
                   1 = A Power-on Reset has occurred  
                   0 = A Power-on Reset has not occurred

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

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



## 6.1 System Reset

The PIC24FJ16MC101/102 family of devices have two types of Reset:

- Cold Reset
- Warm Reset

A cold Reset is the result of a POR or a 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](#).

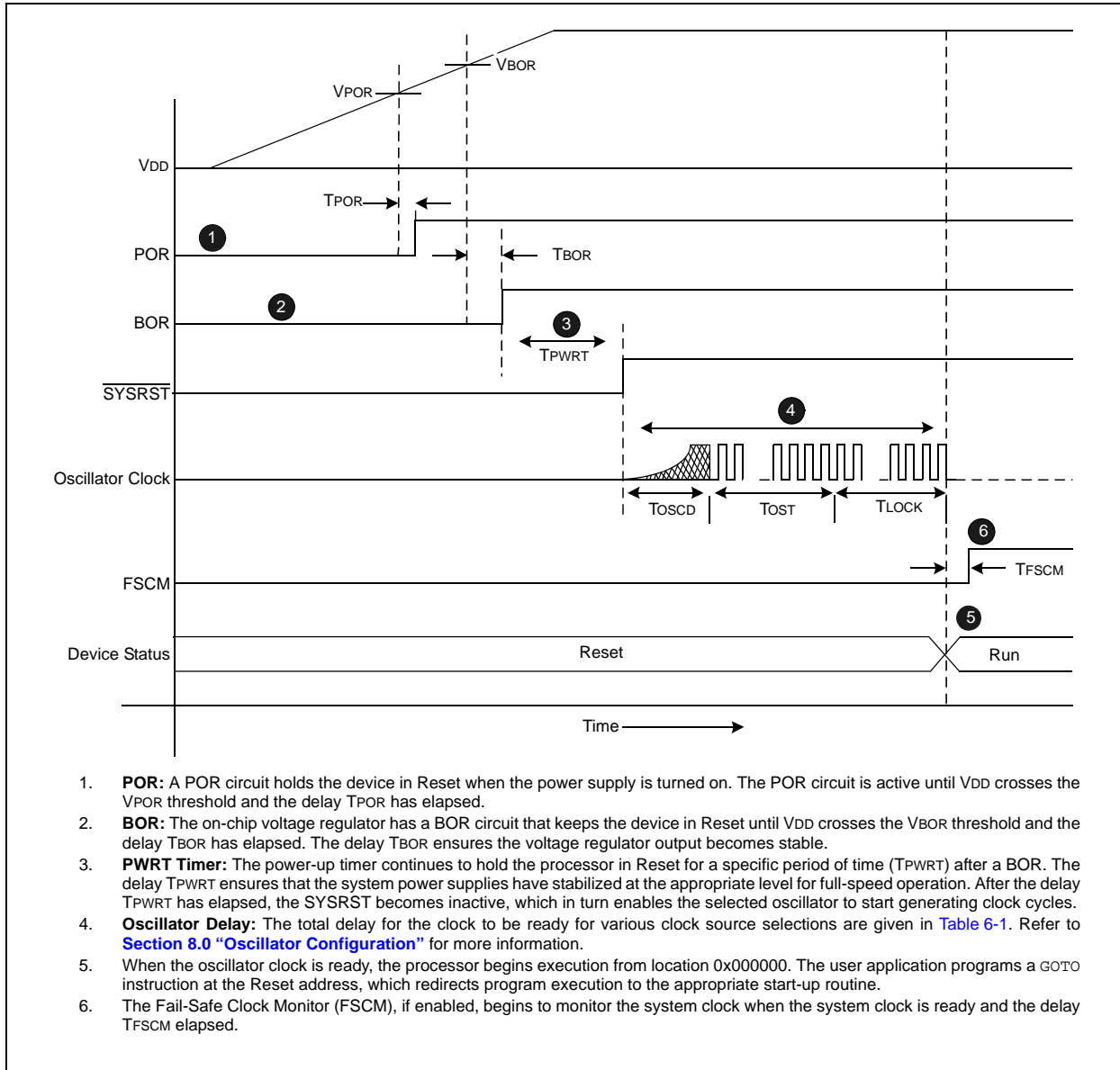
**TABLE 6-1: OSCILLATOR DELAY**

Oscillator Mode	Oscillator Startup Delay	Oscillator Startup Timer	PLL Lock Time	Total Delay
FRC, FRCDIV16, FRCDIVN	TOSCD	—	—	TOSCD
FRCPLL	TOSCD	—	TLOCK	TOSCD + TLOCK
MS	TOSCD	TOST	—	TOSCD + TOST
HS	TOSCD	TOST	—	TOSCD + TOST
EC	—	—	—	—
MSPLL	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  $\mu$ s max for FRC, 70  $\mu$ s 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  $\mu$ s 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.

# PIC24FJ16MC101/102

**FIGURE 6-2: SYSTEM RESET TIMING**



**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	Power-up time delay	64 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 POR

A 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 26.0 “Electrical Characteristics”](#) for details.

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

## 6.3 BOR and PWRT

The on-chip regulator has a BOR circuit that resets the device when the VDD is too low ( $V_{DD} < V_{BOR}$ ) for proper device operation. The BOR circuit keeps the device in Reset until VDD crosses the VBOR threshold and the delay TBOR has elapsed. The delay TBOR ensures the voltage regulator output becomes stable.

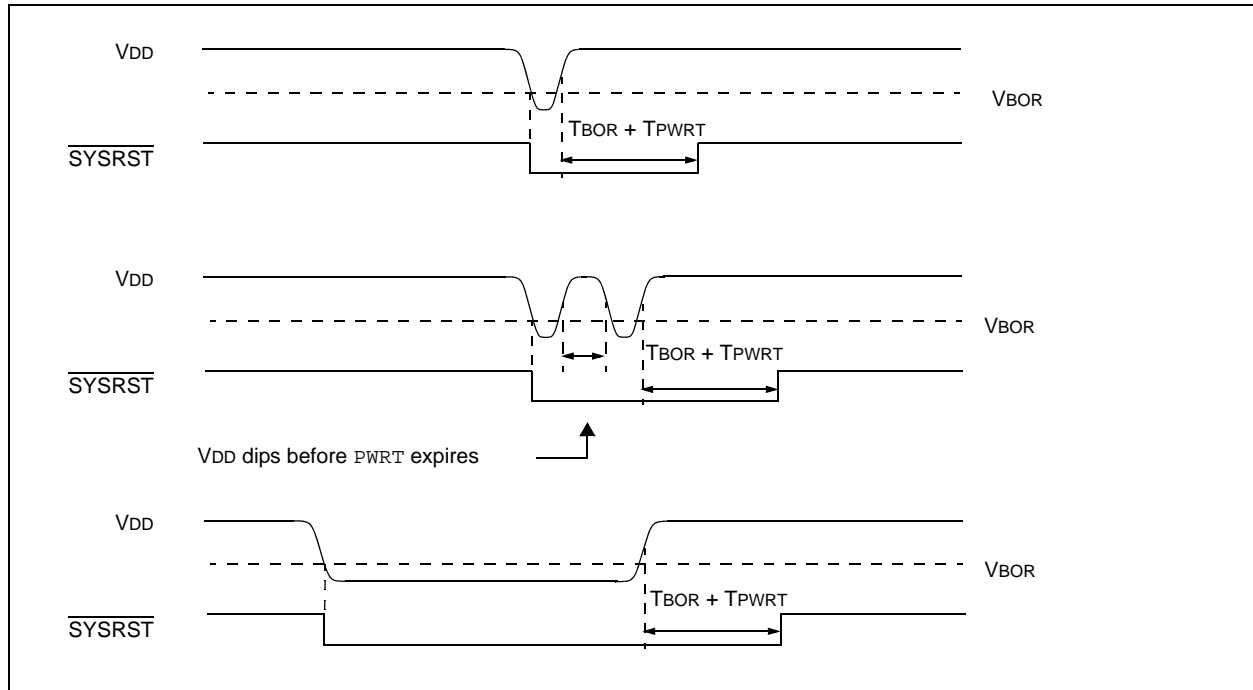
The BOR status bit (BOR) in the Reset Control register (RCON<1>) is set to indicate the Brown-out Reset.

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.

Refer to [Section 23.0 “Special Features”](#) for further details.

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

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



## 6.4 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 26.0 “Electrical Characteristics”](#) for minimum pulse width specifications. The External Reset (MCLR) Pin (EXTR) bit in the Reset Control register (RCON) is set to indicate the MCLR Reset.

### 6.4.1 EXTERNAL SUPERVISORY CIRCUIT

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

### 6.4.2 INTERNAL SUPERVISORY CIRCUIT

When using the internal power supervisory circuit to Reset the device, the external Reset pin (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.

## 6.5 Software RESET Instruction (SWR)

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

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

## 6.6 Watchdog Time-out Reset (WDTO)

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

The Watchdog Timer Time-out Flag bit (WDTO) in the Reset Control register (RCON<4>) is set to indicate the Watchdog Reset. Refer to [Section 23.4 “Watchdog Timer \(WDT\)”](#) for more information on Watchdog Reset.

## 6.7 Trap Conflict Reset

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

## 6.8 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.

<b>Note:</b> The configuration mismatch feature and associated Reset flag is not available on all devices.
--

## 6.9 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.

### 6.9.1 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 0x3F, which is an illegal opcode value.

### 6.9.2 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.

### 6.9.3 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.

## 6.10 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 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>)	MCLR Reset	POR
SWR (RCON<6>)	RESET instruction	POR, BOR
WDTO (RCON<4>)	WDT Time-out	PWRSV instruction, CLRWDT instruction, POR, BOR
SLEEP (RCON<3>)	PWRSV #SLEEP instruction	POR, BOR
IDLE (RCON<2>)	PWRSV #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.

# PIC24FJ16MC101/102

---

---

NOTES:

## 7.0 INTERRUPT CONTROLLER

**Note 1:** This data sheet summarizes the features of the PIC24FJ16MC101/102 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. “Interrupts”** (DS39707) in the “PIC24F Family Reference Manual”, which is available from the Microchip web site ([www.microchip.com](http://www.microchip.com)).

**2:** It is important to note that the specifications in **Section 26.0 “Electrical Characteristics”** of this data sheet, supercede any specifications that may be provided in PIC24F Family Reference Manual sections.

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

The Interrupt Controller reduces the numerous peripheral interrupt request signals to a single interrupt request signal to the PIC24FJ16MC101/102 CPU. It has the following features:

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

### 7.1 Interrupt Vector Table

The Interrupt Vector Table (IVT) is shown in [Figure 7-1](#). The IVT resides in program memory, starting at location 000004h. The IVT contains 126 vectors consisting of eight non-maskable trap vectors, plus up to 118 sources of interrupt. In general, each interrupt source has its own vector. Each interrupt vector contains a 24-bit-wide address. The value programmed into each interrupt vector location is the starting address of the associated Interrupt Service Routine (ISR).

Interrupt vectors are prioritized in terms of their natural priority. This 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.

PIC24FJ16MC101/102 devices implement up to 26 unique interrupts and 4 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 way to switch between an application and a support environment without requiring the interrupt vectors to be reprogrammed. This feature also enables switching between applications to facilitate 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.

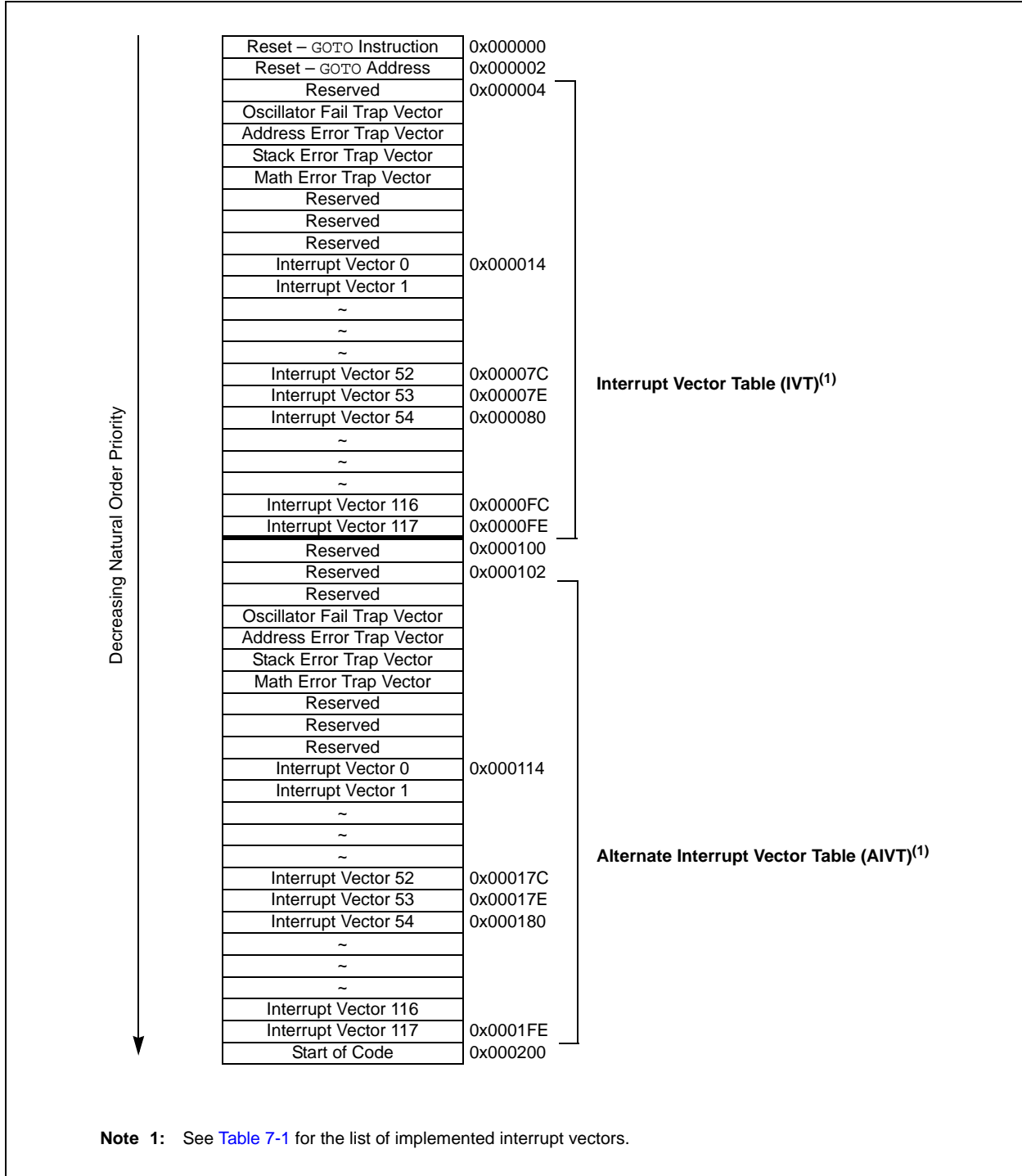
## 7.2 Reset Sequence

A device Reset is not a true exception because the interrupt controller is not involved in the Reset process. The PIC24FJ16MC101/102 device clears its registers in response to a Reset, forcing the PC to zero. The microcontroller then begins program execution at location 0x000000. A GOTO instruction at the Reset address can redirect program execution to the appropriate start-up routine.

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

# PIC24FJ16MC101/102

**FIGURE 7-1: PIC24FJ16MC101/102 INTERRUPT VECTOR TABLE**





**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	CMP – Comparator Interrupt
27	19	0x00003A	0x00013A	Change Notification Interrupt
28	20	0x00003C	0x00013C	INT1 – External Interrupt 1
29-36	21-28	0x00003E-0x00004C	0x00013E-0x00014C	Reserved
37	29	0x00004E	0x00014E	INT2 – External Interrupt 2
38-44	30-36	0x000050-0x00005A	0x000150-0x00015C	Reserved
45	37	0x00005E	0x00015E	IC3 – Input Capture 3
46-64	38-56	0x000060-0x000084	0x000160-0x000184	Reserved
65	57	0x000086	0x000186	PWM1 – PWM1 Period Match
66-69	58-61	0x000088-0x00008E	0x000188-0x00018E	Reserved
70	62	0x000090	0x000190	RTCC – Real-Time Clock and Calendar
71	63	0x000092	0x000192	FLTA1 – PWM1 Fault A
72	64	0x000094	0x000194	FLTB1 – PWM1 Fault B
73	65	0x000096	0x000196	U1E – UART1 Error
74-84	66-76	0x000098-0x0000AC	0x000198-0x0001AC	Reserved
85	77	0x0000AE	0x0001AE	CTMU – Charge Time Measurement Unit
86-125	78-117	0x0000B0-0x0000FE	0x0001B0-0x0001FE	Reserved

# PIC24FJ16MC101/102

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

The PIC24FJ16MC101/102 devices implement a total of 22 registers for the interrupt controller:

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

### 7.3.1 INTCON1 AND INTCON2

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

### 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 vector number (VECNUM<6:0>) and interrupt level (ILR<3:0>) bit fields in the INTTREG register. The new interrupt priority level is the priority of the pending interrupt.

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

### 7.3.6 STATUS/CONTROL REGISTERS

Although they are not specifically part of the interrupt control hardware, two of the CPU Control registers contain bits that control interrupt functionality.

- The CPU STATUS register, SR, contains the IPL<2:0> bits (SR<7:5>). These bits indicate the current CPU interrupt priority level. The user application can change the current CPU priority level by writing to the IPL bits.
- The CORCON register contains the IPL3 bit which, together with IPL<2:0>, also indicates the current CPU priority level. IPL3 is a read-only bit so that trap events cannot be masked by the user software.

All Interrupt registers are described in [Register 7-1](#) through [Register 7-27](#) in the following pages.

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

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

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

### Legend:

C = 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<sup>(2)</sup>

- 111 = CPU Interrupt Priority Level is 7 (15), user interrupts disabled
- 110 = CPU Interrupt Priority Level is 6 (14)
- 101 = CPU Interrupt Priority Level is 5 (13)
- 100 = CPU Interrupt Priority Level is 4 (12)
- 011 = CPU Interrupt Priority Level is 3 (11)
- 010 = CPU Interrupt Priority Level is 2 (10)
- 001 = CPU Interrupt Priority Level is 1 (9)
- 000 = CPU Interrupt Priority Level is 0 (8)

**Note 1:** For complete register details, see [Register 3-1: "SR: CPU Status Register"](#).

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

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

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

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

U-0	U-0	U-0	U-0	R/C-0	R/W-0	U-0	U-0
—	—	—	—	IPL3 <sup>(2)</sup>	PSV	—	—
bit 7							bit 0

### Legend:

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

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

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

**Note 1:** For complete register details, see [Register 3-2: "CORCON: Core Control Register"](#).

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

# PIC24FJ16MC101/102

## REGISTER 7-3: INTCON1: INTERRUPT CONTROL REGISTER 1

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

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

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

- bit 15      **NSTDIS:** Interrupt Nesting Disable bit  
                  1 = Interrupt nesting is disabled  
                  0 = Interrupt nesting is enabled
- bit 14-5    **Unimplemented:** Read as '0'
- bit 4        **MATHERR:** Arithmetic Error 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
- 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

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

- bit 15      **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

# PIC24FJ16MC101/102

## 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

# PIC24FJ16MC101/102

## 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

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	INT1IF	CNIF	CMPIF	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 12-5     **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        **CMPIF:** Comparator Interrupt Flag Status bit  
1 = Interrupt request has occurred  
0 = Interrupt request has not occurred
- bit 1        **MI2C1IF:** I2C1 Master Events Interrupt Flag Status bit  
1 = Interrupt request has occurred  
0 = Interrupt request has not occurred
- bit 0        **SI2C1IF:** I2C1 Slave Events Interrupt Flag Status bit  
1 = Interrupt request has occurred  
0 = Interrupt request has not occurred



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

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

U-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
—	—	IC3IF	—	—	—	—	—
bit 7							bit 0

**Legend:**

R = Readable bit                      W = Writable bit                      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              **IC3IF:** Input Capture Channel 3 Interrupt Flag Status bit  
                          1 = Interrupt request has occurred  
                          0 = Interrupt request has not occurred
- bit 4-0            **Unimplemented:** Read as '0'

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

R/W-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0	U-0
FLTA1IF	RTCCIF	—	—	—	—	PWM1IF	—
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            **FLTA1IF:** PWM1 Fault A Interrupt Flag Status bit  
                          1 = Interrupt request has occurred  
                          0 = Interrupt request has not occurred
- bit 14            **RTCCIF:** RTCC Interrupt Flag Status bit  
                          1 = Interrupt request has occurred  
                          0 = Interrupt request has not occurred
- bit 13-10       **Unimplemented:** Read as '0'
- bit 9             **PWM1IF:** PWM1 Interrupt Flag Status bit  
                          1 = Interrupt request has occurred  
                          0 = Interrupt request has not occurred
- bit 8-0           **Unimplemented:** Read as '0'

# PIC24FJ16MC101/102

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

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

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	U1EIF	FLTB1IF
bit 7							bit 0

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

- bit 15-14     **Unimplemented:** Read as '0'
- bit 13       **CTMUIF:** CTMU Interrupt Flag Status bit  
              1 = Interrupt request has occurred  
              0 = Interrupt request has not occurred
- bit 12-2     **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        **FLTB1IF:** PWM1 Fault B Interrupt Flag Status bit  
              1 = Interrupt request has occurred  
              0 = Interrupt request has not occurred

## REGISTER 7-10: 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 Event Interrupt Enable bit  
             1 = Interrupt request enabled  
             0 = Interrupt request not enabled
- bit 8      **T3IE:** Timer3 Interrupt Enable bit  
             1 = Interrupt request enabled  
             0 = Interrupt request not enabled
- bit 7      **T2IE:** Timer2 Interrupt Enable bit  
             1 = Interrupt request enabled  
             0 = Interrupt request not enabled
- bit 6      **OC2IE:** Output Compare Channel 2 Interrupt Enable bit  
             1 = Interrupt request enabled  
             0 = Interrupt request not enabled
- bit 5      **IC2IE:** Input Capture Channel 2 Interrupt Enable bit  
             1 = Interrupt request enabled  
             0 = Interrupt request not enabled
- bit 4      **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

# PIC24FJ16MC101/102

---

---

## REGISTER 7-10: 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 7-11: 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	

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	INT1IE	CNIE	CMPIE	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 12-5     **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         **CMPIE:** Comparator Interrupt Enable bit  
                   1 = Interrupt request enabled  
                   0 = Interrupt request not enabled
- bit 1         **MI2C1IE:** I2C1 Master Events Interrupt Enable bit  
                   1 = Interrupt request enabled  
                   0 = Interrupt request not enabled
- bit 0         **SI2C1IE:** I2C1 Slave Events Interrupt Enable bit  
                   1 = Interrupt request enabled  
                   0 = Interrupt request not enabled

# PIC24FJ16MC101/102

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

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

U-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
—	—	IC3IE	—	—	—	—	—
bit 7							bit 0

### Legend:

R = Readable bit                      W = Writable bit                      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              **IC3IE:** Input Capture Channel 3 Interrupt Enable bit  
                     1 = Interrupt request enabled  
                     0 = Interrupt request not enabled
- bit 4-0            **Unimplemented:** Read as '0'

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

R/W-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0	U-0
FLTA1IE	RTCCIE	—	—	—	—	PWM1IE	—
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            **FLTA1IE:** PWM1 Fault A Interrupt Enable bit  
                     1 = Interrupt request enabled  
                     0 = Interrupt request not enabled
- bit 14            **RTCCIE:** RTCC Interrupt Enable bit  
                     1 = Interrupt request enabled  
                     0 = Interrupt request not enabled
- bit 13-10        **Unimplemented:** Read as '0'
- bit 9             **PWM1IE:** PWM1 Interrupt Enable bit  
                     1 = Interrupt request enabled  
                     0 = Interrupt request not enabled
- bit 8-0           **Unimplemented:** Read as '0'

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

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

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	U1EIE	FLTB1IE
bit 7						bit 0	

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

- bit 15-14     **Unimplemented:** Read as '0'
- bit 13       **CTMUIE:** CTMU Interrupt Enable bit  
               1 = Interrupt request enabled  
               0 = Interrupt request not enabled
- bit 12-2     **Unimplemented:** Read as '0'
- bit 1         **U1EIE:** UART1 Error Interrupt Enable bit  
               1 = Interrupt request enabled  
               0 = Interrupt request not enabled
- bit 0         **FLTB1IE:** PWM1 Fault B Interrupt Enable bit  
               1 = Interrupt request has occurred  
               0 = Interrupt request has not occurred

# PIC24FJ16MC101/102

## REGISTER 7-15: IPC0: INTERRUPT PRIORITY CONTROL REGISTER 0

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	T1IP<2:0>			—	OC1IP<2:0>		
bit 15				bit 8			

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	IC1IP<2:0>			—	INT0IP<2:0>		
bit 7				bit 0			

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

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

bit 14-12 **T1IP<2:0>:** Timer1 Interrupt Priority bits  
 111 = Interrupt is priority 7 (highest priority interrupt)  
 •  
 •  
 •  
 001 = Interrupt is priority 1  
 000 = Interrupt source is disabled

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

bit 10-8 **OC1IP<2:0>:** Output Compare Channel 1 Interrupt Priority bits  
 111 = Interrupt is priority 7 (highest priority interrupt)  
 •  
 •  
 •  
 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 7-16: IPC1: INTERRUPT PRIORITY CONTROL REGISTER 1

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	T2IP<2:0>			—	OC2IP<2:0>		
bit 15				bit 8			

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	IC2IP<2:0>			—	—	—	—
bit 7				bit 0			

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

- bit 15      **Unimplemented:** Read as '0'
- bit 14-12    **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'

# PIC24FJ16MC101/102

## REGISTER 7-17: IPC2: INTERRUPT PRIORITY CONTROL REGISTER 2

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	U1RXIP<2:0>			—	SPI1IP<2:0>		
bit 15				bit 8			

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	SPI1EIP<2:0>			—	T3IP<2:0>		
bit 7				bit 0			

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

- bit 15      **Unimplemented:** Read as '0'
- bit 14-12    **U1RXIP<2:0>:** UART1 Receiver Interrupt Priority bits  
 111 = Interrupt is priority 7 (highest priority interrupt)  
 •  
 •  
 •  
 001 = Interrupt is priority 1  
 000 = Interrupt source is disabled
- bit 11      **Unimplemented:** Read as '0'
- bit 10-8    **SPI1IP<2:0>:** SPI1 Event Interrupt Priority bits  
 111 = Interrupt is priority 7 (highest priority interrupt)  
 •  
 •  
 •  
 001 = Interrupt is priority 1  
 000 = Interrupt source is disabled
- 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 7-18: 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

# PIC24FJ16MC101/102

## REGISTER 7-19: IPC4: INTERRUPT PRIORITY CONTROL REGISTER 4

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

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

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

- bit 15      **Unimplemented:** Read as '0'
- bit 14-12    **CNIP<2:0>:** Change Notification Interrupt Priority bits  
 111 = Interrupt is priority 7 (highest priority interrupt)  
 •  
 •  
 •  
 001 = Interrupt is priority 1  
 000 = Interrupt source is disabled
- bit 11      **Unimplemented:** Read as '0'
- bit 10-8    **CMPIP<2:0>:** Comparator Interrupt Priority bits  
 111 = Interrupt is priority 7 (highest priority interrupt)  
 •  
 •  
 •  
 001 = Interrupt is priority 1  
 000 = Interrupt source is disabled
- bit 7      **Unimplemented:** Read as '0'
- bit 6-4    **MI2C1IP<2:0>:** I2C1 Master Events Interrupt Priority bits  
 111 = Interrupt is priority 7 (highest priority interrupt)  
 •  
 •  
 •  
 001 = Interrupt is priority 1  
 000 = Interrupt source is disabled
- bit 3      **Unimplemented:** Read as '0'
- bit 2-0    **SI2C1IP<2:0>:** I2C1 Slave Events Interrupt Priority bits  
 111 = Interrupt is priority 7 (highest priority interrupt)  
 •  
 •  
 •  
 001 = Interrupt is priority 1  
 000 = Interrupt source is disabled

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

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

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

### Legend:

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

bit 15-3                      **Unimplemented:** Read as '0'  
 bit 2-0                      **INT1IP<2:0>:** External Interrupt 1 Priority bits  
                                     111 = Interrupt is priority 7 (highest priority interrupt)  
                                     •  
                                     •  
                                     •  
                                     001 = Interrupt is priority 1  
                                     000 = Interrupt source is disabled

## REGISTER 7-21: 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'

# PIC24FJ16MC101/102

## REGISTER 7-22: IPC9: INTERRUPT PRIORITY CONTROL REGISTER 9

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
—	IC3IP<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                      **IC3IP<2:0>:** External Interrupt 3 Priority bits  
     111 = Interrupt is priority 7 (highest priority interrupt)  
     •  
     •  
     •  
     001 = Interrupt is priority 1  
     000 = Interrupt source is disabled  
 bit 3-0                      **Unimplemented:** Read as '0'

## REGISTER 7-23: IPC14: INTERRUPT PRIORITY CONTROL REGISTER 14

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
—	PWM1IP<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                      **PWM1IP<2:0>:** PWM1 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 7-24: IPC15: INTERRUPT PRIORITY CONTROL REGISTER 15

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	FLTA1IP<2:0>			—	RTCCIP<2:0>		
bit 15				bit 8			

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

### Legend:

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

- bit 15                      **Unimplemented:** Read as '0'
- bit 14-12                **FLTA1IP<2:0>:** PWM1 Fault A Interrupt Priority bits
  - 111 = Interrupt is priority 7 (highest priority interrupt)
  - 
  - 
  - 
  - 001 = Interrupt is priority 1
  - 000 = Interrupt source is disabled
- bit 11                      **Unimplemented:** Read as '0'
- bit 10-8                **RTCCIP<2:0>:** RTCC Interrupt Priority bits
  - 111 = Interrupt is priority 7 (highest priority interrupt)
  - 
  - 
  - 
  - 001 = Interrupt is priority 1
  - 000 = Interrupt source is disabled
- bit 7-0                      **Unimplemented:** Read as '0'

# PIC24FJ16MC101/102

## REGISTER 7-25: 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	R/W-0	R/W-0	R/W-0
—	U1EIP<2:0>			—	FLT B1IP<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

**Unimplemented:** Read as '0'

bit 2-0

**FLT B1IP<2:0>:** PWM1 Fault B Interrupt Priority bits

111 = Interrupt is priority 7 (highest priority interrupt)

- 
- 
- 

001 = Interrupt is priority 1

000 = Interrupt source is disabled



## REGISTER 7-26: IPC19: INTERRUPT PRIORITY CONTROL REGISTER 19

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

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

### Legend:

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

bit 15-7                      **Unimplemented:** Read as '0'  
 bit 6-4                      **CTMUIP<2:0>:** CTMU Interrupt Priority bits  
                                     111 = Interrupt is priority 7 (highest priority interrupt)  
                                     •  
                                     •  
                                     •  
                                     001 = Interrupt is priority 1  
                                     000 = Interrupt source is disabled  
 bit 3-0                      **Unimplemented:** Read as '0'

# PIC24FJ16MC101/102

## REGISTER 7-27: 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 into the appropriate IPCx register. The priority level will depend on the specific application and type of interrupt source. If multiple priority levels are not desired, the IPCx register control bits for all enabled interrupt sources can be programmed to the same non-zero value.

**Note:** At a device Reset, the IPCx registers are initialized such that all user interrupt sources are assigned to priority level 4.

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

### 7.4.2 INTERRUPT SERVICE ROUTINE

The method used to declare an ISR and initialize the IVT with the correct vector address depends on the programming language (C or assembler) and the language development tool suite used to develop the application.

In general, the user application must clear the interrupt flag in the appropriate IFSx register for the source of interrupt that the ISR handles. Otherwise, program will re-enter the ISR immediately after exiting the routine. If the ISR is coded in assembly language, it must be terminated using a `RETFIE` instruction to unstack the saved PC value, SRL value and old CPU priority level.

### 7.4.3 TRAP SERVICE ROUTINE

A Trap Service Routine (TSR) is coded like an ISR, except that the appropriate trap status flag in the INTCON1 register must be cleared to avoid re-entry into the TSR.

### 7.4.4 INTERRUPT DISABLE

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.

# PIC24FJ16MC101/102

---

---

NOTES:

## 8.0 OSCILLATOR CONFIGURATION

**Note 1:** This data sheet summarizes the features of the PIC24FJ16MC101/102 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. “Oscillator”** (DS39700) in the “PIC24F Family Reference Manual”, which is available from the Microchip web site ([www.microchip.com](http://www.microchip.com)).

**2:** It is important to note that the specifications in **Section 26.0 “Electrical Characteristics”** of this data sheet, supercede any specifications that may be provided in PIC24F Family Reference Manual sections.

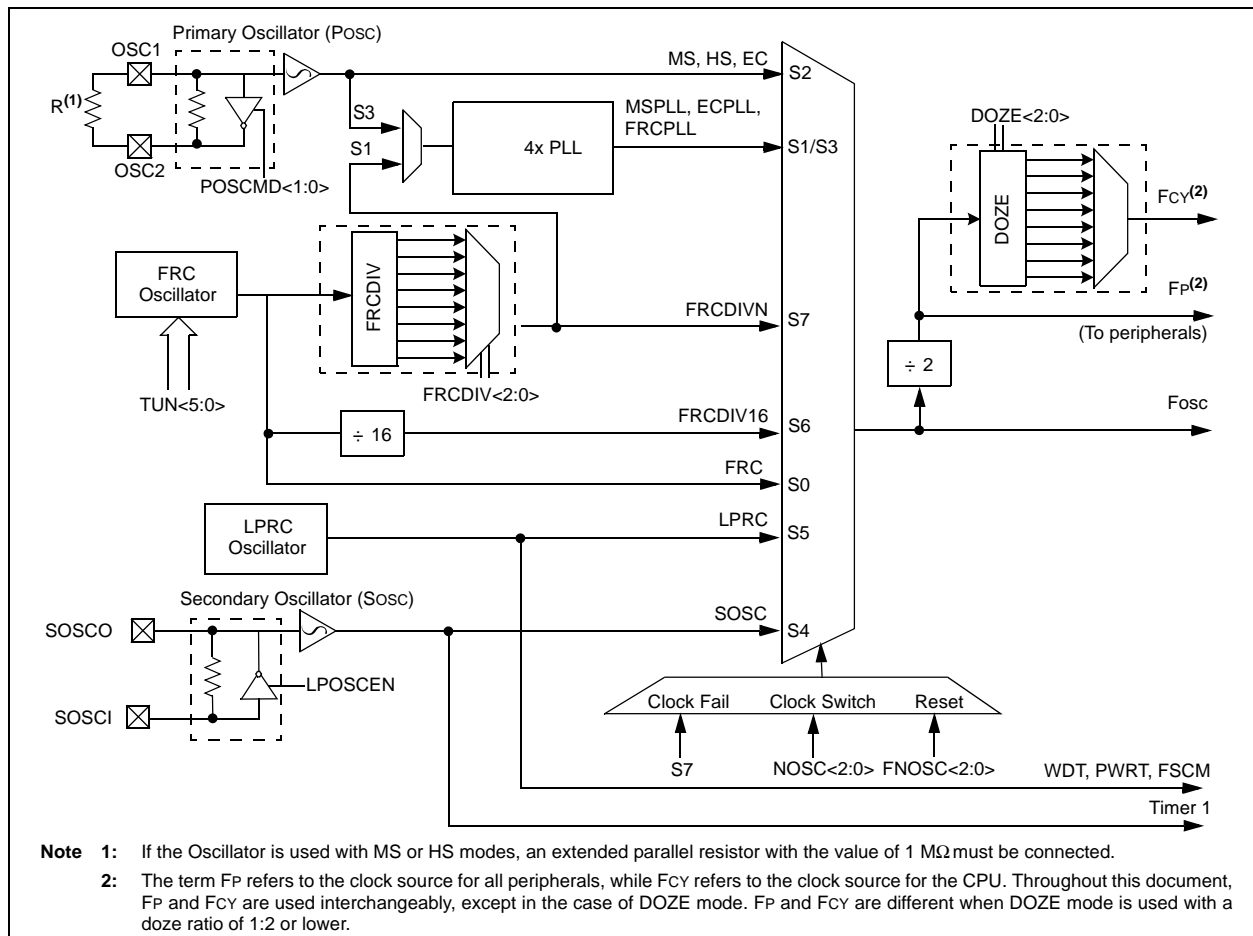
**3:** Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 “Memory Organization”** in this data sheet for device-specific register and bit information.

The PIC24FJ16MC101/102 oscillator system provides:

- External and internal oscillator options as clock sources
- An on-chip 4x Phase-Locked Loop (PLL) to scale the internal operating frequency to the required system clock frequency
- An internal FRC oscillator that can also be used with the PLL, thereby allowing full-speed operation without any external clock generation hardware
- Clock switching between various clock sources
- Programmable clock postscaler for system power savings
- A Fail-Safe Clock Monitor (FSCM) that detects clock failure and takes fail-safe measures
- A Clock Control register (OSCCON)
- Nonvolatile Configuration bits for main oscillator selection

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

**FIGURE 8-1: PIC24FJ16MC101/102 OSCILLATOR SYSTEM DIAGRAM**



## 8.1 CPU Clocking System

The PIC24FJ16MC101/102 devices provide seven system clock options:

- Fast RC (FRC) Oscillator
- FRC Oscillator with 4x PLL
- Primary (MS, HS or EC) Oscillator
- Primary Oscillator with 4x 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  $FRCDIV_{<2:0>}$  ( $CLKDIV_{<10:8>}$ ) bits.

The FRC frequency depends on the FRC accuracy (see [Table 26-18](#)) and the value of the FRC Oscillator Tuning register (see [Register 8-3](#)).

#### 8.1.1.2 Primary

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

- MS (Crystal): Crystals and ceramic resonators in the range of 4 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 32 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 SOSC1 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 PLL

The clock signals generated by the FRC and primary oscillators can be optionally applied to an on-chip 4x Phase-Locked Loop (PLL) to provide faster output frequencies for device operation. PLL configuration is described in [Section 8.1.3 “PLL Configuration”](#).

### 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 23.1 “Configuration Bits”](#) for further details.) The Initial Oscillator Selection Configuration bits,  $FNOSC_{<2:0>}$  ( $FOSCSEL_{<2:0>}$ ), and the Primary Oscillator Mode Select Configuration bits,  $POSCMD_{<1:0>}$  ( $FOSC_{<1:0>}$ ), select the oscillator source that is used at a Power-on Reset. The FRC primary oscillator is the default (unprogrammed) selection.

The Configuration bits allow users to choose 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 PIC24FJ16MC101/102 architecture.

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

#### EQUATION 8-1: DEVICE OPERATING FREQUENCY

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

## 8.1.3 PLL CONFIGURATION

The primary oscillator and internal FRC oscillator can optionally use an on-chip 4x PLL to obtain higher speeds of operation.

For example, suppose a 8 MHz crystal is being used with the selected oscillator mode of MS with PLL. This provides a Fosc of 8 MHz \* 4 = 32 MHz. The resultant device operating speed is 32/2 = 16 MIPS.

### EQUATION 8-2: MS WITH PLL MODE EXAMPLE

$$F_{CY} = \frac{F_{OSC}}{2} = \frac{1}{2} (8000000 \cdot 4) = 16 \text{ MIPS}$$

**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 (MS) with PLL (MSPLL)	Primary	01	011	—
Primary Oscillator (EC) with PLL (ECPLL)	Primary	00	011	1
Primary Oscillator (HS)	Primary	10	010	—
Primary Oscillator (MS)	Primary	01	010	—
Primary Oscillator (EC)	Primary	00	010	1
Fast RC Oscillator (FRC) with Divide-by-n and 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.

# PIC24FJ16MC101/102

## REGISTER 8-1: OSCCON: OSCILLATOR CONTROL REGISTER<sup>(1)</sup>

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

R/W-0	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	

<b>Legend:</b>	y = Value set from Configuration bits on POR	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 **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 (MS, EC) with PLL  
 010 = Primary Oscillator (MS, HS, EC)  
 001 = Fast RC Oscillator (FRC) with Divide-by-n and with PLL (FRCPLL)  
 000 = Fast RC Oscillator (FRC)

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

bit 10-8 **NOSC<2:0>:** New Oscillator Selection bits<sup>(2)</sup>  
 111 = Fast RC Oscillator (FRC) with Divide-by-n  
 110 = Fast RC Oscillator (FRC) with Divide-by-16  
 101 = Low-Power RC Oscillator (LPRC)  
 100 = Secondary Oscillator (SOSC)  
 011 = Primary Oscillator (MS, EC) with PLL  
 010 = Primary Oscillator (MS, HS, EC)  
 001 = Fast RC Oscillator (FRC) with Divide-by-n and PLL (FRCPLL)  
 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 = Peripheral pin select is locked, write to peripheral pin select registers not allowed  
 0 = Peripheral pin select is not locked, write to peripheral pin select registers 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'

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'

**Note 1:** Writes to this register require an unlock sequence. Refer to **Section 6. "Oscillator"** (DS39700) in the "PIC24F 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.



## REGISTER 8-1: OSCCON: OSCILLATOR CONTROL REGISTER<sup>(1)</sup> (CONTINUED)

- 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 6. “Oscillator”** (DS39700) in the *“PIC24F 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.

# PIC24FJ16MC101/102

## REGISTER 8-2: CLKDIV: CLOCK DIVISOR REGISTER

R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0
ROI	DOZE<2:0> <sup>(2,3)</sup>			DOZEN <sup>(1,2,3)</sup>	FRCDIV<2:0>		
bit 15				bit 8			

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

### Legend:

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

- bit 15                      **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<sup>(2,3)</sup>  
 111 = Fcy/128  
 110 = Fcy/64  
 101 = Fcy/32  
 100 = Fcy/16  
 011 = Fcy/8 (default)  
 010 = Fcy/4  
 001 = Fcy/2  
 000 = Fcy/1
- bit 11                      **DOZEN:** DOZE Mode Enable bit<sup>(1,2,3)</sup>  
 1 = DOZE<2:0> field specifies the ratio between the peripheral clocks and the processor clocks  
 0 = Processor clock/peripheral clock ratio forced to 1:1
- bit 10-8                      **FRCDIV<2:0>:** Internal Fast RC Oscillator Postscaler bits  
 111 = FRC divide by 256  
 110 = FRC divide by 64  
 101 = FRC divide by 32  
 100 = FRC divide by 16  
 011 = FRC divide by 8  
 010 = FRC divide by 4  
 001 = FRC divide by 2  
 000 = FRC divide by 1 (default)
- bit 7-0                      **Unimplemented:** Read as '0'

- Note 1:** This bit is cleared when the ROI bit is set and an interrupt occurs.  
**Note 2:** If DOZEN = 1, writes to DOZE<2:0> are ignored.  
**Note 3:** If DOZE<2:0> = 000, the DOZEN bit cannot be set by the user; writes are ignored.

## REGISTER 8-3: OSCTUN: FRC OSCILLATOR TUNING REGISTER

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

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	TUN<5:0> <sup>(1)</sup>					
bit 7							bit 0

### Legend:

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

bit 15-6      **Unimplemented:** Read as '0'

bit 5-0      **TUN<5:0>:** FRC Oscillator Tuning bits<sup>(1)</sup>

011111 = Center frequency +11.625% (8.23 MHz)

011110 = Center frequency +11.25% (8.20 MHz)

•

•

•

000001 = Center frequency +0.375% (7.40 MHz)

000000 = Center frequency (7.37 MHz nominal)

111111 = Center frequency -0.375% (7.345 MHz)

•

•

•

100001 = Center frequency -11.625% (6.52 MHz)

100000 = Center frequency -12% (6.49 MHz)

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

## 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, PIC24FJ16MC101/102 devices have a safeguard lock built into the switch process.

**Note:** Primary Oscillator mode has three different submodes (MS, 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 23.1 "Configuration Bits"](#) for further details.) If the FCKSM1 Configuration bit is unprogrammed ('1'), the clock switching function and Fail-Safe Clock Monitor function are disabled. This is the default setting.

The NOSC control bits (OSCCON<10:8>) do not control the clock selection when clock switching is disabled. However, the COSC bits (OSCCON<14:12>) reflect the clock source selected by the FNOSC Configuration bits.

The OSWEN control bit (OSCCON<0>) has no effect when clock switching is disabled. It is held at '0' at all times.

### 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 (OSCCON<0>) to initiate the oscillator switch.

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

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

2. If a valid clock switch has been initiated, the LOCK (OSCCON<5>) and the CF (OSCCON<3>) status bits are cleared.
3. The new oscillator is turned on by the hardware if it is not currently running. If a crystal oscillator must be turned on, the hardware waits until the Oscillator Start-up Timer (OST) expires. If the new source is using the PLL, the hardware waits until a PLL lock is detected (LOCK = 1).
4. The hardware waits for 10 clock cycles from the new clock source and then performs the clock switch.
5. The hardware clears the OSWEN bit to indicate a successful clock transition. In addition, the NOSC bit values are transferred to the COSC status bits.
6. The old clock source is turned off at this time, with the exception of LPRC (if WDT or FSCM are enabled) or LP (if LPOSCEN remains set).

**Note 1:** The processor continues to execute code throughout the clock switching sequence. Timing-sensitive code should not be executed during this time.

**2:** Direct clock switches between any primary oscillator mode with PLL and FRCPLL mode are not permitted. This applies to clock switches in either direction. In these instances, the application must switch to FRC mode as a transition clock source between the two PLL modes.

**3:** Refer to [Section 6. "Oscillator"](#) (DS39700) in the *"PIC24F 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 PIC24FJ16MC101/102 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 (WDT)”** (DS39697) and **Section 10. “Power-Saving Features”** (DS39698) in the *“PIC24F Family Reference Manual”*, which are available from the Microchip web site ([www.microchip.com](http://www.microchip.com)).

- 2: It is important to note that the specifications in **Section 26.0 “Electrical Characteristics”** of this data sheet, supercede any specifications that may be provided in PIC24F Family Reference Manual sections.
- 3: 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 PIC24FJ16MC101/102 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. PIC24FJ16MC101/102 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

PIC24FJ16MC101/102 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”**.

#### 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 Instruction-Based Power-Saving Modes

PIC24FJ16MC101/102 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 in Sleep mode if the WDT is enabled
- The WDT, if enabled, is automatically cleared prior to entering Sleep mode
- Some device features or peripherals may continue to operate. This includes 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.

## 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 device Reset
- A WDT time-out

On wake-up from Idle mode, the clock is reapplied to the CPU and instruction execution will begin (2-4 clock cycles later), starting with the instruction following the `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, this may not be practical. For example, it may be necessary for an application to maintain uninterrupted synchronous communication, even while it is doing nothing else. Reducing system clock speed can introduce communication errors, while using a power-saving mode can stop communications completely.

Doze mode is a simple and effective alternative method to reduce power consumption while the device is still executing code. In this mode, the system clock continues to operate from the same source and at the same speed. Peripheral modules continue to be clocked at the same speed, while the CPU clock speed is reduced. Synchronization between the two clock domains is maintained, allowing the peripherals to access the SFRs while the CPU executes code at a slower rate.

Doze mode is enabled by setting the DOZEN bit (`CLKDIV<11>`). The ratio between peripheral and core clock speed is determined by the `DOZE<2:0>` bits (`CLKDIV<14:12>`). There are eight possible configurations, from 1:1 to 1:128, with 1:1 being the default setting.

Programs can use Doze mode to selectively reduce power consumption in event-driven applications. This allows clock-sensitive functions, such as synchronous communications, to continue without interruption while the CPU idles, waiting for something to invoke an interrupt routine. An automatic return to full-speed CPU operation on interrupts can be enabled by setting the ROI bit (`CLKDIV<15>`). By default, interrupt events have no effect on Doze mode operation.

For example, suppose the device is operating at 20 MIPS and the UART 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 UART 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 PIC24FXXX 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	R/W-0	U-0
—	—	T3MD	T2MD	T1MD	—	PWM1MD	—
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 <sup>(1)</sup>
bit 7						bit 0	

### Legend:

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

- bit 15-14 **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 **Unimplemented:** Read as '0'
- bit 9 **PWM1MD:** PWM1 Module Disable bit  
1 = PWM1 module is disabled  
0 = PWM1 module is enabled
- bit 18 **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<sup>(1)</sup>  
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. When the bit is set, all port pins that have been multiplexed with ANx will be in Digital mode.

# PIC24FJ16MC101/102

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

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	IC3MD	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 15-11     **Unimplemented:** Read as '0'
- bit 10       **IC3MD:** Input Capture 3 Module Disable bit  
               1 = Input Capture 3 module is disabled  
               0 = Input Capture 3 module is enabled
- bit 9         **IC2MD:** Input Capture 2 Module Disable bit  
               1 = Input Capture 2 module is disabled  
               0 = Input Capture 2 module is enabled
- bit 8         **IC1MD:** Input Capture 1 Module Disable bit  
               1 = Input Capture 1 module is disabled  
               0 = Input Capture 1 module is enabled
- bit 7-2      **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



## REGISTER 9-3: PMD3: PERIPHERAL MODULE DISABLE CONTROL REGISTER 3

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	U-0
—	—	—	—	—	CMPMD	RTCCMD	—
bit 15						bit 8	

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

### Legend:

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

- bit 15-11     **Unimplemented:** Read as '0'
- bit 10        **CMPMD:** Comparator Module Disable bit  
                  1 = Comparator module is disabled  
                  0 = Comparator module is enabled
- bit 9         **RTCCMD:** RTCC Module Disable bit  
                  1 = RTCC module is disabled  
                  0 = RTCC module is enabled
- bit 8-0       **Unimplemented:** Read as '0'

## REGISTER 9-4: PMD4: PERIPHERAL MODULE DISABLE CONTROL REGISTER 4

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

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

### Legend:

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

- bit 15-4     **Unimplemented:** Read as '0'
- bit 3         **CTMUMD:** CTMU Module Disable bit  
                  1 = CTMU module is disabled  
                  0 = CTMU module is enabled
- bit 2-0       **Unimplemented:** Read as '0'

# PIC24FJ16MC101/102

---

---

NOTES:

## 10.0 I/O PORTS

**Note 1:** This data sheet summarizes the features of the PIC24FJ16MC101/102 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. “I/O Ports with Peripheral Pin Select (PPS)”** (DS39711) in the *“PIC24F Family Reference Manual”*, which is available from the Microchip web site ([www.microchip.com](http://www.microchip.com)).

- 2: It is important to note that the specifications in **Section 26.0 “Electrical Characteristics”** of this data sheet, supercede any specifications that may be provided in PIC24F Family Reference Manual sections.
- 3: Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 “Memory Organization”** in this data sheet for device-specific register and bit information.

All of the device pins (except VDD, VSS, MCLR, and OSC1/CLKI) are shared among the peripherals and the parallel I/O ports. All I/O input ports feature Schmitt Trigger inputs for improved noise immunity.

### 10.1 Parallel I/O (PIO) Ports

Generally a parallel I/O port that shares a pin with a peripheral is subservient to the peripheral. The peripheral’s output buffer data and control signals are

provided to a pair of multiplexers. The multiplexers select whether the peripheral or the associated port has ownership of the output data and control signals of the I/O pin. The logic also prevents “loop through,” in which a port’s digital output can drive the input of a peripheral that shares the same pin. **Figure 10-1** shows how ports are shared with other peripherals and the associated I/O pin to which they are connected.

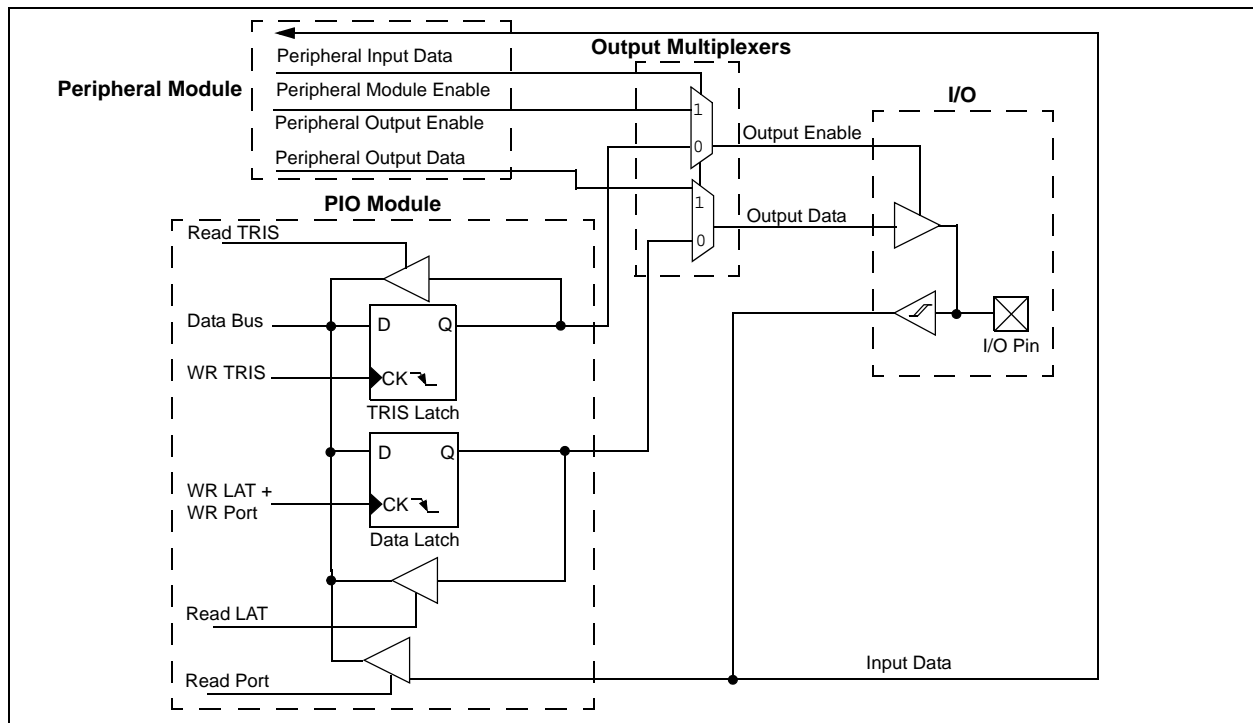
When a peripheral is enabled and the peripheral is actively driving an associated pin, the use of the pin as a general purpose output pin is disabled. The I/O pin can be read, but the output driver for the parallel port bit is disabled. If a peripheral is enabled, but the peripheral is not actively driving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



# PIC24FJ16MC101/102

## 10.1.1 OPEN-DRAIN CONFIGURATION

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

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

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

## 10.2 Configuring Analog Port Pins

The AD1PCFG and TRIS registers control the operation of the analog-to-digital port pins. The port pins that are to function as analog inputs must have their corresponding TRIS bit set (input). If the TRIS bit is cleared (output), the digital output level (VOH or VOL) 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 defined as a digital input (including the ANx pins) can cause the input buffer to consume current that exceeds the device specifications.

### 10.2.1 I/O PORT WRITE/READ TIMING

One instruction cycle is required between a port direction change or port write operation and a read operation of the same port. Typically this instruction would be an NOP. An demonstration is shown in [Example 10-1](#).

#### EXAMPLE 10-1: PORT WRITE/READ EXAMPLE

```
MOV    0xFF00, W0           ; Configure PORTB<15:8> as inputs
MOV    W0, TRISBB          ; and PORTB<7:0> as outputs
NOP                               ; Delay 1 cycle
btss  PORTB, #13           ; Next Instruction
```

## 10.3 Input Change Notification

The input change notification function of the I/O ports allows the PIC24FJ16MC101/102 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 21 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.

## 10.4 Peripheral Pin Select

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.

### 10.4.1 AVAILABLE PINS

The peripheral pin select feature is used with a range of up to 16 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.

### 10.4.2 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.

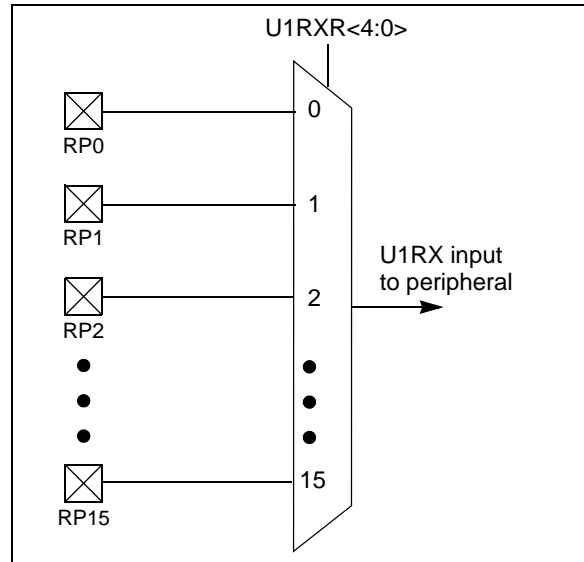
### 10.4.2.1 Input Mapping

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 10-8](#)). 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 RPx 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**



# PIC24FJ16MC101/102

**TABLE 10-1: SELECTABLE INPUT SOURCES (MAPS INPUT TO FUNCTION)<sup>(1)</sup>**

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 3	IC3	RPINR8	IC3R<4:0>
Output Compare Fault A	OCFA	RPINR11	OCFAR<4:0>
UART1 Receive	U1RX	RPINR18	U1RXR<4:0>
UART1 Clear To Send	U1CTS	RPINR18	U1CTSR<4:0>
SPI Slave Select Input	SS1	RPINR21	SS1R<4:0>

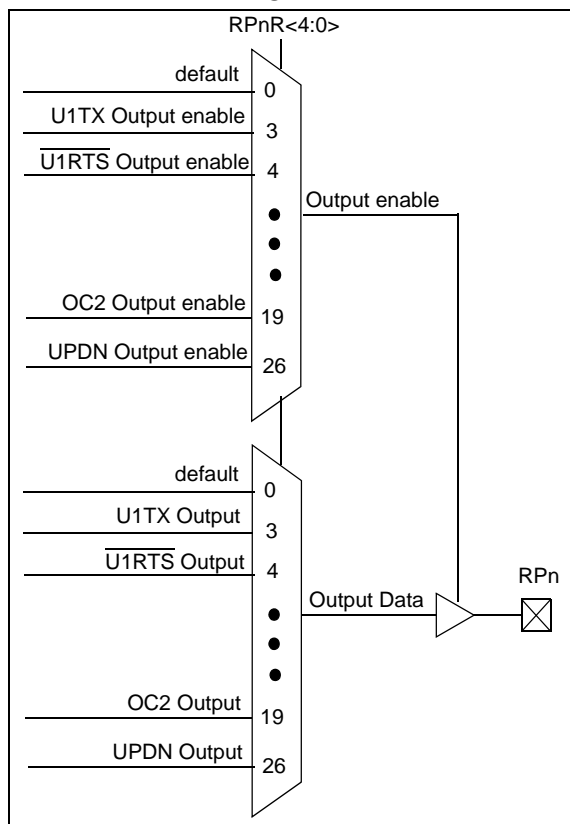
**Note 1:** Unless otherwise noted, all inputs use the Schmitt input buffers.

## 10.4.2.2 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 10-9](#) through [Register 10-16](#)). 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
C1OUT	00001	RPn tied to Comparator 1 Output
C2OUT	00010	RPn tied to Comparator 2 Output
U1TX	00011	RPn tied to UART1 Transmit
U1RTS	00100	RPn tied to UART1 Ready To Send
SS1	01001	RPn tied to SPI1 Slave Select Output
OC1	10010	RPn tied to Output Compare 1
OC2	10011	RPn tied to Output Compare 2
CTPLS	11101	RPn tied to CTMU Pulse Output
C3OUT	11110	RPn tied to Comparator 3 Output

### 10.4.3 CONTROLLING CONFIGURATION CHANGES

Because peripheral remapping can be changed during run time, some restrictions on peripheral remapping are needed to prevent accidental configuration changes. PIC24FJ16MC101/102 devices include three features to prevent alterations to the peripheral map:

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

#### 10.4.3.1 Control Register Lock Sequence

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.

#### 10.4.3.2 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.

#### 10.4.3.3 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 (FOSC<IOL1WAY>) configuration bit blocks the IOLOCK bit from being cleared after it has been set once. If IOLOCK remains set, the register unlock procedure will not execute, and the peripheral pin select control registers cannot be written to. The only way to clear the bit and re-enable peripheral remapping is to perform a device Reset.

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

## 10.5 Peripheral Pin Select Registers

The PIC24FJ16MC101/102 family of devices implement 21 registers for remappable peripheral configuration:

- Input Remappable Peripheral Registers (13)
- Output Remappable Peripheral Registers (8)

**Note:** Input and Output Register values can only be changed if OSCCON<IOLOCK> = 0. See [Section 10.4.3.1 “Control Register Lock Sequence”](#) for a specific command sequence.

# PIC24FJ16MC101/102

## 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 VSS
  - 01111 = Input tied to RP15
  - .
  - .
  - .
  - 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 VSS
  - 01111 = Input tied to RP15
  - .
  - .
  - .
  - 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 Vss  
01111 = Input tied to RP15

.  
.  
.

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 Vss  
01111 = Input tied to RP15

.  
.  
.

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

# PIC24FJ16MC101/102

## 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 VSS

01111 = Input tied to RP15

.

.

.

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 VSS

01111 = Input tied to RP15

.

.

.

00001 = Input tied to RP1

00000 = Input tied to RP0

## REGISTER 10-5: RPINR8: PERIPHERAL PIN SELECT INPUT REGISTER 8

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
—	—	—	IC3R<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                      **IC3R<4:0>:** Assign Input Capture 3 (IC3) to the corresponding pin RPn pin  
                                     11111 = Input tied Vss  
                                     01111 = Input tied to RP15  
                                     .  
                                     .  
                                     .  
                                     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 Vss  
                                     01111 = Input tied to RP15  
                                     .  
                                     .  
                                     .  
                                     00001 = Input tied to RP1  
                                     00000 = Input tied to RP0

# PIC24FJ16MC101/102

## 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 UART1 Clear to Send (U1CTS) to the corresponding RPn pin

11111 = Input tied Vss

01111 = Input tied to RP15

.

.

.

00001 = Input tied to RP1

00000 = Input tied to RP0

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

bit 4-0 **U1RXR<4:0>:** Assign UART1 Receive (U1RX) to the corresponding RPn pin

11111 = Input tied Vss

01111 = Input tied to RP15

.

.

.

00001 = Input tied to RP1

00000 = Input tied to RP0

## REGISTER 10-8: 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 VSS  
                                     01111 = Input tied to RP15  
                                     .  
                                     .  
                                     .  
                                     00001 = Input tied to RP1  
                                     00000 = Input tied to RP0

# PIC24FJ16MC101/102

## REGISTER 10-9: 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 bits (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 bits (see [Table 10-2](#) for peripheral function numbers)

## REGISTER 10-10: 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 bits (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 bits (see [Table 10-2](#) for peripheral function numbers)

## REGISTER 10-11: 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 bits (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 bits (see [Table 10-2](#) for peripheral function numbers)

## REGISTER 10-12: 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 bits (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 bits (see [Table 10-2](#) for peripheral function numbers)

# PIC24FJ16MC101/102

## REGISTER 10-13: 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 bits (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 bits (see [Table 10-2](#) for peripheral function numbers)

## REGISTER 10-14: 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 bits (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 bits (see [Table 10-2](#) for peripheral function numbers)



## REGISTER 10-15: 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 bits (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 bits (see [Table 10-2](#) for peripheral function numbers)

## REGISTER 10-16: 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 bits (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 bits (see [Table 10-2](#) for peripheral function numbers)

# PIC24FJ16MC101/102

---

---

NOTES:

## 11.0 TIMER1

**Note 1:** This data sheet summarizes the features of the PIC24FJ16MC101/102 family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 14. “Timers”** (DS39704) in the *“PIC24F Family Reference Manual”*, which is available from the Microchip web site ([www.microchip.com](http://www.microchip.com)).

- 2: It is important to note that the specifications in **Section 26.0 “Electrical Characteristics”** of this data sheet, supercede any specifications that may be provided in PIC24F Family Reference Manual sections.
- 3: 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:

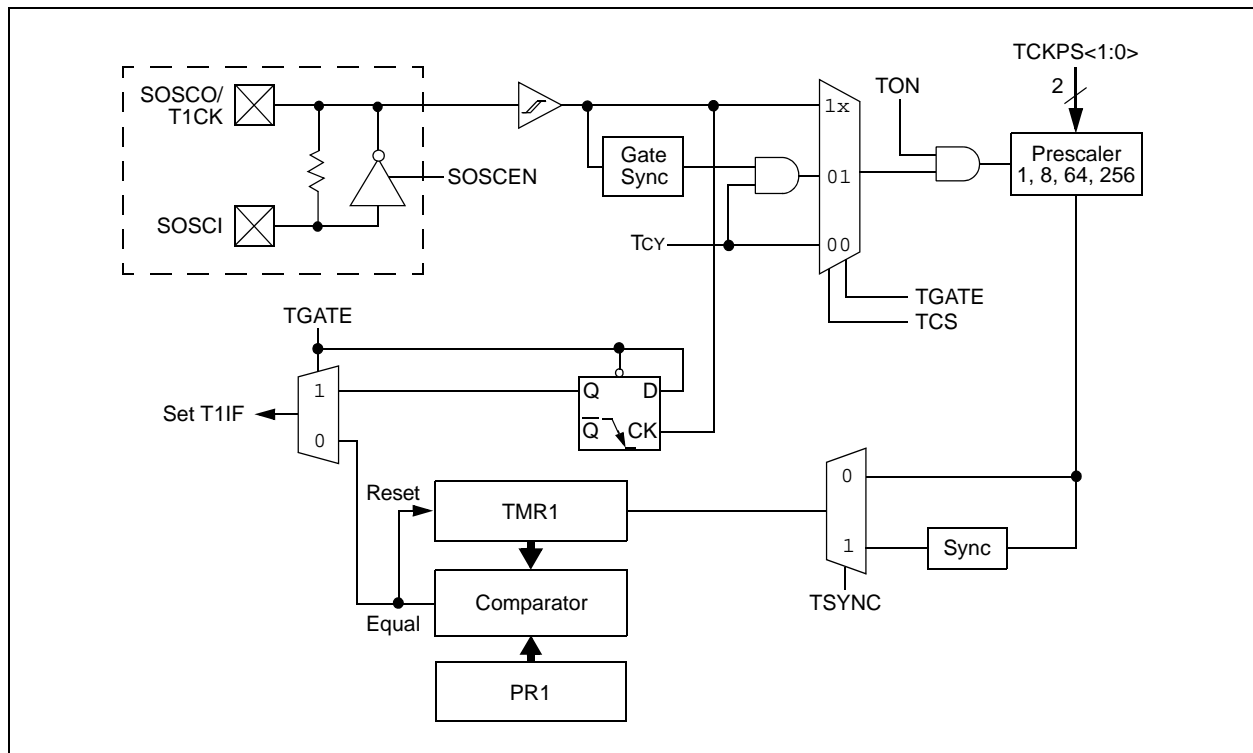
- Timer gate operation
- Selectable prescaler settings
- Timer operation during CPU Idle and Sleep modes
- Interrupt on 16-bit Period register match or falling edge of external gate signal

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

To configure Timer1 for operation:

1. Load the timer value into the TMR1 register.
2. Load the timer period value into the PR1 register.
3. Select the timer prescaler ratio using the TCKPS<1:0> bits in the T1CON register.
4. Set the Clock and Gating modes using the TCS and TGATE bits in the T1CON register.
5. Set or clear the TSYNC bit in T1CON to select synchronous or asynchronous operation.
6. If interrupts are required, set the interrupt enable bit, T1IE. Use the priority bits, T1IP<2:0>, to set the interrupt priority.
7. Set the TON bit (= 1) in the T1CON register.

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



# PIC24FJ16MC101/102

## 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 <sup>(1)</sup>	—	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 <sup>(1)</sup>	—
bit 7						bit 0	

### Legend:

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

- bit 15            **TON:** Timer1 On bit<sup>(1)</sup>  
                   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<sup>(1)</sup>  
                   1 = External clock from pin T1CK (on the rising edge)  
                   0 = Internal clock (FCY)
- bit 0            **Unimplemented:** Read as '0'

**Note 1:** When TCS = 1 and TON = 1, writes to the TMR1 register are inhibited from the CPU.

## 12.0 TIMER2/3 FEATURE

**Note 1:** This data sheet summarizes the features of the PIC24FJ16MC101/102 family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 14. “Timers”** (DS39704) in the *“PIC24F Family Reference Manual”*, which is available from the Microchip web site ([www.microchip.com](http://www.microchip.com)).

**2:** It is important to note that the specifications in **Section 26.0 “Electrical Characteristics”** of this data sheet, supercede any specifications that may be provided in PIC24F Family Reference Manual sections.

**3:** 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 three 2-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 (e.g., 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, 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, 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 bits are 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 timers for 32-bit operation:

1. Set the T32 control bit.
2. Select the prescaler ratio for Timer2 using the TCKPS<1:0> bits.
3. Set the Clock and Gating modes using the corresponding TCS and TGATE bits.
4. Load the timer period value. PR3 contains the msw of the value, while PR2 contains the least significant word (lsw).
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, which always contains the msw of the count, while TMR2 contains the lsw.

### 12.2 16-bit Operation

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

1. Clear the T32 bit corresponding to that timer.
2. Select the timer prescaler ratio using the TCKPS<1:0> bits.
3. Set the Clock and Gating modes using the TCS and TGATE bits.
4. Load the timer period value into the PRx register.
5. If interrupts are required, set the interrupt enable bit, TxIE. Use the priority bits, TxIP<2:0>, to set the interrupt priority.
6. Set the TON bit.

# PIC24FJ16MC101/102

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

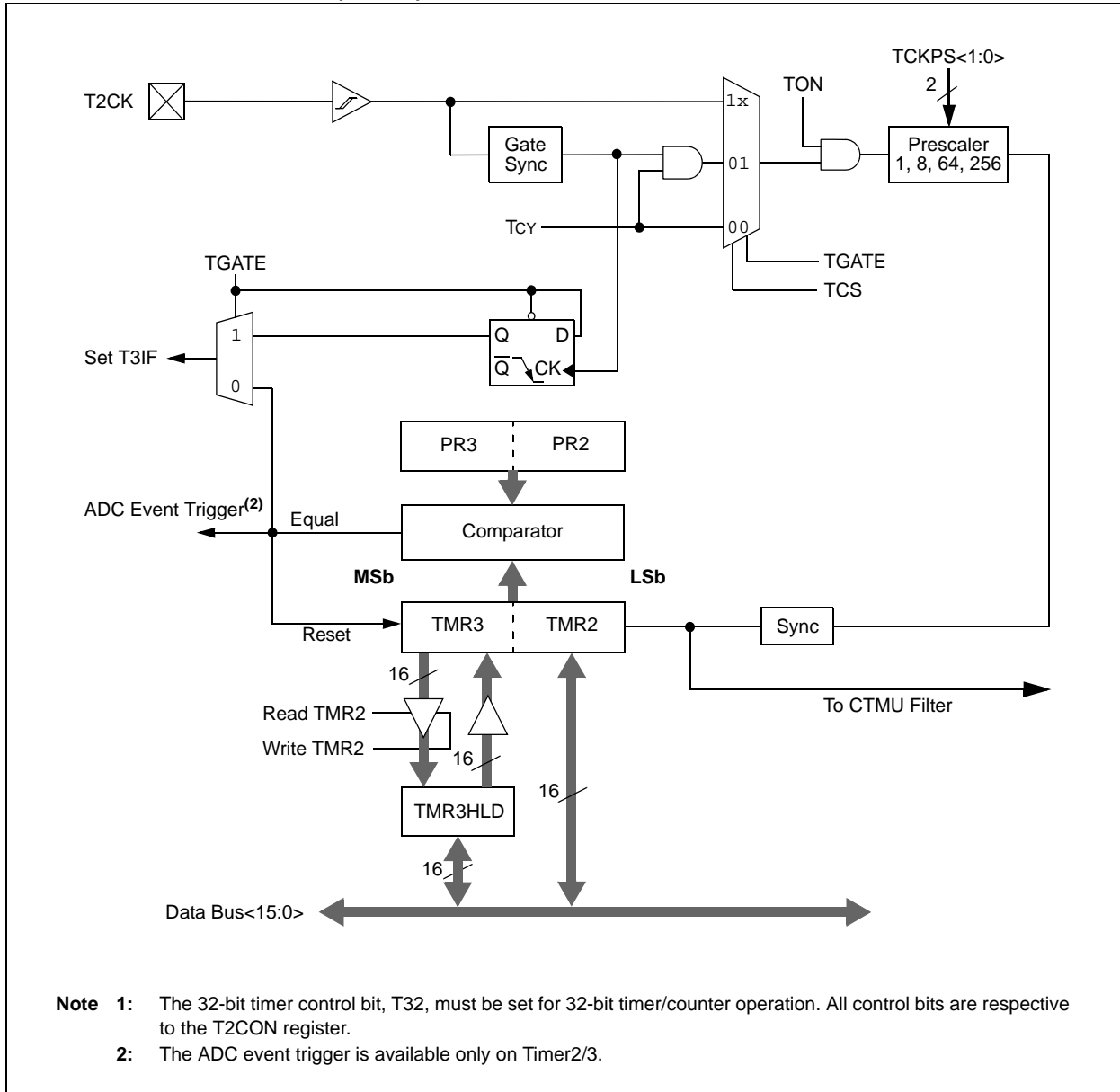


FIGURE 12-2: TIMER2 (16-BIT) BLOCK DIAGRAM

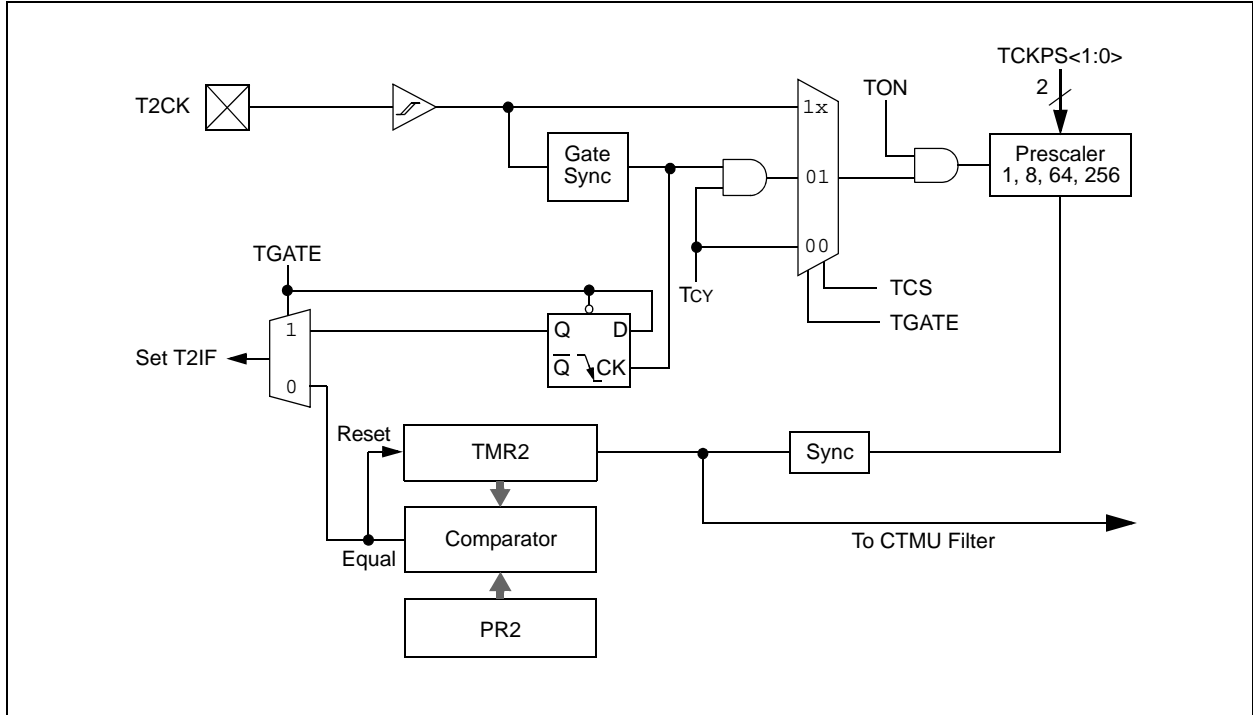
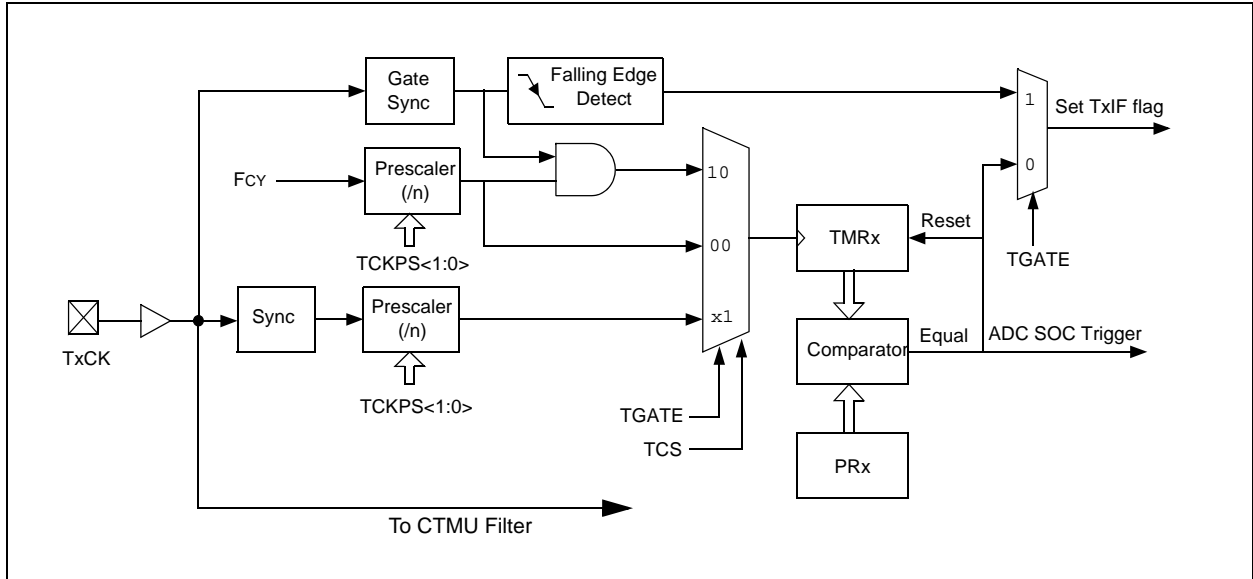


FIGURE 12-3: TIMER3 (16-BIT) BLOCK DIAGRAM



# PIC24FJ16MC101/102

## 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 <sup>(2)</sup>	—	TSIDL <sup>(1)</sup>	—	—	—	—	—
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	U-0
—	TGATE <sup>(2)</sup>	TCKPS<1:0> <sup>(2)</sup>		—	—	TCS <sup>(2)</sup>	—
bit 7							bit 0

### Legend:

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

- bit 15            **TON:** Timer3 On bit<sup>(2)</sup>  
                   1 = Starts 16-bit Timer3  
                   0 = Stops 16-bit Timer3
- bit 14            **Unimplemented:** Read as '0'
- bit 13            **TSIDL:** Stop in Idle Mode bit<sup>(1)</sup>  
                   1 = Discontinue timer operation when device enters Idle mode  
                   0 = Continue timer operation in Idle mode
- bit 12-7        **Unimplemented:** Read as '0'
- bit 6            **TGATE:** Timer3 Gated Time Accumulation Enable bit<sup>(2)</sup>  
                   When TCS = 1:  
                   This bit is ignored.  
                   When TCS = 0:  
                   1 = Gated time accumulation enabled  
                   0 = Gated time accumulation disabled
- bit 5-4        **TCKPS<1:0>:** Timer3 Input Clock Prescale Select bits<sup>(2)</sup>  
                   11 = 1:256 prescale value  
                   10 = 1:64 prescale value  
                   01 = 1:8 prescale value  
                   00 = 1:1 prescale value
- bit 3-2        **Unimplemented:** Read as '0'
- bit 1            **TCS:** Timer3 Clock Source Select bit<sup>(2)</sup>  
                   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.
- Note 2:** When the 32-bit timer operation is enabled (T32 = 1) in the Timer Control register (T2CON<3>), these bits have no effect.

# PIC24FJ16MC101/102

---

---

NOTES:

## 13.0 INPUT CAPTURE

**Note 1:** This data sheet summarizes the features of the PIC24FJ16MC101/102 family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 15. “Input Capture”** (DS39701) in the *“PIC24F Family Reference Manual”*, which is available from the Microchip web site ([www.microchip.com](http://www.microchip.com)).

**2:** It is important to note that the specifications in **Section 26.0 “Electrical Characteristics”** of this data sheet, supercede any specifications that may be provided in PIC24F Family Reference Manual sections.

**3:** 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 PIC24FJ16MC101/102 devices support up to eight input capture channels.

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

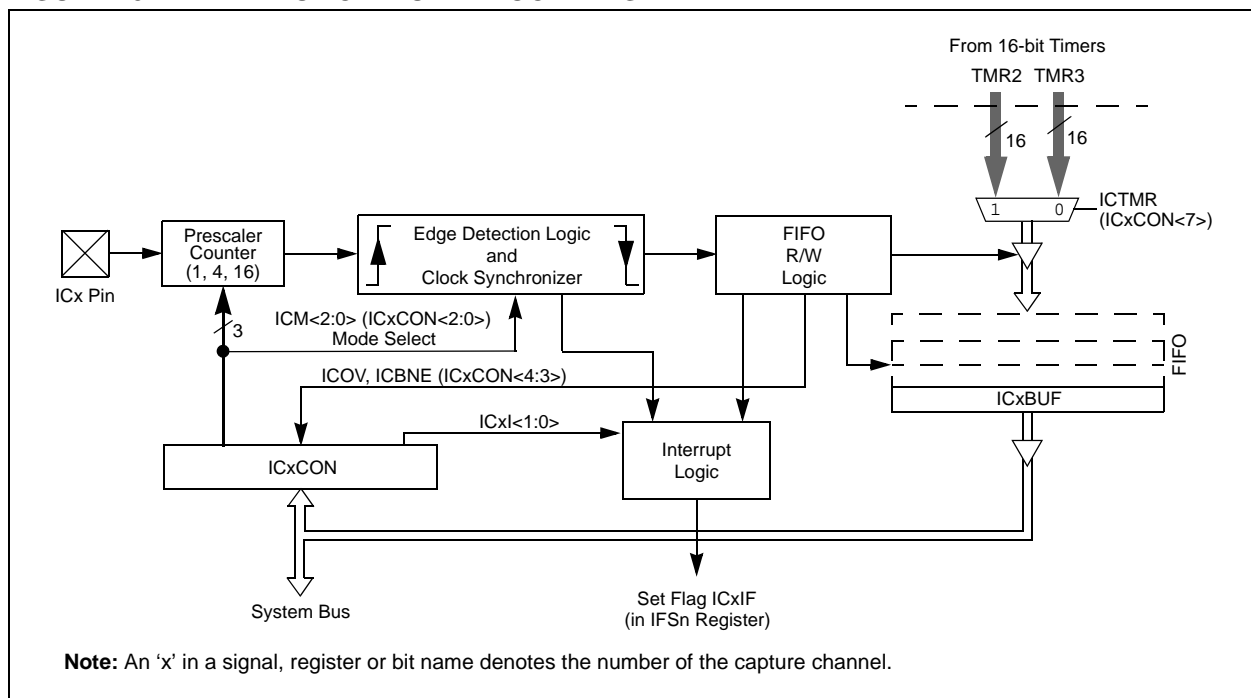
1. Simple Capture Event modes:
  - Capture timer value on every falling edge of input at ICx pin
  - Capture timer value on every rising edge of input at ICx pin
2. Capture timer value on every edge (rising and falling)
3. Prescaler Capture Event modes:
  - Capture timer value on every 4th rising edge of input at ICx pin
  - Capture timer value on every 16th rising edge of input at ICx pin

Each Input Capture channel can select 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**



# PIC24FJ16MC101/102

## 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:**

R = Readable bit                      W = Writable bit                      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 PIC24FJ16MC101/102 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. “Output Compare”** (DS39706) of the “PIC24F Family Reference Manual”, which is available from the Microchip web site ([www.microchip.com](http://www.microchip.com)).

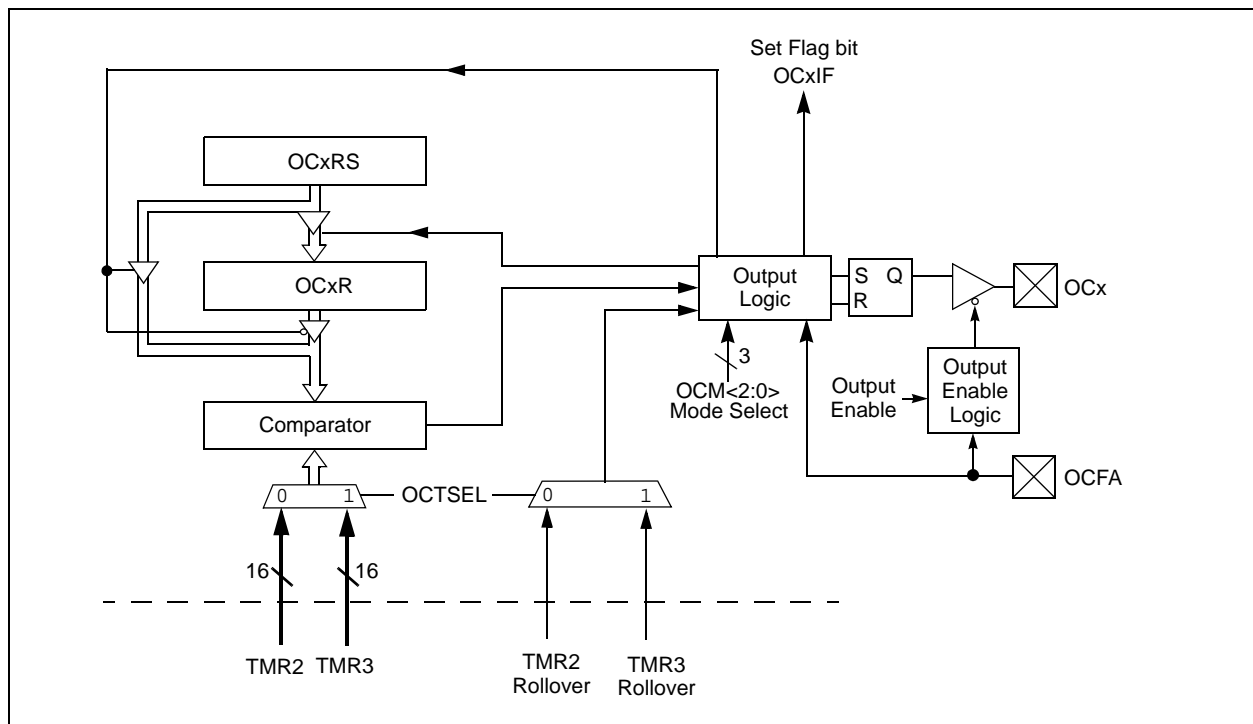
- 2: It is important to note that the specifications in **Section 26.0 “Electrical Characteristics”** of this data sheet, supercede any specifications that may be provided in PIC24F Family Reference Manual sections.
- 3: Some registers and associated bits described in this section may not be available 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**



# PIC24FJ16MC101/102

## 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

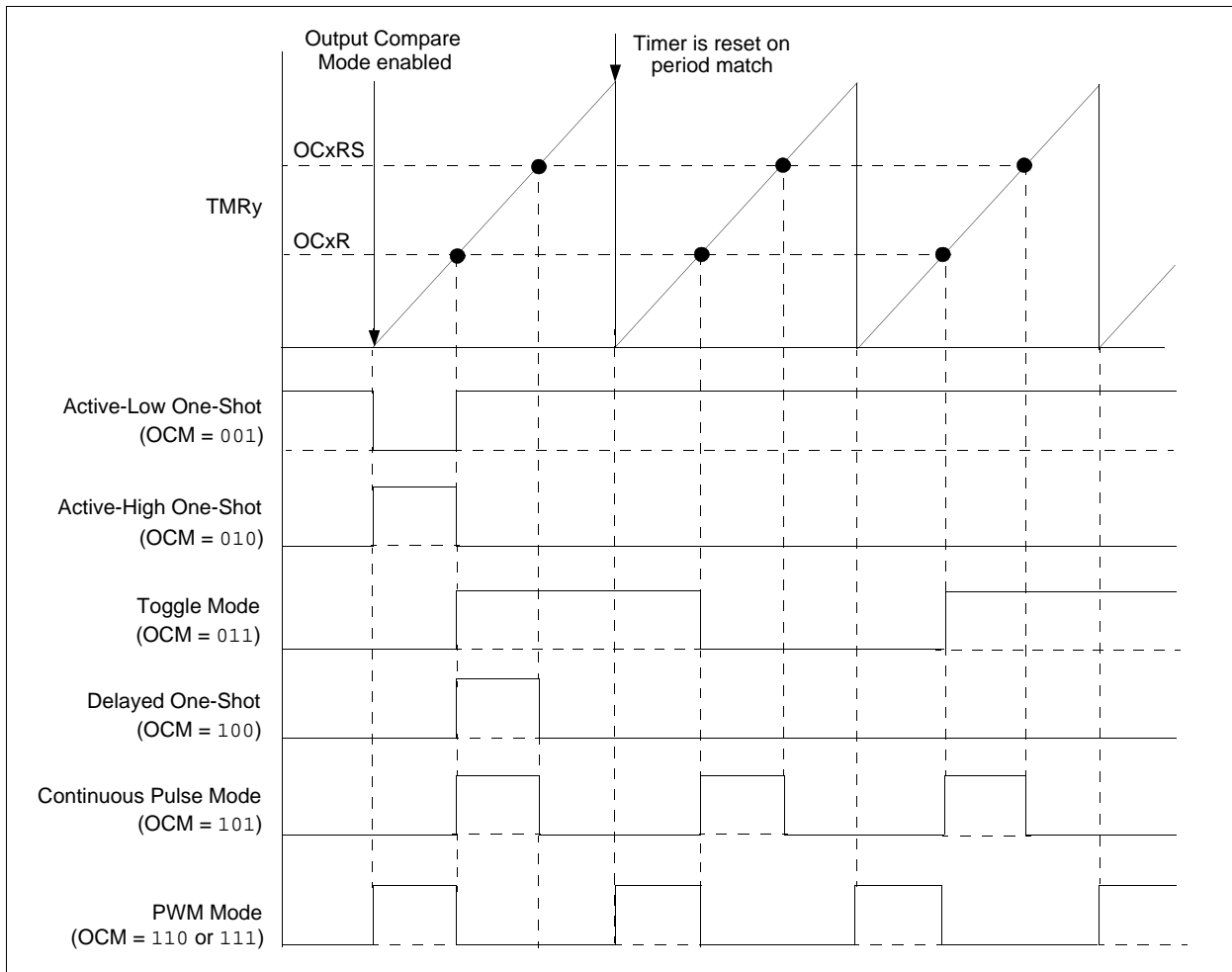
application must disable the associated timer when writing to the output compare control registers to avoid malfunctions.

**Note:** See Section 16. “Output Compare” (DS39706) in the “PIC24F Family Reference Manual” (DS70209) 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**



## 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	

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

- bit 15-14      **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

# PIC24FJ16MC101/102

---

---

NOTES:



## 15.0 MOTOR CONTROL PWM MODULE

**Note 1:** This data sheet summarizes the features of the PIC24FJ16MC101/102 family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 47. “Motor Control PWM”** (DS39735), in the “*PIC24F Family Reference Manual*”, which is available on the Microchip web site ([www.microchip.com](http://www.microchip.com)).

**2:** It is important to note that the specifications in **Section 26.0 “Electrical Characteristics”** of this data sheet, supercede any specifications that may be provided in PIC24F Family Reference Manual sections.

**3:** 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 PIC24FJ16MC101/102 devices have a 6-channel Pulse-Width Modulation (PWM) module.

The PWM module has the following features:

- Up to 16-bit resolution
- On-the-fly PWM frequency changes
- Edge-Aligned and Center-Aligned Output modes
- Single Pulse Generation mode
- Interrupt support for asymmetrical updates in Center-Aligned mode
- Output override control for Electrically Commutative Motor (ECM) operation or BLDC
- Special Event comparator for scheduling other peripheral events
- Fault pins to optionally drive each of the PWM output pins to a defined state
- Duty cycle updates configurable to be immediate or synchronized to the PWM time base

## 15.1 PWM1: 6-Channel PWM Module

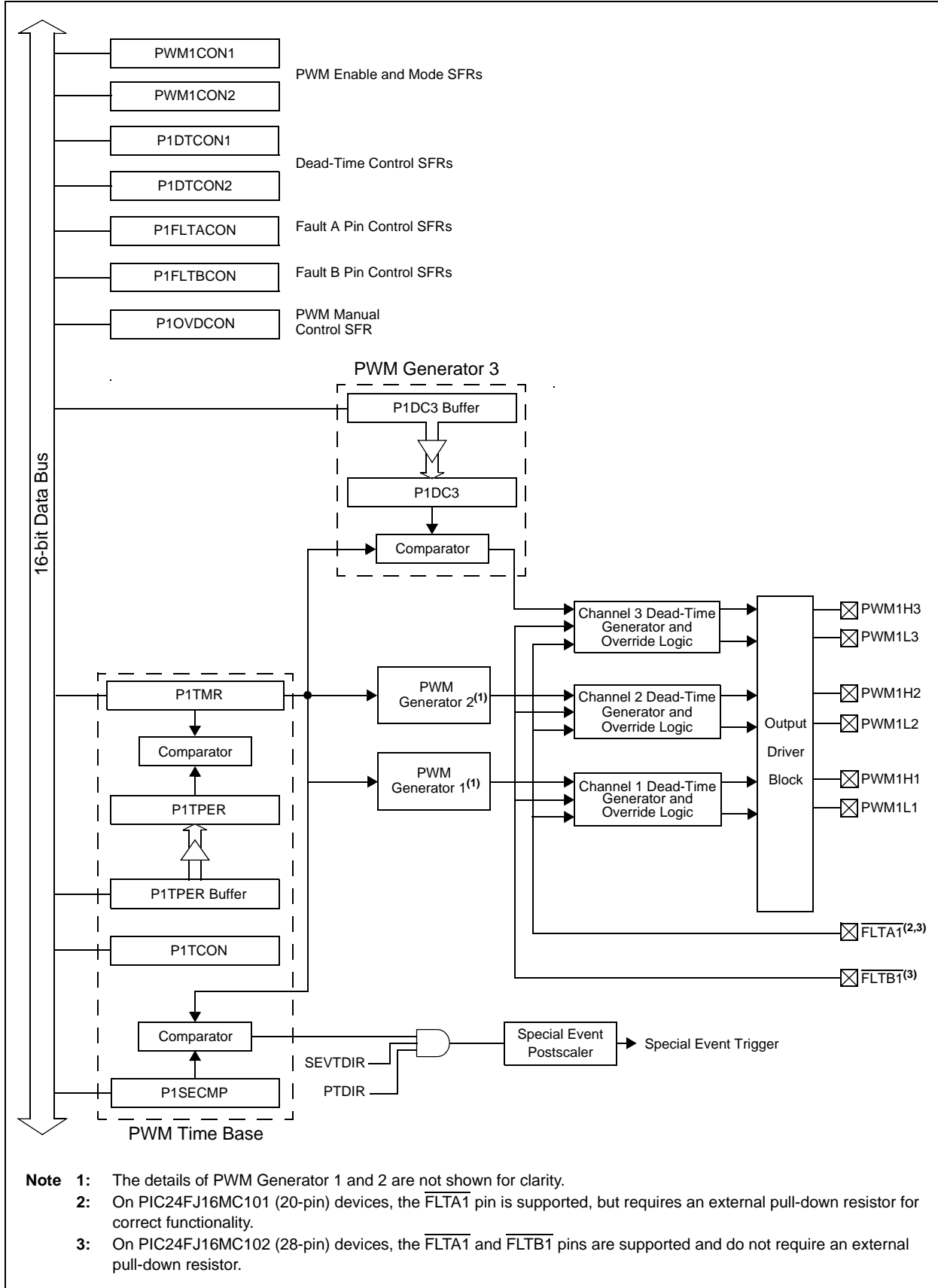
This module simplifies the task of generating multiple synchronized PWM outputs. The following power and motion control applications are supported by the PWM module:

- 3-Phase AC Induction Motor
- Switched Reluctance (SR) Motor
- Brushless DC (BLDC) Motor
- Uninterruptible Power Supply (UPS)

This module contains three duty cycle generators, numbered 1 through 3. The module has six PWM output pins, numbered PWM1H1/PWM1L1 through PWM1H3/PWM1L3. The six I/O pins are grouped into high/low numbered pairs, denoted by the suffix H or L, respectively. For complementary loads, the low PWM pins are always the complement of the corresponding high I/O pin.

# PIC24FJ16MC101/102

**FIGURE 15-1: 6-CHANNEL PWM MODULE BLOCK DIAGRAM (PWM1)**



## 15.2 PWM Faults

The Motor Control PWM module incorporates up to two fault inputs,  $\overline{FLTA1}$  and  $\overline{FLTB1}$ . These fault inputs are implemented with Class B safety features. These features ensure that the PWM outputs enter a safe state when either of the fault inputs is asserted.

The  $\overline{FLTA}$  and  $\overline{FLTB}$  pins, when enabled and having ownership of a pin, also enable a soft internal pull-down resistor. The soft pull-down provides a safety feature by automatically asserting the fault should a break occur in the fault signal connection.

The implementation of internal pull-down resistors is dependent on the device variant. Table 15-1 describes which devices and pins implement the internal pull-down resistors.

**TABLE 15-1: INTERNAL PULL-DOWN RESISTORS ON PWM FAULT PINS**

Device	Fault Pin	Internal Pull-down Implemented?
PIC24FJ16MC101	$\overline{FLTA1}$	No
PIC24FJ16MC102	$\overline{FLTA1}$	Yes
	$\overline{FLTB1}$	Yes

On devices without internal pull-downs on the Fault pin, it is recommended to connect an external pull-down resistor for Class B safety features.

### 15.2.1 PWM FAULTS AT RESET

During any reset event, the PWM module maintains ownership of both PWM Fault pins. At reset, both faults are enabled in latched mode to guarantee the fail-safe power-up of the application. The application software must clear both the PWM faults before enabling the Motor Control PWM module.

The Fault condition must be cleared by the external circuitry driving the fault input pin high and clearing the fault interrupt flag. After the fault pin condition has been cleared, the PWM module restores the PWM output signals on the next PWM period or half-period boundary.

Refer to Section 47. “Motor Control PWM” (DS39735), in the “PIC24F Family Reference Manual” for more information on the PWM faults.

**Note:** The number of PWM faults mapped to the device pins depend on the specific variant. Regardless of the variant, both faults will be enabled during any reset event. The application must clear both  $\overline{FLTA1}$  and  $\overline{FLTB1}$  before enabling the Motor Control PWM module. Refer to the specific device pin diagrams to see which fault pins are mapped to the device pins.

## 15.3 Write-protected Registers

On PIC24FJ16MC101/102 devices, write protection is implemented for the PWMxCON1, PxFLTACON and PxFLTBCON registers. The write protection feature prevents any inadvertent writes to these registers. The write protection feature can be controlled by the PWMLOCK configuration bit in the FOSCSEL configuration register. The default state of the write protection feature is enabled (PWMLOCK = 1). The write protection feature can be disabled by configuring PWMLOCK (FOSCSEL<6>) = 0.

The user application can gain access to these locked registers either by configuring the PWMLOCK (FOSCSEL<6>) = 0, or by performing the unlock sequence. To perform the unlock sequence, the user application must write two consecutive values of (0xABCD and 0x4321) to the PWMxKEY register to perform the unlock operation. The write access to the PWMxCON1, PxFLTACON or PxFLTBCON registers must be the next SFR access following the unlock process. There can be no other SFR accesses during the unlock process and subsequent write access.

To write to all registers, the PWMxCON1, PxFLTACON and PxFLTBCON registers require three unlock operations.

The correct unlocking sequence is described in Example 15-1 and Example 15-2.

# PIC24FJ16MC101/102

## EXAMPLE 15-1: ASSEMBLY CODE EXAMPLE FOR WRITE-PROTECTED REGISTER UNLOCK AND FAULT CLEARING SEQUENCE

```
; FLTAl pin must be pulled high externally in order to clear and disable the fault
; Writing to P1FLTAcon register requires unlock sequence

mov #0xabcd,w10      ; Load first unlock key to w10 register
mov #0x4321,w11      ; Load second unlock key to w11 register
mov #0x0000,w0       ; Load desired value of P1FLTAcon register in w0
mov w10, PWM1KEY     ; Write first unlock key to PWM1KEY register
mov w11, PWM1KEY     ; Write second unlock key to PWM1KEY register
mov w0,P1FLTAcon    ; Write desired value to P1FLTAcon register

; FLTB1 pin must be pulled high externally in order to clear and disable the fault
; Writing to P1FLTBcon register requires unlock sequence

mov #0xabcd,w10      ; Load first unlock key to w10 register
mov #0x4321,w11      ; Load second unlock key to w11 register
mov #0x0000,w0       ; Load desired value of P1FLTBcon register in w0
mov w10, PWM1KEY     ; Write first unlock key to PWM1KEY register
mov w11, PWM1KEY     ; Write second unlock key to PWM1KEY register
mov w0,P1FLTBcon    ; Write desired value to P1FLTBcon register

; Enable all PWMs using PWM1CON1 register
; Writing to PWM1CON1 register requires unlock sequence

mov #0xabcd,w10      ; Load first unlock key to w10 register
mov #0x4321,w11      ; Load second unlock key to w11 register
mov #0x0077,w0       ; Load desired value of PWM1CON1 register in w0
mov w10, PWM1KEY     ; Write first unlock key to PWM1KEY register
mov w11, PWM1KEY     ; Write second unlock key to PWM1KEY register
mov w0,PWM1CON1     ; Write desired value to PWM1CON1 register
```

## EXAMPLE 15-2: C CODE EXAMPLE FOR WRITE-PROTECTED REGISTER UNLOCK AND FAULT CLEARING SEQUENCE

```
// FLTAl pin must be pulled high externally in order to clear and disable the fault
// Writing to P1FLTAcon register requires unlock sequence
// Use builtin function to write 0x0000 to P1FLTAcon register
__builtin_write_PWMSFR(&P1FLTAcon, 0x0000, &PWM1KEY);

// FLTB1 pin must be pulled high externally in order to clear and disable the fault
// Writing to P1FLTBcon register requires unlock sequence
// Use builtin function to write 0x0000 to P1FLTBcon register
__builtin_write_PWMSFR(&P1FLTBcon, 0x0000, &PWM1KEY);

// Enable all PWMs using PWM1CON1 register
// Writing to PWM1CON1 register requires unlock sequence
// Use builtin function to write 0x0077 to PWM1CON1 register
__builtin_write_PWMSFR(&PWM1CON1, 0x0077, &PWM1KEY);
```

## REGISTER 15-1: P<sub>x</sub>TCON: PWM TIME BASE CONTROL REGISTER

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
PTEN	—	PTSIDL	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTOPS<3:0>			PTCKPS<1:0>		PTMOD<1:0>		
bit 7							bit 0

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

- bit 15      **PTEN:** PWM Time Base Timer Enable bit  
             1 = PWM time base is on  
             0 = PWM time base is off
- bit 14      **Unimplemented:** Read as '0'
- bit 13      **PTSIDL:** PWM Time Base Stop in Idle Mode bit  
             1 = PWM time base halts in CPU Idle mode  
             0 = PWM time base runs in CPU Idle mode
- bit 12-8    **Unimplemented:** Read as '0'
- bit 7-4     **PTOPS<3:0>:** PWM Time Base Output Postscale Select bits  
             1111 = 1:16 postscale  
             •  
             •  
             •  
             0001 = 1:2 postscale  
             0000 = 1:1 postscale
- bit 3-2     **PTCKPS<1:0>:** PWM Time Base Input Clock Prescale Select bits  
             11 = PWM time base input clock period is 64 T<sub>CY</sub> (1:64 prescale)  
             10 = PWM time base input clock period is 16 T<sub>CY</sub> (1:16 prescale)  
             01 = PWM time base input clock period is 4 T<sub>CY</sub> (1:4 prescale)  
             00 = PWM time base input clock period is T<sub>CY</sub> (1:1 prescale)
- bit 1-0     **PTMOD<1:0>:** PWM Time Base Mode Select bits  
             11 = PWM time base operates in a Continuous Up/Down Count mode with interrupts for double PWM updates  
             10 = PWM time base operates in a Continuous Up/Down Count mode  
             01 = PWM time base operates in Single Pulse mode  
             00 = PWM time base operates in a Free-Running mode

# PIC24FJ16MC101/102

## REGISTER 15-2: P<sub>x</sub>TMR: PWM TIMER COUNT VALUE REGISTER

R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTDIR	PTMR<14:8>						
bit 15	bit 8						

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTMR<7:0>							
bit 7	bit 0						

### Legend:

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

bit 15                      **PTDIR:** PWM Time Base Count Direction Status bit (read-only)

1 = PWM time base is counting down  
 0 = PWM time base is counting up

bit 14-0                      **PTMR <14:0>:** PWM Time Base Register Count Value bits

## REGISTER 15-3: P<sub>x</sub>TPER: PWM TIME BASE PERIOD REGISTER

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	PTPER<14:8>						
bit 15	bit 8						

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTPER<7:0>							
bit 7	bit 0						

### Legend:

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

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

bit 14-0                      **PTPER<14:0>:** PWM Time Base Period Value bits

## REGISTER 15-4: PxSECMP: SPECIAL EVENT COMPARE REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SEVTDIR <sup>(1)</sup>	SEVTCMP<14:8> <sup>(2)</sup>						
bit 15	bit 8						

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SEVTCMP<7:0> <sup>(2)</sup>							
bit 7	bit 0						

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

bit 15      **SEVTDIR:** Special Event Trigger Time Base Direction bit<sup>(1)</sup>  
 1 = A Special Event Trigger will occur when the PWM time base is counting down  
 0 = A Special Event Trigger will occur when the PWM time base is counting up

bit 14-0      **SEVTCMP<14:0>:** Special Event Compare Value bits<sup>(2)</sup>

- Note 1:** SEVTDIR is compared with PTDIR (PxTMR<15>) to generate the Special Event Trigger.  
**Note 2:** PxSECMP<14:0> is compared with PxTMR<14:0> to generate the Special Event Trigger.

# PIC24FJ16MC101/102

## REGISTER 15-5: PWMxCON1: PWM CONTROL REGISTER 1<sup>(1)</sup>

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	PMOD3	PMOD2	PMOD1
bit 15					bit 8		

U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
—	PEN3H	PEN2H	PEN1H	—	PEN3L	PEN2L	PEN1L
bit 7				bit 0			

### Legend:

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

- bit 15-11        **Unimplemented:** Read as '0'
- bit 10-8        **PMOD3:PMOD1:** PWM I/O Pair Mode bits  
                   1 = PWM I/O pin pair is in the Independent PWM Output mode  
                   0 = PWM I/O pin pair is in the Complementary Output mode
- bit 7            **Unimplemented:** Read as '0'
- bit 6-4        **PEN3H:PEN1H:** PWMxH I/O Enable bits  
                   1 = PWMxH pin is enabled for PWM output  
                   0 = PWMxH pin disabled, I/O pin becomes general purpose I/O
- bit 3            **Unimplemented:** Read as '0'
- bit 2-0        **PEN3L:PEN1L:** PWMxL I/O Enable bits  
                   1 = PWMxL pin is enabled for PWM output  
                   0 = PWMxL pin disabled, I/O pin becomes general purpose I/O

**Note 1:** The PWMxCON1 register is a write-protected register. Refer to [Section 15.3 “Write-protected Registers”](#) for more information on the unlock sequence.



## REGISTER 15-6: PWMxCON2: PWM CONTROL REGISTER 2

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	SEVOPS<3:0>			
bit 15				bit 8			

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	IUE	OSYNC	UDIS
bit 7				bit 0			

### Legend:

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

- bit 15-12      **Unimplemented:** Read as '0'
- bit 11-8      **SEVOPS<3:0>:** PWM Special Event Trigger Output Postscale Select bits
  - 1111 = 1:16 postscale
  - 
  - 
  - 
  - 0001 = 1:2 postscale
  - 0000 = 1:1 postscale
- bit 7-3      **Unimplemented:** Read as '0'
- bit 2      **IUE:** Immediate Update Enable bit
  - 1 = Updates to the active PxDC registers are immediate
  - 0 = Updates to the active PxDC registers are synchronized to the PWM time base
- bit 1      **OSYNC:** Output Override Synchronization bit
  - 1 = Output overrides via the PxOVDCON register are synchronized to the PWM time base
  - 0 = Output overrides via the PxOVDCON register occur on next Tcy boundary
- bit 0      **UDIS:** PWM Update Disable bit
  - 1 = Updates from Duty Cycle and Period Buffer registers are disabled
  - 0 = Updates from Duty Cycle and Period Buffer registers are enabled

# PIC24FJ16MC101/102

## REGISTER 15-7: PxDTCON1: DEAD-TIME CONTROL REGISTER 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DTBPS<1:0>		DTB<5:0>					
bit 15		bit 8					

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DTAPS<1:0>		DTA<5:0>					
bit 7		bit 0					

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 15-14 **DTBPS<1:0>**: Dead-Time Unit B Prescale Select bits

11 = Clock period for Dead-Time Unit B is 8 TCY

10 = Clock period for Dead-Time Unit B is 4 TCY

01 = Clock period for Dead-Time Unit B is 2 TCY

00 = Clock period for Dead-Time Unit B is TCY

bit 13-8 **DTB<5:0>**: Unsigned 6-bit Dead-Time Value for Dead-Time Unit B bits

bit 7-6 **DTAPS<1:0>**: Dead-Time Unit A Prescale Select bits

11 = Clock period for Dead-Time Unit A is 8 TCY

10 = Clock period for Dead-Time Unit A is 4 TCY

01 = Clock period for Dead-Time Unit A is 2 TCY

00 = Clock period for Dead-Time Unit A is TCY

bit 5-0 **DTA<5:0>**: Unsigned 6-bit Dead-Time Value for Dead-Time Unit A bits

## REGISTER 15-8: PxDTCON2: DEAD-TIME CONTROL REGISTER 2

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

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	DTS3A	DTS3I	DTS2A	DTS2I	DTS1A	DTS1I
bit 7						bit 0	

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

- bit 15-6      **Unimplemented:** Read as '0'
- bit 5        **DTS3A:** Dead-Time Select for PWM3 Signal Going Active bit  
1 = Dead time provided from Unit B  
0 = Dead time provided from Unit A
- bit 4        **DTS3I:** Dead-Time Select for PWM3 Signal Going Inactive bit  
1 = Dead time provided from Unit B  
0 = Dead time provided from Unit A
- bit 3        **DTS2A:** Dead-Time Select for PWM2 Signal Going Active bit  
1 = Dead time provided from Unit B  
0 = Dead time provided from Unit A
- bit 2        **DTS2I:** Dead-Time Select for PWM2 Signal Going Inactive bit  
1 = Dead time provided from Unit B  
0 = Dead time provided from Unit A
- bit 1        **DTS1A:** Dead-Time Select for PWM1 Signal Going Active bit  
1 = Dead time provided from Unit B  
0 = Dead time provided from Unit A
- bit 0        **DTS1I:** Dead-Time Select for PWM1 Signal Going Inactive bit  
1 = Dead time provided from Unit B  
0 = Dead time provided from Unit A

# PIC24FJ16MC101/102

## REGISTER 15-9: PxFLTAcon: FAULT A CONTROL REGISTER<sup>(1,2,3,4,5)</sup>

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	FAOV3H	FAOV3L	FAOV2H	FAOV2L	FAOV1H	FAOV1L
bit 15						bit 8	

R/W-0	U-0	U-0	U-0	U-0	R/W-1	R/W-1	R/W-1
FLTAM	—	—	—	—	FAEN3	FAEN2	FAEN1
bit 7						bit 0	

### Legend:

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

- bit 15-14      **Unimplemented:** Read as '0'
- bit 13-8      **FAOVxH<3:1>:FAOVxL<3:1>**: Fault Input A PWM Override Value bits  
                     1 = The PWM output pin is driven active on an external Fault input event  
                     0 = The PWM output pin is driven inactive on an external Fault input event
- bit 7          **FLTAM:** Fault A Mode bit  
                     1 = The Fault A input pin functions in the Cycle-by-Cycle mode  
                     0 = The Fault A input pin latches all control pins to the programmed states in PxFLTAcon<13:8>
- bit 6-3      **Unimplemented:** Read as '0'
- bit 2          **FAEN3:** Fault Input A Enable bit  
                     1 = PWMxH3/PWMxL3 pin pair is controlled by Fault Input A  
                     0 = PWMxH3/PWMxL3 pin pair is not controlled by Fault Input A
- bit 1          **FAEN2:** Fault Input A Enable bit  
                     1 = PWMxH2/PWMxL2 pin pair is controlled by Fault Input A  
                     0 = PWMxH2/PWMxL2 pin pair is not controlled by Fault Input A
- bit 0          **FAEN1:** Fault Input A Enable bit  
                     1 = PWMxH1/PWMxL1 pin pair is controlled by Fault Input A  
                     0 = PWMxH1/PWMxL1 pin pair is not controlled by Fault Input A

- Note 1:** On PIC24FJ16MC101 (20-pin) devices, the  $\overline{\text{FLTA1}}$  pin is supported, but requires an external pull-down resistor for correct functionality.
- Note 2:** On PIC24FJ16MC102 (28-pin) devices, the  $\overline{\text{FLTA1}}$  and  $\overline{\text{FLTB1}}$  pins are supported and do not require an external pull-down resistor.
- Note 3:** The PxFLTAcon register is a write-protected register. Refer to [Section 15.3 “Write-protected Registers”](#) for more information on the unlock sequence.
- Note 4:** Comparator outputs are not internally connected to the PWM Fault control logic. If using the Comparator modules for Fault generation, the user must externally connect the desired comparator output pin to the dedicated  $\overline{\text{FLTA1}}$  or  $\overline{\text{FLTB1}}$  input pin.
- Note 5:** During any reset event, the  $\overline{\text{FLTA1}}$  pin is enabled by default and must be cleared as described in [Section 15.2 “PWM Faults”](#).

## REGISTER 15-10: PxFLTBCON: FAULT B CONTROL REGISTER<sup>(1,2,3,4,5)</sup>

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	FBOV3H	FBOV3L	FBOV2H	FBOV2L	FBOV1H	FBOV1L
bit 15							bit 8

R/W-0	U-0	U-0	U-0	U-0	R/W-1	R/W-1	R/W-1
FLTBM	—	—	—	—	FBEN3	FBEN2	FBEN1
bit 7							bit 0

### Legend:

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

- bit 15-14      **Unimplemented:** Read as '0'
- bit 13-8      **FBOVxH<3:1>:FBOVxL<3:1>:** Fault Input B PWM Override Value bits  
                   1 = The PWM output pin is driven active on an external Fault input event  
                   0 = The PWM output pin is driven inactive on an external Fault input event
- bit 7          **FLTBM:** Fault B Mode bit  
                   1 = The Fault B input pin functions in the Cycle-by-Cycle mode  
                   0 = The Fault B input pin latches all control pins to the programmed states in PxFLTBCON<13:8>
- bit 6-3      **Unimplemented:** Read as '0'
- bit 2          **FBEN3:** Fault Input B Enable bit  
                   1 = PWMxH3/PWMxL3 pin pair is controlled by Fault Input B  
                   0 = PWMxH3/PWMxL3 pin pair is not controlled by Fault Input B
- bit 1          **FBEN2:** Fault Input B Enable bit  
                   1 = PWMxH2/PWMxL2 pin pair is controlled by Fault Input B  
                   0 = PWMxH2/PWMxL2 pin pair is not controlled by Fault Input B
- bit 0          **FBEN1:** Fault Input B Enable bit  
                   1 = PWMxH1/PWMxL1 pin pair is controlled by Fault Input B  
                   0 = PWMxH1/PWMxL1 pin pair is not controlled by Fault Input B

- Note 1:** On PIC24FJ16MC101 (20-pin) devices, the  $\overline{\text{FLTA1}}$  pin is supported, but requires an external pull-down resistor for correct functionality.
- 2:** On PIC24FJ16MC102 (28-pin) devices, the  $\overline{\text{FLTA1}}$  and  $\overline{\text{FLTB1}}$  pins are supported and do not require an external pull-down resistor.
- 3:** The PxFLTBACON register is a write-protected register. Refer to [Section 15.3 “Write-protected Registers”](#) for more information on the unlock sequence.
- 4:** Comparator outputs are not internally connected to the PWM Fault control logic. If using the Comparator modules for Fault generation, the user must externally connect the desired comparator output pin to the dedicated  $\overline{\text{FLTA1}}$  or  $\overline{\text{FLTB1}}$  input pin.
- 5:** During any reset event, the  $\overline{\text{FLTB1}}$  pin is enabled by default and must be cleared as described in [Section 15.2 “PWM Faults”](#).

# PIC24FJ16MC101/102

## REGISTER 15-11: P<sub>x</sub>OVDCON: OVERRIDE CONTROL REGISTER

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	
—	—	POVD3H	POVD3L	POVD2H	POVD2L	POVD1H	POVD1L	
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	
—	—	POUT3H	POUT3L	POUT2H	POUT2L	POUT1H	POUT1L	
bit 7								bit 0

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 15-14 **Unimplemented:** Read as '0'

bit 13-8 **POVDxH<3:1>:POVDxL<3:1>:** PWM Output Override bits

1 = Output on PWMx I/O pin is controlled by the PWM generator

0 = Output on PWMx I/O pin is controlled by the value in the corresponding POUTxH:POUTxL bit

bit 7-6 **Unimplemented:** Read as '0'

bit 5-0 **POUTxH<3:1>:POUTxL<3:1>:** PWM Manual Output bits

1 = PWMx I/O pin is driven active when the corresponding POVDxH:POVDxL bit is cleared

0 = PWMx I/O pin is driven inactive when the corresponding POVDxH:POVDxL bit is cleared

## REGISTER 15-12: PxDC1: PWM DUTY CYCLE REGISTER 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PDC1<15:8>								
bit 15								bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PDC1<7:0>								
bit 7								bit 0

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

bit 15-0      **PDC1<15:0>**: PWM Duty Cycle 1 Value bits

## REGISTER 15-13: PxDC2: PWM DUTY CYCLE REGISTER 2

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PDC2<15:8>								
bit 15								bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PDC2<7:0>								
bit 7								bit 0

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

bit 15-0      **PDC2<15:0>**: PWM Duty Cycle 2 Value bits

## REGISTER 15-14: PxDC3: PWM DUTY CYCLE REGISTER 3

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PDC3<15:8>								
bit 15								bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PDC3<7:0>								
bit 7								bit 0

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

bit 15-0      **PDC3<15:0>**: PWM Duty Cycle 3 Value bits

# PIC24FJ16MC101/102

## REGISTER 15-15: PWMxKEY: PWM KEY UNLOCK REGISTER<sup>(1)</sup>

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PWMKEY<15:8>							
bit 15				bit 8			

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PWMKEY<7:0>							
bit 7				bit 0			

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 15-0

**PWMKEY<15:0>**: PWM Key Unlock bits

If the PWMLOCK Configuration bit is asserted (PWMLOCK = 1), the PWMxCON1, PxFLTACON and PxFLTBCON registers are writable only after the proper sequence is written to the PWMxKEY register.

If the PWMLOCK Configuration bit is deasserted (PWMLOCK = 0) the PWMxCON1, PxFLTACON and PxFLTBCON registers are writable at all times.

Refer to **Section 47. "Motor Control PWM"** (DS39735) in the *"PIC24F Family Reference Manual"* for further details about the unlock sequence.



## 16.0 SERIAL PERIPHERAL INTERFACE (SPI)

**Note 1:** This data sheet summarizes the features of the PIC24FJ16MC101/102 family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 23. “Serial Peripheral Interface (SPI)”** (DS39699) in the *“PIC24F Family Reference Manual”*, which is available from the Microchip web site ([www.microchip.com](http://www.microchip.com)).

**2:** It is important to note that the specifications in **Section 26.0 “Electrical Characteristics”** of this data sheet, supercede any specifications that may be provided in PIC24F Family Reference Manual sections.

**3:** 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, etc. The SPI module is compatible with SPI and SIO<sup>®</sup> from Motorola<sup>®</sup>.

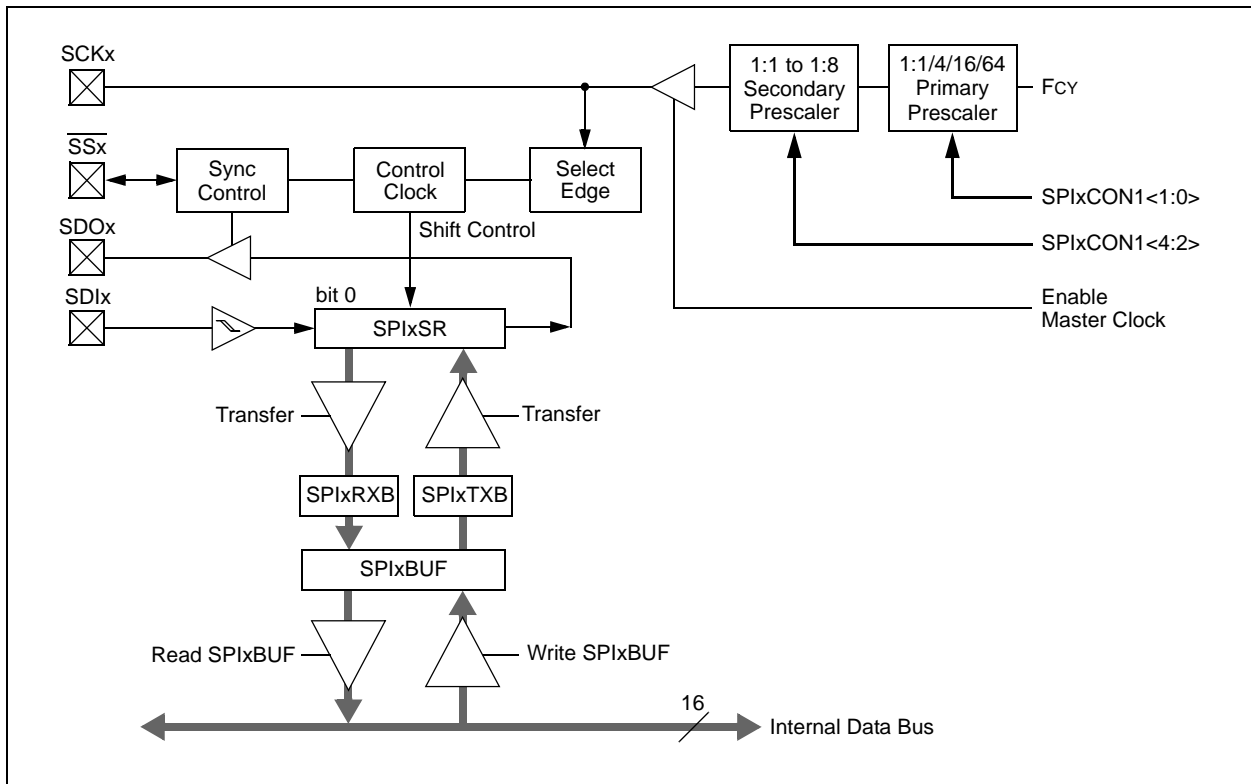
Each SPI module consists of a 16-bit shift register, SPIxSR (where x = 1 or 2), used for shifting data in and out, and a buffer register, SPIxBUF. A control register, SPIxCON, configures the module. Additionally, a status register, SPIxSTAT, indicates status conditions.

The serial interface consists of four pins:

- SDIx (serial data input)
- SDOx (serial data output)
- SCKx (shift clock input or output)
- SSx (active low slave select).

In Master mode operation, SCK is a clock output. In Slave mode, it is a clock input.

**FIGURE 16-1: SPI MODULE BLOCK DIAGRAM**



# PIC24FJ16MC101/102

## REGISTER 16-1: SPIxSTAT: SPIx STATUS AND CONTROL REGISTER

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

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

<b>Legend:</b>	C = Clearable bit		
R = Readable bit	W = Writable bit	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  $\overline{SSx}$  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 16-2: SPIxCON1: SPIx CONTROL REGISTER 1

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	DISSCK	DISSDO	MODE16	SMP	CKE <sup>(1)</sup>
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SSEN <sup>(2)</sup>	CKP	MSTEN	SPRE<2:0> <sup>(3)</sup>			PPRE<1:0> <sup>(3)</sup>	
bit 7							bit 0

### Legend:

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

- bit 15-13      **Unimplemented:** Read as '0'
- bit 12      **DISSCK:** Disable SCKx pin bit (SPI Master modes only)  
                  1 = Internal SPI clock is disabled, pin functions as I/O  
                  0 = Internal SPI clock is enabled
- bit 11      **DISSDO:** Disable SDOx pin bit  
                  1 = SDOx pin is not used by module; pin functions as I/O  
                  0 = SDOx pin is controlled by the module
- bit 10      **MODE16:** Word/Byte Communication Select bit  
                  1 = Communication is word-wide (16 bits)  
                  0 = Communication is byte-wide (8 bits)
- bit 9      **SMP:** SPIx Data Input Sample Phase bit  
                  Master mode:  
                  1 = Input data sampled at end of data output time  
                  0 = Input data sampled at middle of data output time  
                  Slave mode:  
                  SMP must be cleared when SPIx is used in Slave mode.
- bit 8      **CKE:** SPIx Clock Edge Select bit<sup>(1)</sup>  
                  1 = Serial output data changes on transition from active clock state to Idle clock state (see bit 6)  
                  0 = Serial output data changes on transition from Idle clock state to active clock state (see bit 6)
- bit 7      **SSEN:** Slave Select Enable bit (Slave mode)  
                  1 =  $\overline{SSx}$  pin used for Slave mode  
                  0 =  $\overline{SSx}$  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.

# PIC24FJ16MC101/102

---

## REGISTER 16-2: SPIxCON1: SPIx CONTROL REGISTER 1 (CONTINUED)

bit 4-2      **SPRE<2:0>**: Secondary Prescale bits (Master mode)<sup>(3)</sup>

111 = Secondary prescale 1:1

110 = Secondary prescale 2:1

•

•

•

000 = Secondary prescale 8:1

bit 1-0      **PPRE<1:0>**: Primary Prescale bits (Master mode)<sup>(3)</sup>

11 = Primary prescale 1:1

10 = Primary prescale 4:1

01 = Primary prescale 16:1

00 = Primary prescale 64:1

**Note 1:** The CKE bit is not used in the Framed SPI modes. Program this bit to '0' for the Framed SPI modes (FRMEN = 1).

**2:** This bit must be cleared when FRMEN = 1.

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

## REGISTER 16-3: SPIxCON2: SPIx CONTROL REGISTER 2

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0
FRMEN	SPIFSD	FRMPOL	—	—	—	—	—
bit 15						bit 8	

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

### Legend:

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

- bit 15            **FRMEN:** Framed SPIx Support bit  
                   1 = Framed SPIx support enabled ( $\overline{SSx}$  pin used as frame sync pulse input/output)  
                   0 = Framed SPIx support disabled
- bit 14            **SPIFSD:** Frame Sync Pulse Direction Control bit  
                   1 = Frame sync pulse input (slave)  
                   0 = Frame sync pulse output (master)
- bit 13            **FRMPOL:** Frame Sync Pulse Polarity bit  
                   1 = Frame sync pulse is active-high  
                   0 = Frame sync pulse is active-low
- bit 12-2        **Unimplemented:** Read as '0'
- bit 1            **FRMDLY:** Frame Sync Pulse Edge Select bit  
                   1 = Frame sync pulse coincides with first bit clock  
                   0 = Frame sync pulse precedes first bit clock
- bit 0            **Unimplemented:** This bit must not be set to '1' by the user application.

# PIC24FJ16MC101/102

---

---

NOTES:

## 17.0 INTER-INTEGRATED CIRCUIT™ (I<sup>2</sup>C™)

**Note 1:** This data sheet summarizes the features of the PIC24FJ16MC101/102 family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 24. “Inter-Integrated Circuit™ (I<sup>2</sup>C™)”** (DS39702) in the *“PIC24F Family Reference Manual”*, which is available from the Microchip web site ([www.microchip.com](http://www.microchip.com)).

**2:** It is important to note that the specifications in **Section 26.0 “Electrical Characteristics”** of this data sheet, supercede any specifications that may be provided in PIC24F Family Reference Manual sections.

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

The Inter-Integrated Circuit™ (I<sup>2</sup>C™) module provides complete hardware support for both Slave and Multi-Master modes of the I<sup>2</sup>C serial communication standard, with a 16-bit interface.

The I<sup>2</sup>C module has a 2-pin interface:

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

The I<sup>2</sup>C module offers the following key features:

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

## 17.1 Operating Modes

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

The I<sup>2</sup>C module can operate either as a slave or a master on an I<sup>2</sup>C bus.

The following types of I<sup>2</sup>C operation are supported:

- I<sup>2</sup>C slave operation with 7-bit addressing
- I<sup>2</sup>C slave operation with 10-bit addressing
- I<sup>2</sup>C master operation with 7-bit or 10-bit addressing

For details about the communication sequence in each of these modes, refer to the Microchip web site ([www.microchip.com](http://www.microchip.com)) for the latest *“PIC24F Family Reference Manual”* sections.

## 17.2 I<sup>2</sup>C Registers

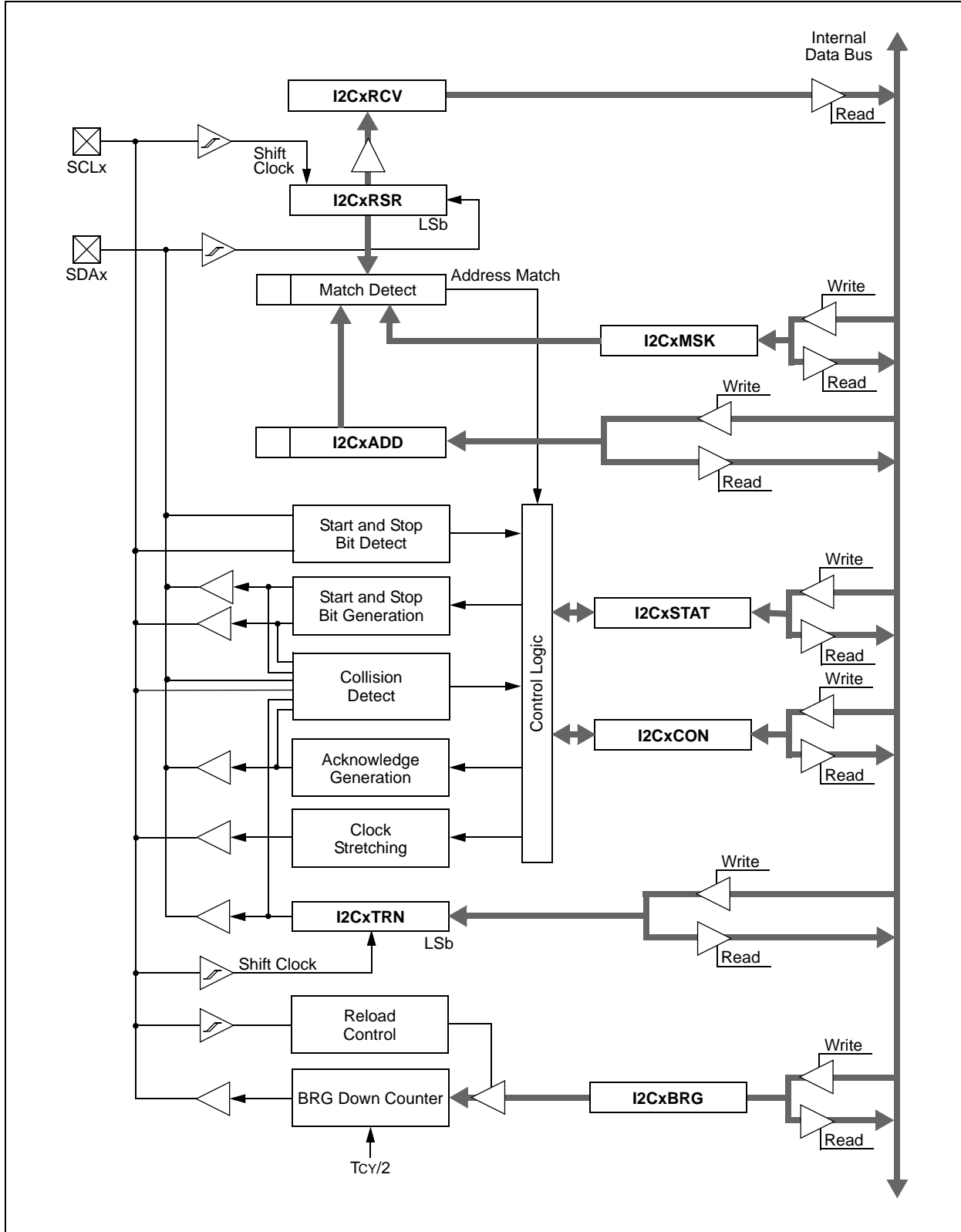
I2CxCON and I2CxSTAT are control and status registers, respectively. The I2CxCON register is readable and writable. The lower six bits of I2CxSTAT are read-only. The remaining bits of the I2CxSTAT are read/write:

- I2CxRSR is the shift register used for shifting data
- 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
- I2CxADD register holds the slave address
- ADD10 status bit 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.

# PIC24FJ16MC101/102

FIGURE 17-1: I<sup>2</sup>C™ BLOCK DIAGRAM (x = 1)





## REGISTER 17-1: I2CxCON: I2Cx CONTROL REGISTER

R/W-0	U-0	R/W-0	R/W-1 HC	R/W-0	R/W-0	R/W-0	R/W-0
I2CEN	—	I2CSIDL	SCLREL	IPMIEN	A10M	DISSLW	SMEN
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0 HC	R/W-0 HC	R/W-0 HC	R/W-0 HC	R/W-0 HC
GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN
bit 7							bit 0

<b>Legend:</b>	U = Unimplemented bit, read as '0'		
R = Readable bit	W = Writable bit	HS = 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<sup>2</sup>C pins are controlled by port functions.
- bit 14      **Unimplemented:** Read as '0'
- bit 13      **I2CSIDL:** Stop in Idle Mode bit  
 1 = Discontinue module operation when device enters an Idle mode  
 0 = Continue module operation in Idle mode
- bit 12      **SCLREL:** SCLx Release Control bit (when operating as I<sup>2</sup>C slave)  
 1 = Release SCLx clock  
 0 = Hold SCLx clock low (clock stretch)  
If STREN = 1:  
 Bit is R/W (i.e., software can write '0' to initiate stretch and write '1' to release clock). Hardware clear at beginning of every slave data byte transmission. Hardware clear at end every of slave address byte reception. Hardware clear at every slave data byte reception.  
If STREN = 0:  
 Bit is R/S (i.e., software can only write '1' to release clock). Hardware clear at beginning of every slave data byte transmission. Hardware clear at end of every slave address byte reception.
- bit 11      **IPMIEN:** Intelligent Peripheral Management Interface (IPMI) Enable bit  
 1 = IPMI mode is enabled; all addresses Acknowledged  
 0 = IPMI mode disabled
- bit 10      **A10M:** 10-bit Slave Address bit  
 1 = I2CxADD is a 10-bit slave address  
 0 = I2CxADD is a 7-bit slave address
- bit 9        **DISSLW:** Disable Slew Rate Control bit  
 1 = Slew rate control disabled  
 0 = Slew rate control enabled
- bit 8        **SMEN:** SMBus Input Levels bit  
 1 = Enable I/O pin thresholds compliant with SMBus specification  
 0 = Disable SMBus input thresholds
- bit 7        **GCEN:** General Call Enable bit (when operating as I<sup>2</sup>C slave)  
 1 = Enable interrupt when a general call address is received in the I2CxRSR (module is enabled for reception)  
 0 = General call address disabled
- bit 6        **STREN:** SCLx Clock Stretch Enable bit (when operating as I<sup>2</sup>C slave)  
 Used in conjunction with SCLREL bit.  
 1 = Enable software or receive clock stretching  
 0 = Disable software or receive clock stretching

# PIC24FJ16MC101/102

---

---

## REGISTER 17-1: I2CxCON: I2Cx CONTROL REGISTER (CONTINUED)

- bit 5      **ACKDT:** Acknowledge Data bit (when operating as I<sup>2</sup>C master, applicable during master receive)  
Value that will be transmitted when the software initiates an Acknowledge sequence.  
1 = Send NACK during Acknowledge  
0 = Send ACK during Acknowledge
- bit 4      **ACKEN:** Acknowledge Sequence Enable bit  
(when operating as I<sup>2</sup>C master, applicable during master receive)  
1 = Initiate Acknowledge sequence on SDAx and SCLx pins and transmit ACKDT data bit.  
Hardware clear at end of master Acknowledge sequence.  
0 = Acknowledge sequence not in progress
- bit 3      **RCEN:** Receive Enable bit (when operating as I<sup>2</sup>C master)  
1 = Enables Receive mode for I<sup>2</sup>C. Hardware clear at end of eighth bit of master receive data byte.  
0 = Receive sequence not in progress
- bit 2      **PEN:** Stop Condition Enable bit (when operating as I<sup>2</sup>C master)  
1 = Initiate Stop condition on SDAx and SCLx pins. Hardware clear at end of master Stop sequence.  
0 = Stop condition not in progress
- bit 1      **RSEN:** Repeated Start Condition Enable bit (when operating as I<sup>2</sup>C master)  
1 = Initiate Repeated Start condition on SDAx and SCLx pins. Hardware clear at end of  
master Repeated Start sequence.  
0 = Repeated Start condition not in progress
- bit 0      **SEN:** Start Condition Enable bit (when operating as I<sup>2</sup>C master)  
1 = Initiate Start condition on SDAx and SCLx pins. Hardware clear at end of master Start sequence.  
0 = Start condition not in progress

## REGISTER 17-2: 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	

<b>Legend:</b>	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<sup>2</sup>C master, applicable to master transmit operation)  
1 = NACK received from slave  
0 = ACK received from slave  
Hardware set or clear at end of slave Acknowledge.
- bit 14      **TRSTAT:** Transmit Status bit (when operating as I<sup>2</sup>C master, applicable to master transmit operation)  
1 = Master transmit is in progress (8 bits + ACK)  
0 = Master transmit is not in progress  
Hardware set at beginning of master transmission. Hardware clear at end of slave Acknowledge.
- bit 13-11   **Unimplemented:** Read as '0'
- bit 10      **BCL:** Master Bus Collision Detect bit  
1 = A bus collision has been detected during a master operation  
0 = No collision  
Hardware set at detection of bus collision.
- bit 9        **GCSTAT:** General Call Status bit  
1 = General call address was received  
0 = General call address was not received  
Hardware set when address matches general call address. Hardware clear at Stop detection.
- bit 8        **ADD10:** 10-bit Address Status bit  
1 = 10-bit address was matched  
0 = 10-bit address was not matched  
Hardware set at match of 2nd byte of matched 10-bit address. Hardware clear at Stop detection.
- bit 7        **IWCOL:** Write Collision Detect bit  
1 = An attempt to write the I2CxTRN register failed because the I<sup>2</sup>C module is busy  
0 = No collision  
Hardware set at occurrence of write to I2CxTRN while busy (cleared by software).
- bit 6        **I2COV:** Receive Overflow Flag bit  
1 = A byte was received while the I2CxRCV register is still holding the previous byte  
0 = No overflow  
Hardware set at attempt to transfer I2CxRSR to I2CxRCV (cleared by software).
- bit 5        **D\_A:** Data/Address bit (when operating as I<sup>2</sup>C slave)  
1 = Indicates that the last byte received was data  
0 = Indicates that the last byte received was device address  
Hardware clear at device address match. Hardware set by reception of slave byte.
- bit 4        **P:** Stop bit  
1 = Indicates that a Stop bit has been detected last  
0 = Stop bit was not detected last  
Hardware set or clear when Start, Repeated Start or Stop detected.

# PIC24FJ16MC101/102

---

---

## REGISTER 17-2: I2CxSTAT: I2Cx STATUS REGISTER (CONTINUED)

- bit 3           **S:** Start bit  
1 = Indicates that a Start (or Repeated Start) bit has been detected last  
0 = Start bit was not detected last  
Hardware set or clear when Start, Repeated Start or Stop detected.
- bit 2           **R\_W:** Read/Write Information bit (when operating as I<sup>2</sup>C slave)  
1 = Read – indicates data transfer is output from slave  
0 = Write – indicates data transfer is input to slave  
Hardware set or clear after reception of I<sup>2</sup>C device address byte.
- bit 1           **RBF:** Receive Buffer Full Status bit  
1 = Receive complete, I2CxRCV is full  
0 = Receive not complete, I2CxRCV is empty  
Hardware set when I2CxRCV is written with received byte. Hardware clear when software reads I2CxRCV.
- bit 0           **TBF:** Transmit Buffer Full Status bit  
1 = Transmit in progress, I2CxTRN is full  
0 = Transmit complete, I2CxTRN is empty  
Hardware set when software writes I2CxTRN. Hardware clear at completion of data transmission.

## REGISTER 17-3: 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

# PIC24FJ16MC101/102

---

---

NOTES:

## 18.0 UNIVERSAL ASYNCHRONOUS RECEIVER TRANSMITTER (UART)

**Note 1:** This data sheet summarizes the features of the PIC24FJ16MC101/102 family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 21. "UART"** (DS39708) in the "PIC24F Family Reference Manual", which is available from the Microchip web site ([www.microchip.com](http://www.microchip.com)).

**2:** It is important to note that the specifications in **Section 26.0 "Electrical Characteristics"** of this data sheet, supercede any specifications that may be provided in PIC24F Family Reference Manual sections.

**3:** 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 PIC24FJ16MC101/102 device family. The UART is a full-duplex asynchronous system that can communicate with peripheral devices, such as personal computers, LIN 2.0, and RS-232, and RS-485 interfaces. The module also supports a hardware flow control option with the UxCTS and UxRTS pins and also includes an IrDA® encoder and decoder.

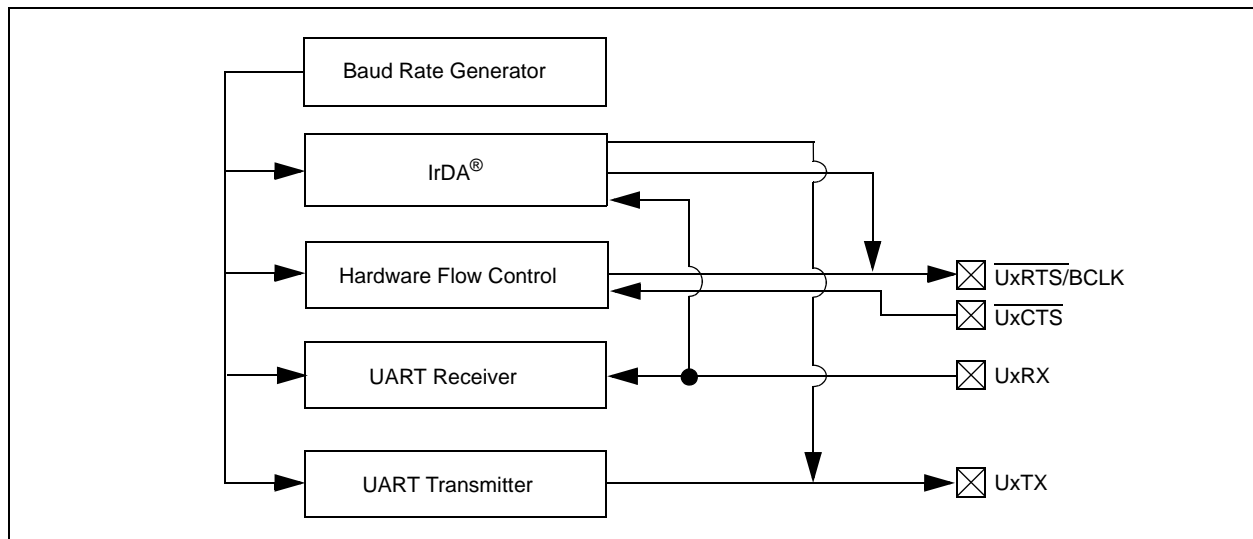
The primary features of the UART module are:

- Full-Duplex, 8-bit or 9-bit Data Transmission through the UxTX and UxRX pins
- Even, Odd, or No Parity Options (for 8-bit data)
- One or two stop bits
- Hardware flow control option with  $\overline{UxCTS}$  and  $\overline{UxRTS}$  pins
- Fully integrated Baud Rate Generator with 16-bit prescaler
- Baud rates ranging from 0.4 Mbps to 6 bps at 16x mode at 16 MIPS
- Baud rates ranging from 1.6 Mbps to 24.4 bps at 4x mode at 16 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 18-1**. The UART module consists of these key hardware elements:

- Baud Rate Generator
- Asynchronous Transmitter
- Asynchronous Receiver

**FIGURE 18-1: UART SIMPLIFIED BLOCK DIAGRAM**



# PIC24FJ16MC101/102

## REGISTER 18-1: UxMODE: UARTx MODE REGISTER

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

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

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

- bit 15      **UARTEN:** UARTx Enable bit<sup>(1)</sup>  
 1 = UARTx is enabled; all UARTx pins are controlled by UARTx as defined by UEN<1:0>  
 0 = UARTx is disabled; all UARTx pins are controlled by port latches; UARTx power consumption minimal
- bit 14      **Unimplemented:** Read as '0'
- bit 13      **USIDL:** Stop in Idle Mode bit  
 1 = Discontinue module operation when device enters Idle mode  
 0 = Continue module operation in Idle mode
- bit 12      **IREN:** IrDA<sup>®</sup> Encoder and Decoder Enable bit<sup>(2)</sup>  
 1 = IrDA encoder and decoder enabled  
 0 = IrDA encoder and decoder disabled
- bit 11      **RTSMD:** Mode Selection for UxRTS Pin bit  
 1 = 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 Pin Enable bits  
 11 = UxTX, UxRX and BCLK pins are enabled and used; UxCTS pin controlled by port latches  
 10 = UxTX, UxRX, UxCTS and UxRTS pins are enabled and used  
 01 = UxTX, UxRX and UxRTS pins are enabled and used; UxCTS pin controlled by port latches  
 00 = UxTX and UxRX pins are enabled and used; UxCTS and UxRTS/BCLK pins controlled by port latches
- bit 7      **WAKE:** Wake-up on Start bit Detect During Sleep Mode Enable bit  
 1 = UARTx will continue to sample the UxRX pin; interrupt generated on falling edge; bit cleared in hardware on following rising edge  
 0 = No wake-up enabled
- bit 6      **LPBACK:** UARTx Loopback Mode Select bit  
 1 = Enable Loopback mode  
 0 = Loopback mode is disabled
- bit 5      **ABAUD:** Auto-Baud Enable bit  
 1 = Enable baud rate measurement on the next character – requires reception of a Sync field (55h) before other data; cleared in hardware upon completion  
 0 = Baud rate measurement disabled or completed

- Note 1:** Refer to **Section 21. “UART”** (DS39708) in the *“PIC24F 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 18-1: UxMODE: UARTx MODE REGISTER (CONTINUED)

bit 4	<b>URXINV:</b> Receive Polarity Inversion bit 1 = UxRX Idle state is '0' 0 = UxRX Idle state is '1'
bit 3	<b>BRGH:</b> High Baud Rate Enable bit 1 = BRG generates 4 clocks per bit period (4x baud clock, High-Speed mode) 0 = BRG generates 16 clocks per bit period (16x baud clock, Standard mode)
bit 2-1	<b>PDSEL&lt;1:0&gt;:</b> Parity and Data Selection bits 11 = 9-bit data, no parity 10 = 8-bit data, odd parity 01 = 8-bit data, even parity 00 = 8-bit data, no parity
bit 0	<b>STSEL:</b> Stop Bit Selection bit 1 = Two Stop bits 0 = One Stop bit

- Note 1:** Refer to **Section 21. "UART"** (DS39708) in the *"PIC24F 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).

# PIC24FJ16MC101/102

## REGISTER 18-2: UxSTA: UARTx STATUS AND CONTROL REGISTER

R/W-0	R/W-0	R/W-0	U-0	R/W-0 HC	R/W-0	R-0	R-1
UTXISEL1	UTXINV	UTXISEL0	—	UTXBRK	UTXEN <sup>(1)</sup>	UTXBF	TRMT
bit 15						bit 8	

R/W-0	R/W-0	R/W-0	R-1	R-0	R-0	R/C-0	R-0
URXISEL<1:0>		ADDEN	RIDL	PERR	FERR	OERR	URXDA
bit 7						bit 0	

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

- bit 15,13      **UTXISEL<1:0>**: Transmission Interrupt Mode Selection bits  
 11 = Reserved; do not use  
 10 = Interrupt when a character is transferred to the Transmit Shift Register, and as a result, the transmit buffer becomes empty  
 01 = Interrupt when the last character is shifted out of the Transmit Shift Register; all transmit operations are completed  
 00 = Interrupt when a character is transferred to the Transmit Shift Register (this implies there is at least one character open in the transmit buffer)
- bit 14      **UTXINV**: Transmit Polarity Inversion bit  
If IREN = 0:  
 1 = UxTX Idle state is '0'  
 0 = UxTX Idle state is '1'  
If IREN = 1:  
 1 = IrDA<sup>®</sup> encoded UxTX Idle state is '1'  
 0 = IrDA encoded UxTX Idle state is '0'
- bit 12      **Unimplemented**: Read as '0'
- bit 11      **UTXBRK**: Transmit Break bit  
 1 = Send Sync Break on next transmission – Start bit, followed by twelve '0' bits, followed by Stop bit; cleared by hardware upon completion  
 0 = Sync Break transmission disabled or completed
- bit 10      **UTXEN**: Transmit Enable bit<sup>(1)</sup>  
 1 = Transmit enabled, UxTX pin controlled by UARTx  
 0 = Transmit disabled, any pending transmission is aborted and buffer is reset. UxTX pin controlled by port.
- bit 9      **UTXBF**: Transmit Buffer Full Status bit (read-only)  
 1 = Transmit buffer is full  
 0 = Transmit buffer is not full, at least one more character can be written
- bit 8      **TRMT**: Transmit Shift Register Empty bit (read-only)  
 1 = Transmit Shift Register is empty and transmit buffer is empty (the last transmission has completed)  
 0 = Transmit Shift Register is not empty, a transmission is in progress or queued
- bit 7-6      **URXISEL<1:0>**: Receive Interrupt Mode Selection bits  
 11 = Interrupt is set on UxRSR transfer making the receive buffer full (i.e., has 4 data characters)  
 10 = Interrupt is set on UxRSR transfer making the receive buffer 3/4 full (i.e., has 3 data characters)  
 0x = Interrupt is set when any character is received and transferred from the UxRSR to the receive buffer. Receive buffer has one or more characters.

**Note 1:** Refer to **Section 21. "UART"** (DS39708) in the *"PIC24F Family Reference Manual"* for information on enabling the UART module for transmit operation.

## REGISTER 18-2: 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-only/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 21. “UART”** (DS39708) in the *“PIC24F Family Reference Manual”* for information on enabling the UART module for transmit operation.

# PIC24FJ16MC101/102

---

---

NOTES:

## 19.0 10-BIT ANALOG-TO-DIGITAL CONVERTER (ADC)

**Note 1:** This data sheet summarizes the features of the PIC24FJ16MC101/102 family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 46. “10-bit Analog-to-Digital Converter (ADC) with 4 Simultaneous Conversions”** (DS39737) in the *“PIC24F Family Reference Manual”*, which is available from the Microchip web site ([www.microchip.com](http://www.microchip.com)).

**2:** It is important to note that the specifications in **Section 26.0 “Electrical Characteristics”** of this data sheet, supercede any specifications that may be provided in PIC24F Family Reference Manual sections.

**3:** 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 PIC24FJ16MC101/102 devices have up to six ADC module input channels.

### 19.1 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 six analog input pins
- Four Sample and Hold circuits for 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

Depending on the particular device pinout, the ADC can have up to six analog input pins, designated AN0 through AN5.

Block diagrams of the ADC module are shown in [Figure 19-1](#) and [Figure 19-2](#).

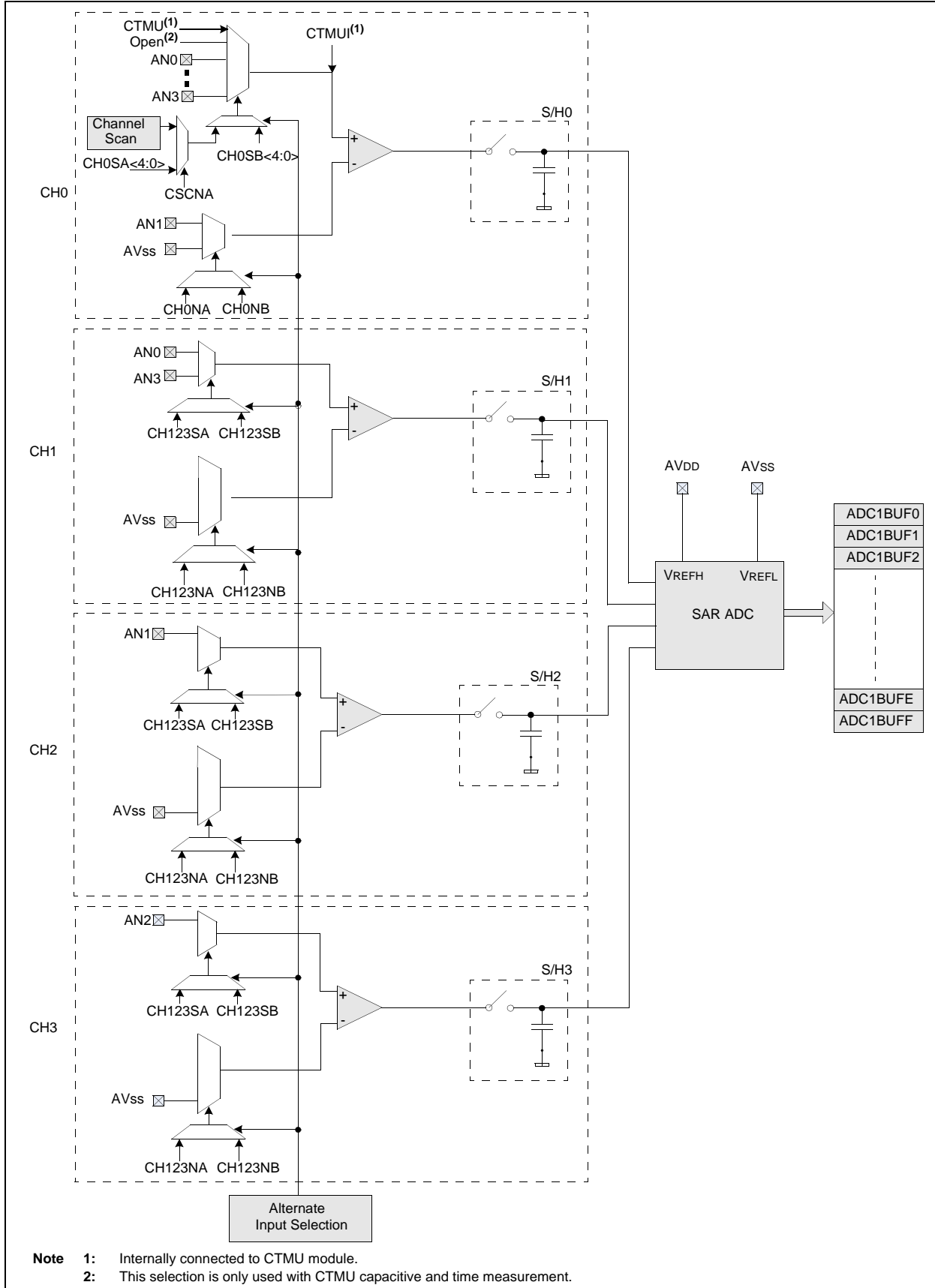
## 19.2 ADC Initialization

To configure the ADC module:

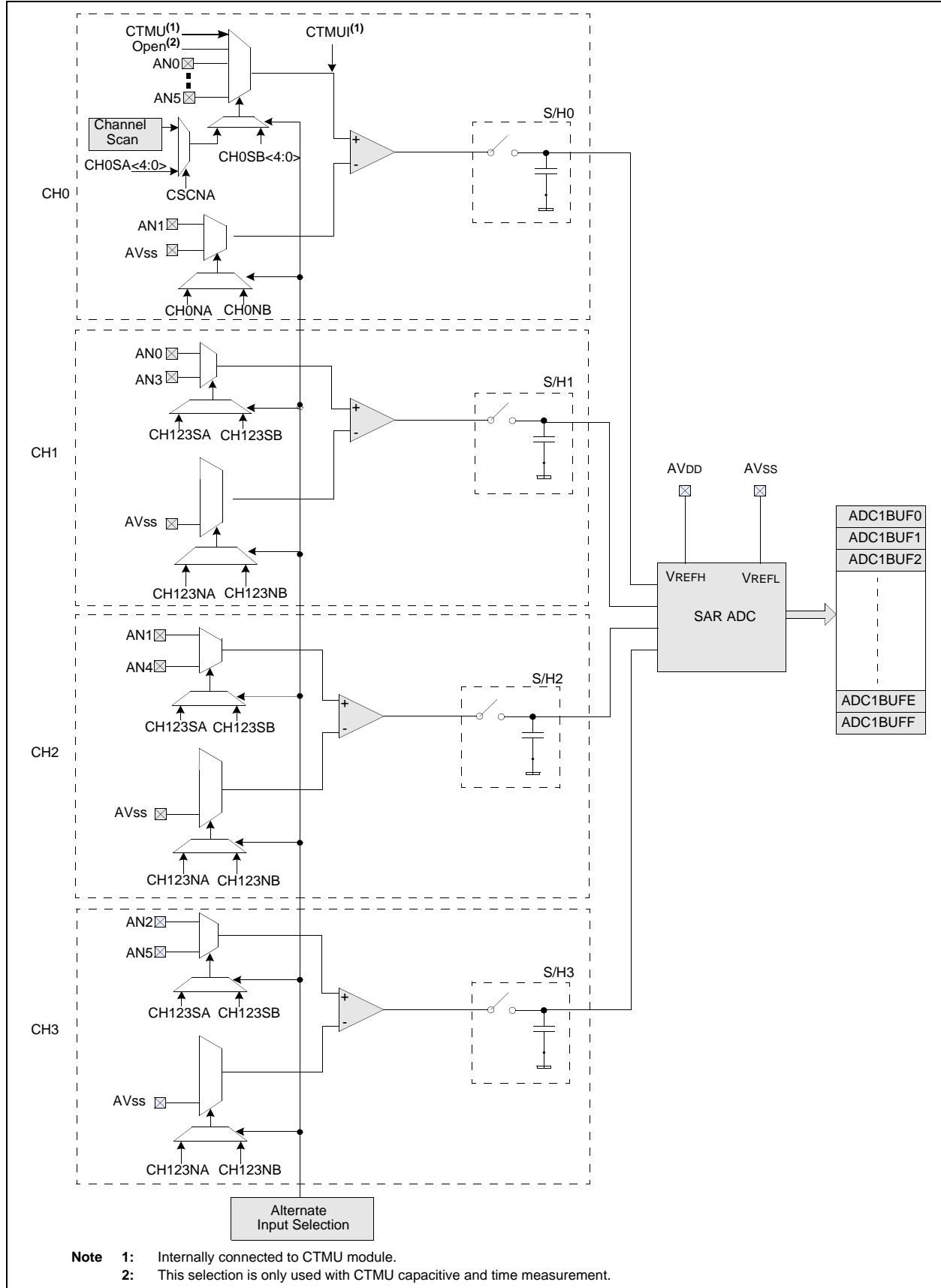
1. Select port pins as analog inputs (ADxPCFGH<15:0> or ADxPCFGL<15:0>).
2. Select voltage reference source to match expected range on analog inputs (ADxCON2<15:13>).
3. Select the analog conversion clock to match the desired data rate with the processor clock (ADxCON3<7:0>).
4. Determine how many sample-and-hold channels will be used (ADxCON2<9:8> and ADxPCFGH<15:0> or ADxPCFGL<15:0>).
5. Select the appropriate sample/conversion sequence (ADxCON1<7:5> and ADxCON3<12:8>).
6. Select the way conversion results are presented in the buffer (ADxCON1<9:8>).
7. Turn on the ADC module (ADxCON1<15>).
8. Configure ADC interrupt (if required):
  - a) Clear the ADxIF bit.
  - b) Select the ADC interrupt priority.

# PIC24FJ16MC101/102

**FIGURE 19-1: ADC1 BLOCK DIAGRAM FOR PIC24FJ16MC101 DEVICES**

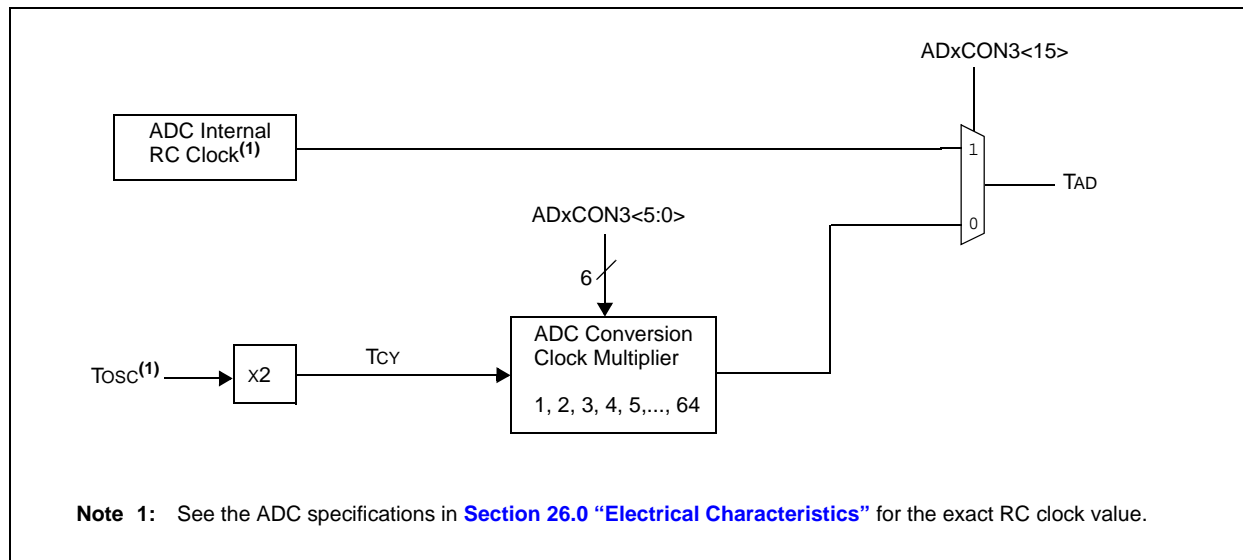


**FIGURE 19-2: ADC1 BLOCK DIAGRAM FOR PIC24FJ16MC102 DEVICES**



# PIC24FJ16MC101/102

FIGURE 19-3: ADC CONVERSION CLOCK PERIOD BLOCK DIAGRAM





## REGISTER 19-1: AD1CON1: ADC1 CONTROL REGISTER 1

R/W-0	U-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0
ADON	—	ADSIDL	—	—	—	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	R/C-0
SSRC<2:0>			—	SIMSAM	ASAM	SAMP	DONE
bit 7						bit 0	

<b>Legend:</b>	HC = Cleared by hardware	HS = Set by hardware	C = Clearable bit
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

- bit 15      **ADON:** ADC Operating Mode bit  
 1 = ADC module is operating  
 0 = ADC is off
- bit 14      **Unimplemented:** Read as '0'
- bit 13      **ADSIDL:** Stop in Idle Mode bit  
 1 = Discontinue module operation when device enters Idle mode  
 0 = Continue module operation in Idle mode
- bit 12-10   **Unimplemented:** Read as '0'
- bit 9-8      **FORM<1:0>:** Data Output Format bits  
 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)
- bit 7-5      **SSRC<2:0>:** Sample Clock Source Select bits  
 111 = Internal counter ends sampling and starts conversion (auto-convert)  
 110 = CTMU  
 101 = Reserved  
 100 = Reserved  
 011 = Motor Control PWM 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)  
 1 = Samples CH0, CH1, CH2, CH3 simultaneously (when CHPS<1:0> = 1x); or  
      Samples CH0 and CH1 simultaneously (when CHPS<1:0> = 01)  
 0 = Samples multiple channels individually in sequence
- bit 2      **ASAM:** ADC Sample Auto-Start bit  
 1 = Sampling begins immediately after last conversion. SAMP bit is auto-set.  
 0 = Sampling begins when SAMP bit is set
- bit 1      **SAMP:** ADC Sample Enable bit  
 1 = ADC sample-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.

# PIC24FJ16MC101/102

---

---

## REGISTER 19-1: AD1CON1: ADC1 CONTROL REGISTER 1 (CONTINUED)

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 19-2: 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

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

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

	ADREF+	ADREF-
xxx	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

- 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 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

# PIC24FJ16MC101/102

## REGISTER 19-3: 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> <sup>(1)</sup>				
bit 15							bit 8

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

### Legend:

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

bit 15            **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<sup>(1)</sup>  
                   11111 = 31 TAD  
                   •  
                   •  
                   •  
                   00001 = 1 TAD  
                   00000 = 0 TAD

bit 7-0         **ADCS<7:0>:** ADC Conversion Clock Select bits<sup>(2)</sup>  
                   11111111 = Reserved  
                   •  
                   •  
                   •  
                   •  
                   01000000 = Reserved  
                   00111111 =  $T_{CY} \cdot (ADCS<7:0> + 1) = 64 \cdot T_{CY} = TAD$   
                   •  
                   •  
                   •  
                   00000010 =  $T_{CY} \cdot (ADCS<7:0> + 1) = 3 \cdot T_{CY} = TAD$   
                   00000001 =  $T_{CY} \cdot (ADCS<7:0> + 1) = 2 \cdot T_{CY} = TAD$   
                   00000000 =  $T_{CY} \cdot (ADCS<7:0> + 1) = 1 \cdot T_{CY} = TAD$

**Note 1:** This bit only used if AD1CON1<7:5> (SSRC<2:0>) = 1.  
**Note 2:** This bit is not used if AD1CON3<15> (ADRC) = 1.

## REGISTER 19-4: 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  
 11 = Reserved  
 10 = Reserved  
 0x = CH1, CH2, CH3 negative input is AVss
- bit 8          **CH123SB:** Channel 1, 2, 3 Positive Input Select for Sample B bit  
**PIC24FJ16MC101 devices only:**  
 1 = CH1 positive input is AN3, CH2 and CH3 positive inputs are not connected  
 0 = CH1 positive input is AN0, CH2 positive input is AN1, CH3 positive input is AN2  
**PIC24FJ16MC102 devices only:**  
 1 = CH1 positive input is AN3, CH2 positive input is AN4, CH3 positive input is AN5  
 0 = CH1 positive input is AN0, CH2 positive input is AN1, CH3 positive input is AN2
- bit 7-3      **Unimplemented:** Read as '0'
- bit 2-1      **CH123NA<1:0>:** Channel 1, 2, 3 Negative Input Select for Sample A bits  
 11 = Reserved  
 10 = Reserved  
 0x = CH1, CH2, CH3 negative input is AVss
- bit 0          **CH123SA:** Channel 1, 2, 3 Positive Input Select for Sample A bit  
**PIC24FJ16MC101 devices only:**  
 1 = CH1 positive input is AN3, CH2 and CH3 positive inputs are not connected  
 0 = CH1 positive input is AN0, CH2 positive input is AN1, CH3 positive input is AN2  
**PIC24FJ16MC102 devices only:**  
 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

# PIC24FJ16MC101/102

**REGISTER 19-5: 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> <sup>(1)</sup>				
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> <sup>(1)</sup>				
bit 7							bit 0

**Legend:**

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

- bit 15            **CH0NB:** Channel 0 Negative Input Select for Sample B bit  
                   1 = Channel 0 negative input is AN1  
                   0 = Channel 0 negative input is AVss
- bit 14-13       **Unimplemented:** Read as '0'
- bit 12-8        **CH0SB<4:0>:** Channel 0 Positive Input Select for Sample B bits<sup>(1)</sup>  
                   **PIC24FJ16MC101 devices only:**  
                   01110 = No channels connected; all inputs are floating (used for CTMU)  
                   01101 = Channel 0 positive input is connected to CTMU temperature sensor  
                   00011 = Channel 0 positive input is AN3  
                   00010 = Channel 0 positive input is AN2  
                   00001 = Channel 0 positive input is AN1  
                   00000 = Channel 0 positive input is AN0  
                   **PIC24FJ16MC102 devices only:**  
                   01110 = No channels connected; all inputs are floating (used for CTMU)  
                   01101 = Channel 0 positive input is connected to CTMU temperature sensor  
                   00101 = Channel 0 positive input is AN5  
                   00100 = Channel 0 positive input is AN4  
                   00011 = Channel 0 positive input is AN3  
                   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 AVss
- bit 6-5        **Unimplemented:** Read as '0'
- bit 4-0        **CH0SA<4:0>:** Channel 0 Positive Input Select for Sample A bits<sup>(1)</sup>  
                   **PIC24FJ16MC101 devices only:**  
                   01110 = No channels connected; all inputs are floating (used for CTMU)  
                   01101 = Channel 0 positive input is connected to CTMU temperature sensor  
                   00011 = Channel 0 positive input is AN3  
                   00010 = Channel 0 positive input is AN2  
                   00001 = Channel 0 positive input is AN1  
                   00000 = Channel 0 positive input is AN0  
                   **PIC24FJ16MC102 devices only:**  
                   01110 = No channels connected; all inputs are floating (used for CTMU)  
                   01101 = Channel 0 positive input is connected to CTMU temperature sensor  
                   00101 = Channel 0 positive input is AN5  
                   00100 = Channel 0 positive input is AN4  
                   00011 = Channel 0 positive input is AN3  
                   00010 = Channel 0 positive input is AN2  
                   00001 = Channel 0 positive input is AN1  
                   00000 = Channel 0 positive input is AN0

**Note 1:** All other values than those listed are Reserved.

## REGISTER 19-6: AD1CSSL: ADC1 INPUT SCAN SELECT REGISTER LOW<sup>(1,2,3)</sup>

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

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	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 15-6      **Unimplemented:** Read as '0'

bit 5-0      **CSS<5:0>:** ADC Input Scan Selection bits

1 = Select ANx for input scan

0 = Skip ANx for input scan

**Note 1:** On devices without 6 analog inputs, all AD1CSSL bits can be selected by user application. However, inputs selected for scan without a corresponding input on device converts VREFL.

**2:** CSSx = ANx, where x = 0 through 5.

**3:** CTMU temperature sensor input cannot be scanned.

# PIC24FJ16MC101/102

## REGISTER 19-7: AD1PCFGL: ADC1 PORT CONFIGURATION REGISTER LOW<sup>(1,2,3)</sup>

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

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	PCFG5 <sup>(4)</sup>	PCFG4 <sup>(4)</sup>	PCFG3 <sup>(4)</sup>	PCFG2 <sup>(4)</sup>	PCFG1 <sup>(4)</sup>	PCFG0 <sup>(4)</sup>
bit 7							bit 0

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 15-6 **Unimplemented:** Read as '0'

bit 5-0 **PCFG<5:0>:** ADC Port Configuration Control bits<sup>(4)</sup>

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 6 analog inputs, all PCFG bits are R/W by user. However, PCFG bits are ignored on ports without a corresponding input on device.
- 2:** PCFGx = ANx, where x = 0 through 5.
- 3:** PCFGx bits have no effect if the ADC module is disabled by setting ADxMD bit in the PMDx register. When the bit is set, all port pins that have been multiplexed with ANx will be in Digital mode.
- 4:** Pins shared with analog functions (i.e., ANx), are analog by default and therefore, must be set by the user to enable any digital function on that pin. Reading any port pin with the analog function enabled will return a '0', regardless of the signal input level.



## 20.0 COMPARATOR MODULE

**Note 1:** This data sheet summarizes the features of the PIC24FJ16MC101/102 families of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 48. “Comparator with Blanking”** (DS39741) of the “PIC24F Family Reference Manual”, which is available from the Microchip website ([www.microchip.com](http://www.microchip.com)).

**2:** It is important to note that the specifications in **Section 26.0 “Electrical Characteristics”** of this data sheet, supercede any specifications that may be provided in PIC24F Family Reference Manual sections.

**3:** 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 PIC24FJ16MC101/102 Comparator module provides three comparators that can be configured in different ways. As shown in [Figure 20-1](#), individual comparator options are specified by the Comparator module’s Special Function Register (SFR) control bits.

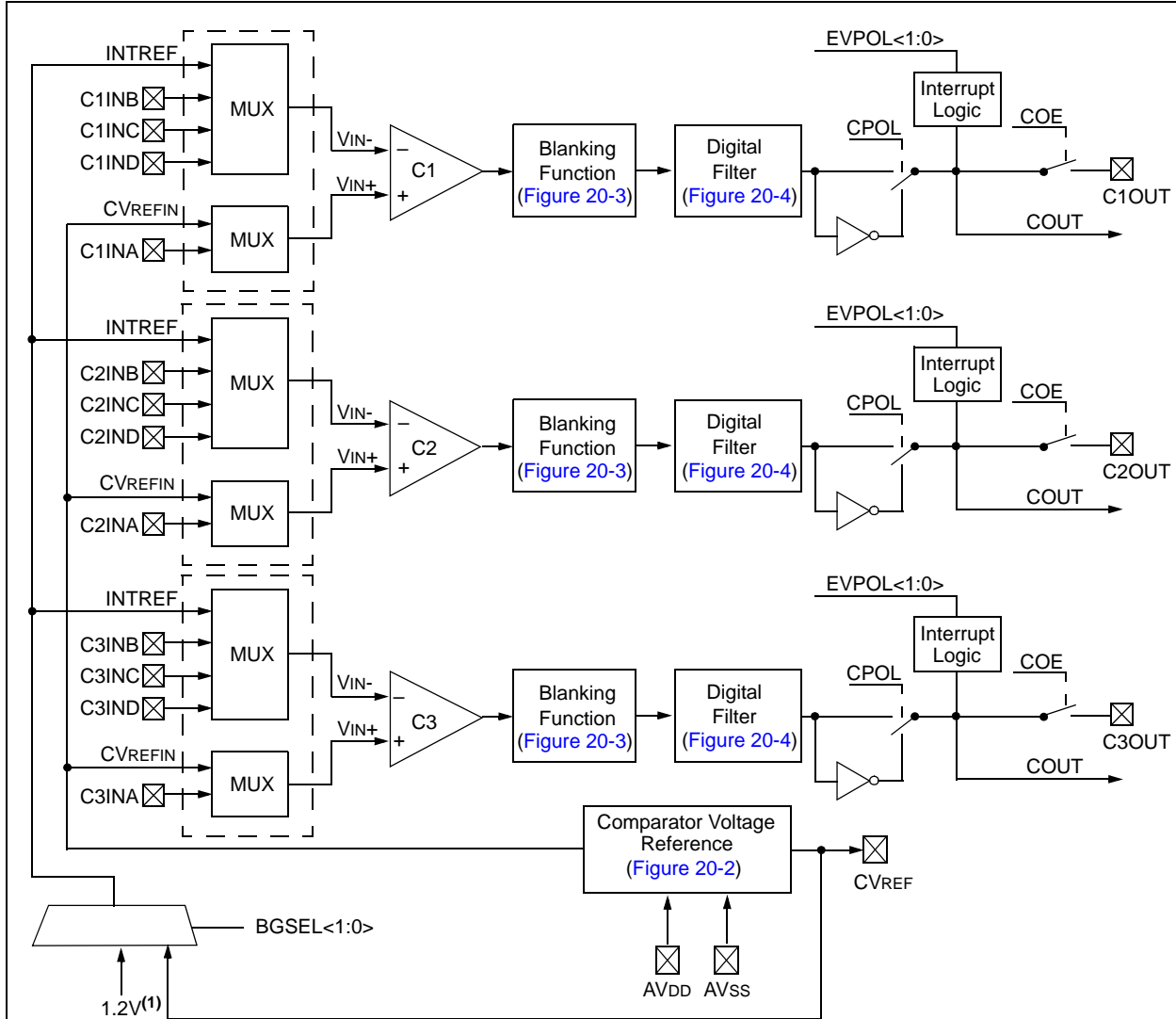
These options allow users to:

- Select the edge for trigger and interrupt generation
- Select low-power control
- Configure the comparator voltage reference and band gap
- Configure output blanking and masking

The comparator operating mode is determined by the input selections (i.e., whether the input voltage is compared to a second input voltage, to an internal voltage reference).

# PIC24FJ16MC101/102

**FIGURE 20-1: COMPARATOR I/O OPERATING MODES**



**Note 1:** This reference voltage is generated internally on the device. Refer to [Section 26.0 "Electrical Characteristics"](#) for the specified voltage range.

FIGURE 20-2: COMPARATOR VOLTAGE REFERENCE BLOCK DIAGRAM

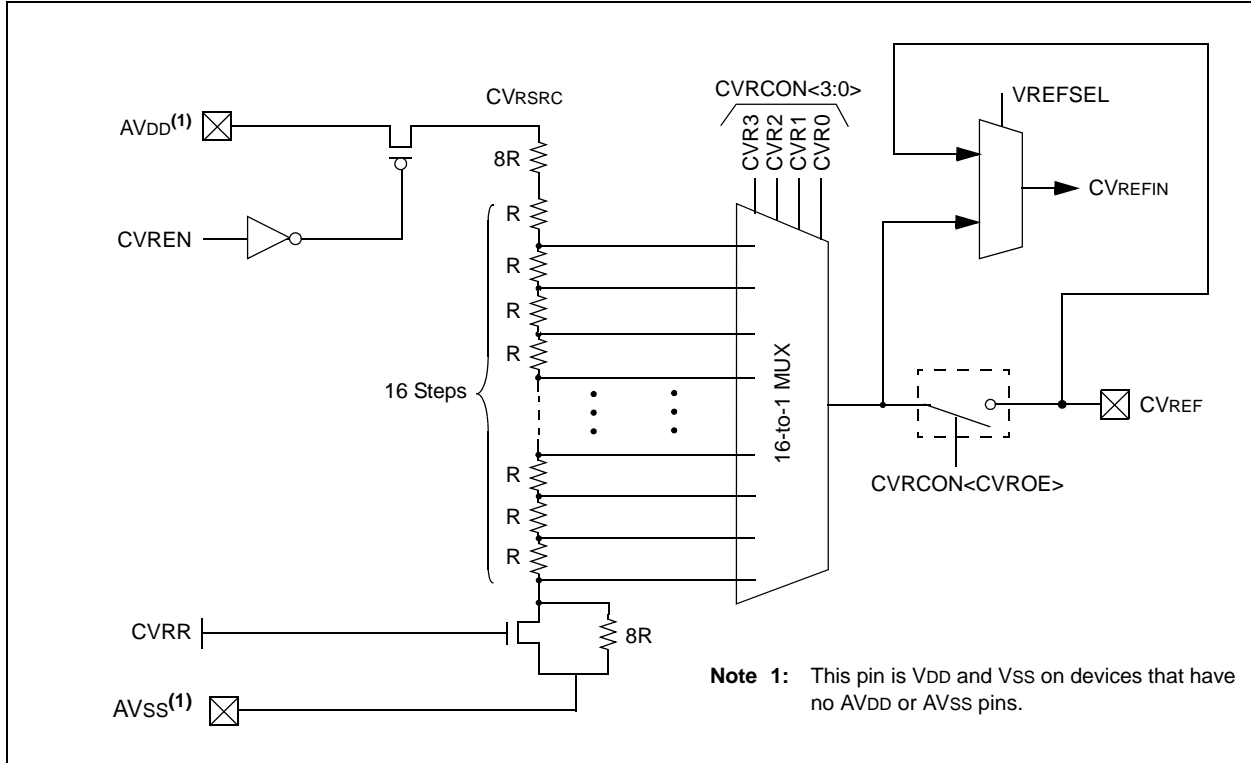
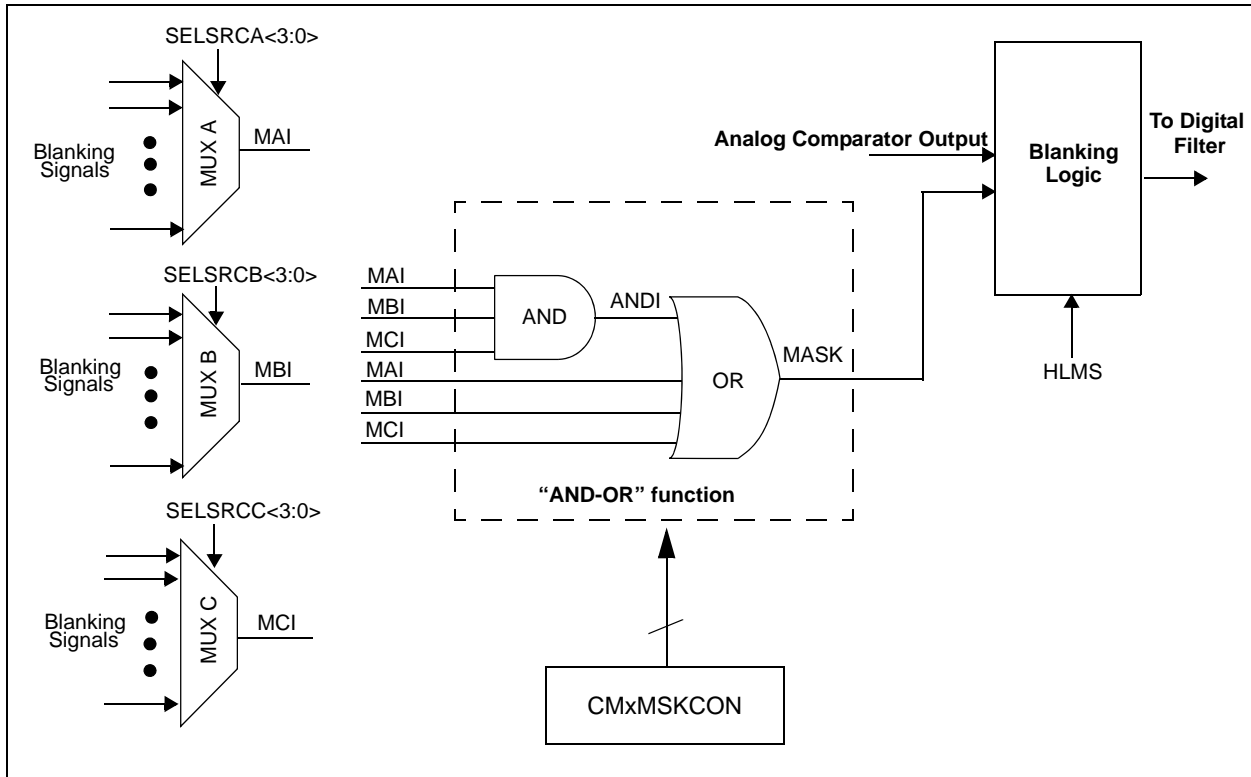
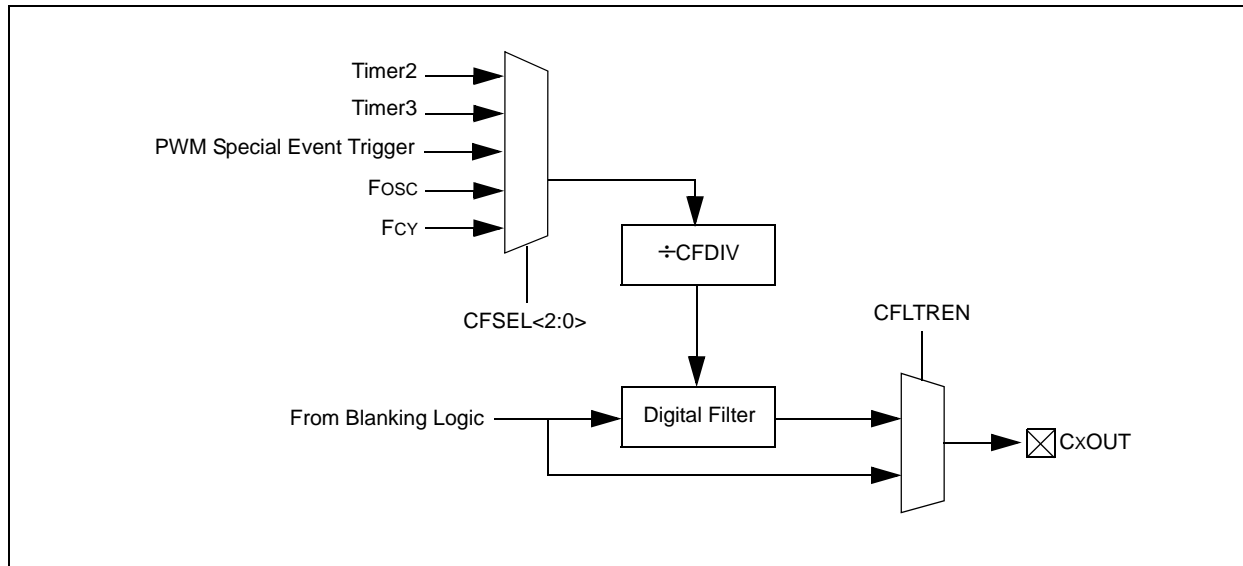


FIGURE 20-3: USER PROGRAMMABLE BLANKING FUNCTION BLOCK DIAGRAM



# PIC24FJ16MC101/102

FIGURE 20-4: DIGITAL FILTER INTERCONNECT BLOCK DIAGRAM



## REGISTER 20-1: CMSTAT: COMPARATOR STATUS REGISTER

R/W-0	U-0	U-0	U-0	U-0	R-0	R-0	R-0
CMSIDL	—	—	—	—	C3EVT	C2EVT	C1EVT
bit 15						bit 8	

U-0	U-0	U-0	U-0	U-0	R-0	R-0	R-0
—	—	—	—	—	C3OUT	C2OUT	C1OUT
bit 7						bit 0	

### Legend:

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

- bit 15            **CMSIDL:** Stop in Idle Mode bit  
                   1 = Discontinue operation of all comparators when device enters Idle mode  
                   0 = Continue operation of all comparators in Idle mode
- bit 14-11       **Unimplemented:** Read as '0'
- bit 10           **C3EVT:** Comparator 3 Event Status bit  
                   1 = Comparator event occurred  
                   0 = Comparator event did not occur
- bit 9            **C2EVT:** Comparator 2 Event Status bit  
                   1 = Comparator event occurred  
                   0 = Comparator event did not occur
- bit 8            **C1EVT:** Comparator 1 Event Status bit  
                   1 = Comparator event occurred  
                   0 = Comparator event did not occur
- bit 7-3         **Unimplemented:** Read as '0'
- bit 2            **C3OUT:** Comparator 3 Output Status bit  
                   When CPOL = 0:  
                   1 =  $V_{IN+} > V_{IN-}$   
                   0 =  $V_{IN+} < V_{IN-}$   
                   When CPOL = 1:  
                   1 =  $V_{IN+} < V_{IN-}$   
                   0 =  $V_{IN+} > V_{IN-}$
- bit 1            **C2OUT:** Comparator 2 Output Status bit  
                   When CPOL = 0:  
                   1 =  $V_{IN+} > V_{IN-}$   
                   0 =  $V_{IN+} < V_{IN-}$   
                   When CPOL = 1:  
                   1 =  $V_{IN+} < V_{IN-}$   
                   0 =  $V_{IN+} > V_{IN-}$
- bit 0            **C1OUT:** Comparator 1 Output Status bit  
                   When CPOL = 0:  
                   1 =  $V_{IN+} > V_{IN-}$   
                   0 =  $V_{IN+} < V_{IN-}$   
                   When CPOL = 1:  
                   1 =  $V_{IN+} < V_{IN-}$   
                   0 =  $V_{IN+} > V_{IN-}$

# PIC24FJ16MC101/102

## REGISTER 20-2: CMxCON: COMPARATOR CONTROL REGISTER

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0
CON	COE	CPOL	—	—	—	CEVT	COUT
bit 15						bit 8	

R/W-0	R/W-0	U-0	R/W-0	U-0	U-0	R/W-0	R/W-0
EVPOL<1:0>		—	CREF	—	—	CCH<1:0>	
bit 7						bit 0	

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

- bit 15      **CON:** Comparator Enable bit  
 1 = Comparator is enabled  
 0 = Comparator is disabled
- bit 14      **COE:** Comparator Output Enable bit  
 1 = Comparator output is present on the CxOUT pin  
 0 = Comparator output is internal only
- bit 13      **CPOL:** Comparator Output Polarity Select bit  
 1 = Comparator output is inverted  
 0 = Comparator output is not inverted
- bit 12-10   **Unimplemented:** Read as '0'
- bit 9        **CEVT:** Comparator Event bit  
 1 = Comparator event according to EVPOL<1:0> settings occurred; disables future triggers and interrupts until the bit is cleared  
 0 = Comparator event did not occur
- bit 8        **COUT:** Comparator Output bit  
When CPOL = 0 (non-inverted polarity):  
 1 = VIN+ > VIN-  
 0 = VIN+ < VIN-  
When CPOL = 1 (inverted polarity):  
 1 = VIN+ < VIN-  
 0 = VIN+ > VIN-
- bit 7-6     **EVPOL<1:0>:** Trigger/Event/Interrupt Polarity Select bits  
 11 = Trigger/Event/Interrupt generated on any change of the comparator output (while CEVT = 0)  
 10 = Trigger/Event/Interrupt generated only on high to low transition of the polarity-selected comparator output (while CEVT = 0)  
     If CPOL = 1 (inverted polarity):  
     Low-to-high transition of the comparator output  
     If CPOL = 0 (non-inverted polarity):  
     High-to-low transition of the comparator output  
 01 = Trigger/Event/Interrupt generated only on low to high transition of the polarity-selected comparator output (while CEVT = 0)  
     If CPOL = 1 (inverted polarity):  
     High-to-low transition of the comparator output  
     If CPOL = 0 (non-inverted polarity):  
     Low-to-high transition of the comparator output  
 00 = Trigger/Event/Interrupt generation is disabled

## REGISTER 20-2: CMxCON: COMPARATOR CONTROL REGISTER (CONTINUED)

- bit 5           **Unimplemented:** Read as '0'
- bit 4           **CREF:** Comparator Reference Select bit (VIN+ input)  
1 = VIN+ input connects to internal CVREFIN voltage  
0 = VIN+ input connects to CxINA pin
- bit 3-2       **Unimplemented:** Read as '0'
- bit 1-0       **CCH<1:0>:** Comparator Channel Select bits  
11 = VIN- input of comparator connects to INTREF  
10 = VIN- input of comparator connects to CxIND pin  
01 = VIN- input of comparator connects to CxINC pin  
00 = VIN- input of comparator connects to CxINB pin

# PIC24FJ16MC101/102

## REGISTER 20-3: CMxMSKSRC: COMPARATOR MASK SOURCE SELECT CONTROL REGISTER

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	RW-0
—	—	—	—	SELSRCC<3:0>			
bit 15				bit 8			

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SELSRCB<3:0>				SELSRCA<3:0>			
bit 7				bit 0			

### Legend:

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

- bit 15-12            **Unimplemented:** Read as '0'
- bit 11-8            **SELSRCC<3:0>:** Mask C Input Select bits
  - 1111 = Reserved
  - 1110 = Reserved
  - 1101 = Reserved
  - 1100 = Reserved
  - 1011 = Reserved
  - 1010 = Reserved
  - 1001 = Reserved
  - 1000 = Reserved
  - 0111 = Reserved
  - 0110 = Reserved
  - 0101 = PWM1H3
  - 0100 = PWM1L3
  - 0011 = PWM1H2
  - 0010 = PWM1L2
  - 0001 = PWM1H1
  - 0000 = PWM1L1
- bit 7-4             **SELSRCB<3:0>:** Mask B Input Select bits
  - 1111 = Reserved
  - 1110 = Reserved
  - 1101 = Reserved
  - 1100 = Reserved
  - 1011 = Reserved
  - 1010 = Reserved
  - 1001 = Reserved
  - 1000 = Reserved
  - 0111 = Reserved
  - 0110 = Reserved
  - 0101 = PWM1H3
  - 0100 = PWM1L3
  - 0011 = PWM1H2
  - 0010 = PWM1L2
  - 0001 = PWM1H1
  - 0000 = PWM1L1



## REGISTER 20-3: CMxMSKSRCA: COMPARATOR MASK SOURCE SELECT CONTROL REGISTER

bit 3-0      **SELSRCA<3:0>**: Mask A Input Select bits

1111 = Reserved  
1110 = Reserved  
1101 = Reserved  
1100 = Reserved  
1011 = Reserved  
1010 = Reserved  
1001 = Reserved  
1000 = Reserved  
0111 = Reserved  
0110 = Reserved  
0101 = PWM1H3  
0100 = PWM1L3  
0011 = PWM1H2  
0010 = PWM1L2  
0001 = PWM1H1  
0000 = PWM1L1

# PIC24FJ16MC101/102

## REGISTER 20-4: CMxMSKCON: COMPARATOR MASK GATING CONTROL REGISTER

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
HLMS	—	OCEN	OCNEN	OBEN	OBNEN	OAEN	OANEN
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
NAGS	PAGS	ACEN	ACNEN	ABEN	ABNEN	AAEN	AANEN
bit 7						bit 0	

### Legend:

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

- bit 15            **HLMS:** High or Low Level Masking Select bits  
                   1 = The masking (blinking) function will prevent any asserted ('0') comparator signal from propagating  
                   0 = The masking (blinking) function will prevent any asserted ('1') comparator signal from propagating
- bit 14            **Unimplemented:** Read as '0'
- bit 13            **OCEN:** OR Gate C Input Inverted Enable bit  
                   1 = MCI is connected to OR gate  
                   0 = MCI is not connected to OR gate
- bit 12            **OCNEN:** OR Gate C Input Inverted Enable bit  
                   1 = Inverted MCI is connected to OR gate  
                   0 = Inverted MCI is not connected to OR gate
- bit 11            **OBEN:** OR Gate B Input Inverted Enable bit  
                   1 = MBI is connected to OR gate  
                   0 = MBI is not connected to OR gate
- bit 10            **OBNEN:** OR Gate B Input Inverted Enable bit  
                   1 = Inverted MBI is connected to OR gate  
                   0 = Inverted MBI is not connected to OR gate
- bit 9             **OAEN:** OR Gate A Input Enable bit  
                   1 = MAI is connected to OR gate  
                   0 = MAI is not connected to OR gate
- bit 8             **OANEN:** OR Gate A Input Inverted Enable bit  
                   1 = Inverted MAI is connected to OR gate  
                   0 = Inverted MAI is not connected to OR gate
- bit 7             **NAGS:** Negative AND Gate Output Select  
                   1 = Inverted ANDI is connected to OR gate  
                   0 = Inverted ANDI is not connected to OR gate
- bit 6             **PAGS:** Positive AND Gate Output Select  
                   1 = ANDI is connected to OR gate  
                   0 = ANDI is not connected to OR gate
- bit 5             **ACEN:** AND Gate A1 C Input Inverted Enable bit  
                   1 = MCI is connected to AND gate  
                   0 = MCI is not connected to AND gate
- bit 4             **ACNEN:** AND Gate A1 C Input Inverted Enable bit  
                   1 = Inverted MCI is connected to AND gate  
                   0 = Inverted MCI is not connected to AND gate
- bit 3             **ABEN:** AND Gate A1 B Input Inverted Enable bit  
                   1 = MBI is connected to AND gate  
                   0 = MBI is not connected to AND gate

## REGISTER 20-4: CMxMSKCON: COMPARATOR MASK GATING CONTROL REGISTER

- bit 2      **ABNEN:** AND Gate A1 B Input Inverted Enable bit  
1 = Inverted MBI is connected to AND gate  
0 = Inverted MBI is not connected to AND gate
- bit 1      **AAEN:** AND Gate A1 A Input Enable bit  
1 = MAI is connected to AND gate  
0 = MAI is not connected to AND gate
- bit 0      **AAEN:** AND Gate A1 A Input Inverted Enable bit  
1 = Inverted MAI is connected to AND gate  
0 = Inverted MAI is not connected to AND gate

# PIC24FJ16MC101/102

## REGISTER 20-5: CMxFLTR: COMPARATOR FILTER CONTROL REGISTER

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

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	CFSEL<2:0>			CFLTREN	CFDIV<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

**CFSEL<2:0>:** Comparator Filter Input Clock Select bits

111 = Reserved

110 = Reserved

101 = Timer3

100 = Timer2

011 = Reserved

010 = PWM Special Event Trigger

001 = FOSC

000 = FCY

bit 3

**CFLTREN:** Comparator Filter Enable bit

1 = Digital filter enabled

0 = Digital filter disabled

bit 2-0

**CFDIV<2:0>:** Comparator Filter Clock Divide Select bits

111 = Clock Divide 1:128

110 = Clock Divide 1:64

101 = Clock Divide 1:32

100 = Clock Divide 1:16

011 = Clock Divide 1:8

010 = Clock Divide 1:4

001 = Clock Divide 1:2

000 = Clock Divide 1:1

## REGISTER 20-6: CVRCON: COMPARATOR VOLTAGE REFERENCE CONTROL REGISTER

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	VREFSEL	BGSEL<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	R/W-0
CVREN	CVROE <sup>(1)</sup>	CVRR	—	CVR<3:0>			
bit 7					bit 0		

### Legend:

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

- bit 15-11      **Unimplemented:** Read as '0'
- bit 10      **VREFSEL:** Voltage Reference Select bit  
             1 = CVREFIN = CVREF pin  
             0 = CVREFIN is generated by the resistor network
- bit 9-8      **BGSEL<1:0>:** Band Gap Reference Source Select bits  
             11 = INTREF = CVREF pin  
             10 = INTREF = 1.2V (nominal)<sup>(2)</sup>  
             0x = Reserved
- bit 7      **CVREN:** Comparator Voltage Reference Enable bit  
             1 = Comparator voltage reference circuit powered on  
             0 = Comparator voltage reference circuit powered down
- bit 6      **CVROE:** Comparator Voltage Reference Output Enable bit<sup>(1)</sup>  
             1 = Voltage level is output on CVREF pin  
             0 = Voltage level is disconnected from CVREF pin
- bit 5      **CVRR:** Comparator Voltage Reference Range Selection bit  
             1 = CVRSRC/24 step size  
             0 = CVRSRC/32 step size
- bit 4      **Unimplemented:** Read as '0'
- bit 3-0      **CVR<3:0>:** Comparator Voltage Reference Value Selection  $0 \leq \text{CVR}<3:0> \leq 15$  bits  
             When CVRR = 1:  
              $\text{CVREFIN} = (\text{CVR}<3:0>/24) \cdot (\text{CVRSRC})$   
             When CVRR = 0:  
              $\text{CVREFIN} = 1/4 \cdot (\text{CVRSRC}) + (\text{CVR}<3:0>/32) \cdot (\text{CVRSRC})$

- Note 1:** CVROE overrides the TRIS bit setting.  
**Note 2:** This reference voltage is generated internally on the device. Refer to [Section 26.0 “Electrical Characteristics”](#) for the specified voltage range.

# PIC24FJ16MC101/102

---

---

NOTES:

## 21.0 REAL-TIME CLOCK AND CALENDAR (RTCC)

**Note 1:** This data sheet summarizes the features of the PIC24FJ16MC101/102 family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 29. “Real-Time Clock and Calendar (RTCC)”** (DS39696) in the “PIC24F Family Reference Manual”, which is available on the Microchip web site ([www.microchip.com](http://www.microchip.com)).

**2:** It is important to note that the specifications in **Section 26.0 “Electrical Characteristics”** of this data sheet, supercede any specifications that may be provided in PIC24F Family Reference Manual sections.

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

This chapter discusses the Real-Time Clock and Calendar (RTCC) module, which is available on PIC24FJ16MC101/102 devices, and its operation.

Some of the key features of the RTCC module are:

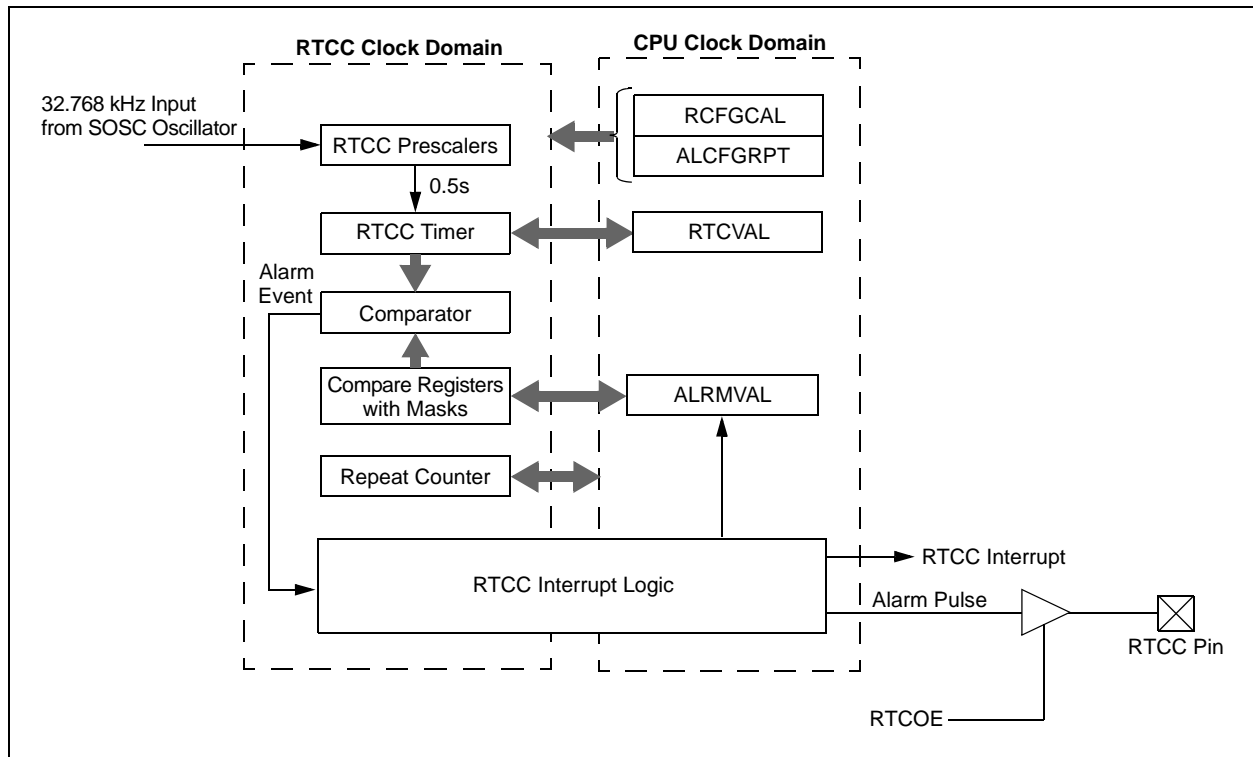
- Time: hours, minutes, and seconds
- 24-hour format (military time)
- Calendar: weekday, date, month and year
- Alarm configurable
- Year range: 2000 to 2099
- Leap year correction
- BCD format for compact firmware
- Optimized for low-power operation
- User calibration with auto-adjust
- Calibration range:  $\pm 2.64$  seconds error per month
- Requirements: External 32.768 kHz clock crystal
- Alarm pulse or seconds clock output on RTCC pin

The RTCC module is intended for applications where accurate time must be maintained for extended periods of time with minimum to no intervention from the CPU. The RTCC module is optimized for low-power usage to provide extended battery lifetime while keeping track of time.

The RTCC module is a 100-year clock and calendar with automatic leap year detection. The range of the clock is from 00:00:00 (midnight) on January 1, 2000 to 23:59:59 on December 31, 2099.

The hours are available in 24-hour (military time) format. The clock provides a granularity of one second with half-second visibility to the user.

**FIGURE 21-1: RTCC BLOCK DIAGRAM**



# PIC24FJ16MC101/102

## 21.1 RTCC Module Registers

The RTCC module registers are organized into three categories:

- RTCC Control Registers
- RTCC Value Registers
- Alarm Value Registers

### 21.1.1 REGISTER MAPPING

To limit the register interface, the RTCC Timer and Alarm Time registers are accessed through corresponding register pointers. The RTCC Value register window (RTCVALH and RTCVALL) uses the RTCPTR bits (RCFGCAL<9:8>) to select the desired timer register pair (see [Table 21-1](#)).

By writing the RTCVALH byte, the RTCC Pointer value, RTCPTR<1:0> bits, decrement by one until they reach '00'. Once they reach '00', the MINUTES and SECONDS value will be accessible through RTCVALH and RTCVALL until the pointer value is manually changed.

**TABLE 21-1: RTCVAL REGISTER MAPPING**

RTCPTR <1:0>	RTCC Value Register Window	
	RTCVAL<15:8>	RTCVAL<7:0>
00	MINUTES	SECONDS
01	WEEKDAY	HOURS
10	MONTH	DAY
11	—	YEAR

The Alarm Value register window (ALRMVALH and ALRMVALL) uses the ALRMPTR bits (ALCFGRPT<9:8>) to select the desired Alarm register pair (see [Table 21-2](#)).

By writing the ALRMVALH byte, the Alarm Pointer value, ALRMPTR<1:0> bits, decrement by one until they reach '00'. Once they reach '00', the ALRMMIN and ALRMSEC value will be accessible through ALRMVALH and ALRMVALL until the pointer value is manually changed.

**TABLE 21-2: ALRMVAL REGISTER MAPPING**

ALRMPTR <1:0>	Alarm Value Register Window	
	ALRMVAL<15:8>	ALRMVAL<7:0>
00	ALRMMIN	ALRMSEC
01	ALRMWD	ALRMHR
10	ALRMMNTH	ALRMDAY
11	—	—

Considering that the 16-bit core does not distinguish between 8-bit and 16-bit read operations, the user must be aware that when reading either the ALRMVALH or ALRMVALL bytes will decrement the ALRMPTR<1:0> value. The same applies to the RTCVALH or RTCVALL bytes with the RTCPTR<1:0> being decremented.

**Note:** This only applies to read operations and not write operations.

### 21.1.2 WRITE LOCK

In order to perform a write to any of the RTCC Timer registers, the RTCWREN bit (RCFGCAL<13>) must be set (refer to [Example 21-1](#)).

**Note:** To avoid accidental writes to the timer, it is recommended that the RTCWREN bit (RCFGCAL<13>) is kept clear at any other time. For the RTCWREN bit to be set, there is only 1 instruction cycle time window allowed between the 55h/AA sequence and the setting of RTCWREN; therefore, it is recommended that code follow the procedure in [Example 21-1](#).

**EXAMPLE 21-1: SETTING THE RTCWREN BIT**

```
MOV    #NVMKEY, W1      ;move the address of NVMKEY into W1
MOV    #0x55, W2
MOV    #0xAA, W3
MOV    W2, [W1]         ;start 55/AA sequence
MOV    W3, [W1]
BSET   RCFGCAL, #13     ;set the RTCWREN bit
```



## REGISTER 21-1: RCFGAL: RTCC CALIBRATION AND CONFIGURATION REGISTER<sup>(1)</sup>

R/W-0	U-0	R/W-0	R-0	R-0	R/W-0	R/W-0	R/W-0
RTCEN <sup>(2)</sup>	—	RTCWREN	RTCSYNC	HALFSEC <sup>(3)</sup>	RTCOE	RTCPTR<1:0>	
bit 15						bit 8	

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CAL<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      **RTCEN:** RTCC Enable bit<sup>(2)</sup>  
 1 = RTCC module is enabled  
 0 = RTCC module is disabled
- bit 14      **Unimplemented:** Read as '0'
- bit 13      **RTCWREN:** RTCC Value Registers Write Enable bit  
 1 = RTCVALH and RTCVALL registers can be written to by the user  
 0 = RTCVALH and RTCVALL registers are locked out from being written to by the user
- bit 12      **RTCSYNC:** RTCC Value Registers Read Synchronization bit  
 1 = RTCVALH, RTCVALL and ALCFGRPT registers can change while reading due to a rollover ripple resulting in an invalid data read. If the register is read twice and results in the same data, the data can be assumed to be valid.  
 0 = RTCVALH, RTCVALL or ALCFGRPT registers can be read without concern over a rollover ripple
- bit 11      **HALFSEC:** Half-Second Status bit<sup>(3)</sup>  
 1 = Second half period of a second  
 0 = First half period of a second
- bit 10      **RTCOE:** RTCC Output Enable bit  
 1 = RTCC output enabled  
 0 = RTCC output disabled
- bit 9-8      **RTCPTR<1:0>:** RTCC Value Register Window Pointer bits  
 Points to the corresponding RTCC Value registers when reading RTCVALH and RTCVALL registers; the RTCPTR<1:0> value decrements on every read or write of RTCVALH until it reaches '00'.  
RTCVAL<15:8>:  
 00 = MINUTES  
 01 = WEEKDAY  
 10 = MONTH  
 11 = Reserved  
RTCVAL<7:0>:  
 00 = SECONDS  
 01 = HOURS  
 10 = DAY  
 11 = YEAR

- Note 1:** The RCFGAL register is only affected by a POR.  
**2:** A write to the RTCEN bit is only allowed when RTCWREN = 1.  
**3:** This bit is read-only. It is cleared to '0' on a write to the lower half of the MINSEC register.

# PIC24FJ16MC101/102

---

---

## REGISTER 21-1: RCFGAL: RTCC CALIBRATION AND CONFIGURATION REGISTER<sup>(1)</sup> (CONTINUED)

bit 7-0      **CAL<7:0>**: RTC Drift Calibration bits

- 01111111 = Maximum positive adjustment; adds 508 RTC clock pulses every one minute
- 
- 
- 
- 00000001 = Minimum positive adjustment; adds 4 RTC clock pulses every one minute
- 00000000 = No adjustment
- 11111111 = Minimum negative adjustment; subtracts 4 RTC clock pulses every one minute
- 
- 
- 
- 10000000 = Maximum negative adjustment; subtracts 512 RTC clock pulses every one minute

- Note 1:** The RCFGAL register is only affected by a POR.
- 2:** A write to the RTCEN bit is only allowed when RTCWREN = 1.
- 3:** This bit is read-only. It is cleared to '0' on a write to the lower half of the MINSEC register.

## REGISTER 21-2: PADCFG1: PAD CONFIGURATION CONTROL REGISTER

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

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	U-0
—	—	—	—	—	—	RTSECSEL <sup>(1)</sup>	—
bit 7						bit 0	

### Legend:

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

bit 15-2      **Unimplemented:** Read as '0'

bit 1      **RTSECSEL:** RTCC Seconds Clock Output Select bit<sup>(1)</sup>

1 = RTCC seconds clock is selected for the RTCC pin

0 = RTCC alarm pulse is selected for the RTCC pin

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

**Note 1:** To enable the actual RTCC output, the RTCOE (RCFGCAL) bit needs to be set.

# PIC24FJ16MC101/102

## REGISTER 21-3: ALCFGRPT: ALARM CONFIGURATION REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ALRMEN	CHIME	AMASK<3:0>			ALRMPTR<1:0>		
bit 15						bit 8	

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ARPT<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      **ALRMEN:** Alarm Enable bit  
 1 = Alarm is enabled (cleared automatically after an alarm event whenever ARPT<7:0> = 0x00 and CHIME = 0)  
 0 = Alarm is disabled
- bit 14      **CHIME:** Chime Enable bit  
 1 = Chime is enabled; ARPT<7:0> bits are allowed to roll over from 0x00 to 0xFF  
 0 = Chime is disabled; ARPT<7:0> bits stop once they reach 0x00
- bit 13-10   **AMASK<3:0>:** Alarm Mask Configuration bits  
 0000 = Every half second  
 0001 = Every second  
 0010 = Every 10 seconds  
 0011 = Every minute  
 0100 = Every 10 minutes  
 0101 = Every hour  
 0110 = Once a day  
 0111 = Once a week  
 1000 = Once a month  
 1001 = Once a year (except when configured for February 29th, once every 4 years)  
 101x = Reserved – do not use  
 11xx = Reserved – do not use
- bit 9-8     **ALRMPTR<1:0>:** Alarm Value Register Window Pointer bits  
 Points to the corresponding Alarm Value registers when reading ALRMVALH and ALRMVALL registers; the ALRMPTR<1:0> value decrements on every read or write of ALRMVALH until it reaches '00'.  
ALRMVAL<15:8>:  
 00 = ALRMMIN  
 01 = ALRMWD  
 10 = ALRMMNTH  
 11 = Unimplemented  
ALRMVAL<7:0>:  
 00 = ALRMSEC  
 01 = ALRMHR  
 10 = ALRMDAY  
 11 = Unimplemented
- bit 7-0     **ARPT<7:0>:** Alarm Repeat Counter Value bits  
 11111111 = Alarm will repeat 255 more times  
 •  
 •  
 •  
 00000000 = Alarm will not repeat  
 The counter decrements on any alarm event. The counter is prevented from rolling over from 0x00 to 0xFF unless CHIME = 1.

## REGISTER 21-4: RTCVAL (WHEN RTCPTR<1:0> = 11): YEAR VALUE REGISTER<sup>(1)</sup>

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

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
YRTEN<3:0>				YRONE<3:0>			
bit 7							bit 0

### Legend:

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

- bit 15-8            **Unimplemented:** Read as '0'
- bit 7-4            **YRTEN<3:0>:** Binary Coded Decimal Value of Year's Tens Digit; contains a value from 0 to 9
- bit 3-0            **YRONE<3:0>:** Binary Coded Decimal Value of Year's Ones Digit; contains a value from 0 to 9

**Note 1:** A write to the YEAR register is only allowed when RTCWREN = 1.

## REGISTER 21-5: RTCVAL (WHEN RTCPTR<1:0> = 10): MONTH AND DAY VALUE REGISTER<sup>(1)</sup>

U-0	U-0	U-0	R-x	R-x	R-x	R-x	R-x
—	—	—	MHTTEN0	MTHONE<3:0>			
bit 15							bit 8

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	DAYTEN<1:0>		DAYONE<3:0>			
bit 7							bit 0

### Legend:

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

- bit 15-13            **Unimplemented:** Read as '0'
- bit 12              **MHTTEN0:** Binary Coded Decimal Value of Month's Tens Digit; contains a value of 0 or 1
- bit 11-8            **MTHONE<3:0>:** Binary Coded Decimal Value of Month's Ones Digit; contains a value from 0 to 9
- bit 7-6              **Unimplemented:** Read as '0'
- bit 5-4              **DAYTEN<1:0>:** Binary Coded Decimal Value of Day's Tens Digit; contains a value from 0 to 3
- bit 3-0              **DAYONE<3:0>:** Binary Coded Decimal Value of Day's Ones Digit; contains a value from 0 to 9

**Note 1:** A write to this register is only allowed when RTCWREN = 1.

# PIC24FJ16MC101/102

## REGISTER 21-6: RTCVAL (WHEN RTCPTR<1:0> = 01): WKDYHR: WEEKDAY AND HOURS VALUE REGISTER<sup>(1)</sup>

U-0	U-0	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x
—	—	—	—	—	WDAY<2:0>		
bit 15					bit 8		

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	HRTEN<1:0>		HRONE<3:0>			
bit 7		bit 0					

### Legend:

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

- bit 15-11        **Unimplemented:** Read as '0'
- bit 10-8        **WDAY<2:0>:** Binary Coded Decimal Value of Weekday Digit; contains a value from 0 to 6
- bit 7-6         **Unimplemented:** Read as '0'
- bit 5-4         **HRTEN<1:0>:** Binary Coded Decimal Value of Hour's Tens Digit; contains a value from 0 to 2
- bit 3-0         **HRONE<3:0>:** Binary Coded Decimal Value of Hour's Ones Digit; contains a value from 0 to 9

**Note 1:** A write to this register is only allowed when RTCWREN = 1.

## REGISTER 21-7: RTCVAL (WHEN RTCPTR<1:0> = 00): MINUTES AND SECONDS VALUE REGISTER

U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	MINTEN<2:0>			MINONE<3:0>			
bit 15					bit 8		

U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	SECTEN<2:0>			SECONE<3:0>			
bit 7		bit 0					

### Legend:

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

- bit 15         **Unimplemented:** Read as '0'
- bit 14-12      **MINTEN<2:0>:** Binary Coded Decimal Value of Minute's Tens Digit; contains a value from 0 to 5
- bit 11-8      **MINONE<3:0>:** Binary Coded Decimal Value of Minute's Ones Digit; contains a value from 0 to 9
- bit 7          **Unimplemented:** Read as '0'
- bit 6-4        **SECTEN<2:0>:** Binary Coded Decimal Value of Second's Tens Digit; contains a value from 0 to 5
- bit 3-0        **SECONE<3:0>:** Binary Coded Decimal Value of Second's Ones Digit; contains a value from 0 to 9

## REGISTER 21-8: ALRMVAL (WHEN ALRMPTR<1:0> = 10): ALARM MONTH AND DAY VALUE REGISTER<sup>(1)</sup>

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	
—	—	—	MHTTEN0	MTHONE<3:0>				
bit 15								bit 8

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	
—	—	DAYTEN<1:0>		DAYONE<3:0>				
bit 7								bit 0

### Legend:

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

- bit 15-13      **Unimplemented:** Read as '0'
- bit 12        **MHTTEN0:** Binary Coded Decimal Value of Month's Tens Digit; contains a value of 0 or 1
- bit 11-8      **MTHONE<3:0>:** Binary Coded Decimal Value of Month's Ones Digit; contains a value from 0 to 9
- bit 7-6        **Unimplemented:** Read as '0'
- bit 5-4        **DAYTEN<1:0>:** Binary Coded Decimal Value of Day's Tens Digit; contains a value from 0 to 3
- bit 3-0        **DAYONE<3:0>:** Binary Coded Decimal Value of Day's Ones Digit; contains a value from 0 to 9

**Note 1:** A write to this register is only allowed when RTCWREN = 1.

## REGISTER 21-9: ALRMVAL (WHEN ALRMPTR<1:0> = 01): ALARM WEEKDAY AND HOURS VALUE REGISTER<sup>(1)</sup>

U-0	U-0	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	
—	—	—	—	—	WDAY2	WDAY1	WDAY0	
bit 15								bit 8

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	
—	—	HRTEN<1:0>		HRONE<3:0>				
bit 7								bit 0

### Legend:

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

- bit 15-11      **Unimplemented:** Read as '0'
- bit 10-8      **WDAY<2:0>:** Binary Coded Decimal Value of Weekday Digit; contains a value from 0 to 6
- bit 7-6        **Unimplemented:** Read as '0'
- bit 5-4        **HRTEN<1:0>:** Binary Coded Decimal Value of Hour's Tens Digit; contains a value from 0 to 2
- bit 3-0        **HRONE<3:0>:** Binary Coded Decimal Value of Hour's Ones Digit; contains a value from 0 to 9

**Note 1:** A write to this register is only allowed when RTCWREN = 1.

# PIC24FJ16MC101/102

## REGISTER 21-10: ALRMVAL (WHEN ALRMPTR<1:0> = 00): ALARM MINUTES AND SECONDS VALUE REGISTER

U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	MINTEN<2:0>			MINONE<3:0>			
bit 15							bit 8

U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	SECTEN<2:0>			SECONE<3:0>			
bit 7							bit 0

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 15

**Unimplemented:** Read as '0'

bit 14-12

**MINTEN<2:0>:** Binary Coded Decimal Value of Minute's Tens Digit; contains a value from 0 to 5

bit 11-8

**MINONE<3:0>:** Binary Coded Decimal Value of Minute's Ones Digit; contains a value from 0 to 9

bit 7

**Unimplemented:** Read as '0'

bit 6-4

**SECTEN<2:0>:** Binary Coded Decimal Value of Second's Tens Digit; contains a value from 0 to 5

bit 3-0

**SECONE<3:0>:** Binary Coded Decimal Value of Second's Ones Digit; contains a value from 0 to 9



## 22.0 CHARGE TIME MEASUREMENT UNIT (CTMU)

**Note 1:** This data sheet summarizes the features of the PIC24FJ16MC101/102 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. “Charge Time Measurement Unit (CTMU)”** (DS39724) in the *“PIC24F Family Reference Manual”*, which is available from the Microchip web site ([www.microchip.com](http://www.microchip.com)).

**2:** It is important to note that the specifications in **Section 26.0 “Electrical Characteristics”** of this data sheet, supercede any specifications that may be provided in PIC24F Family Reference Manual sections.

**3:** 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 Charge Time Measurement Unit is a flexible analog module that provides accurate differential time measurement between pulse sources, as well as asynchronous pulse generation. Its key features include:

- Four edge input trigger sources
- Polarity control for each edge source
- Control of edge sequence
- Control of response to edges
- Precise time measurement resolution of 200 ps
- Accurate current source suitable for capacitive measurement
- On-chip temperature measurement using a built-in diode

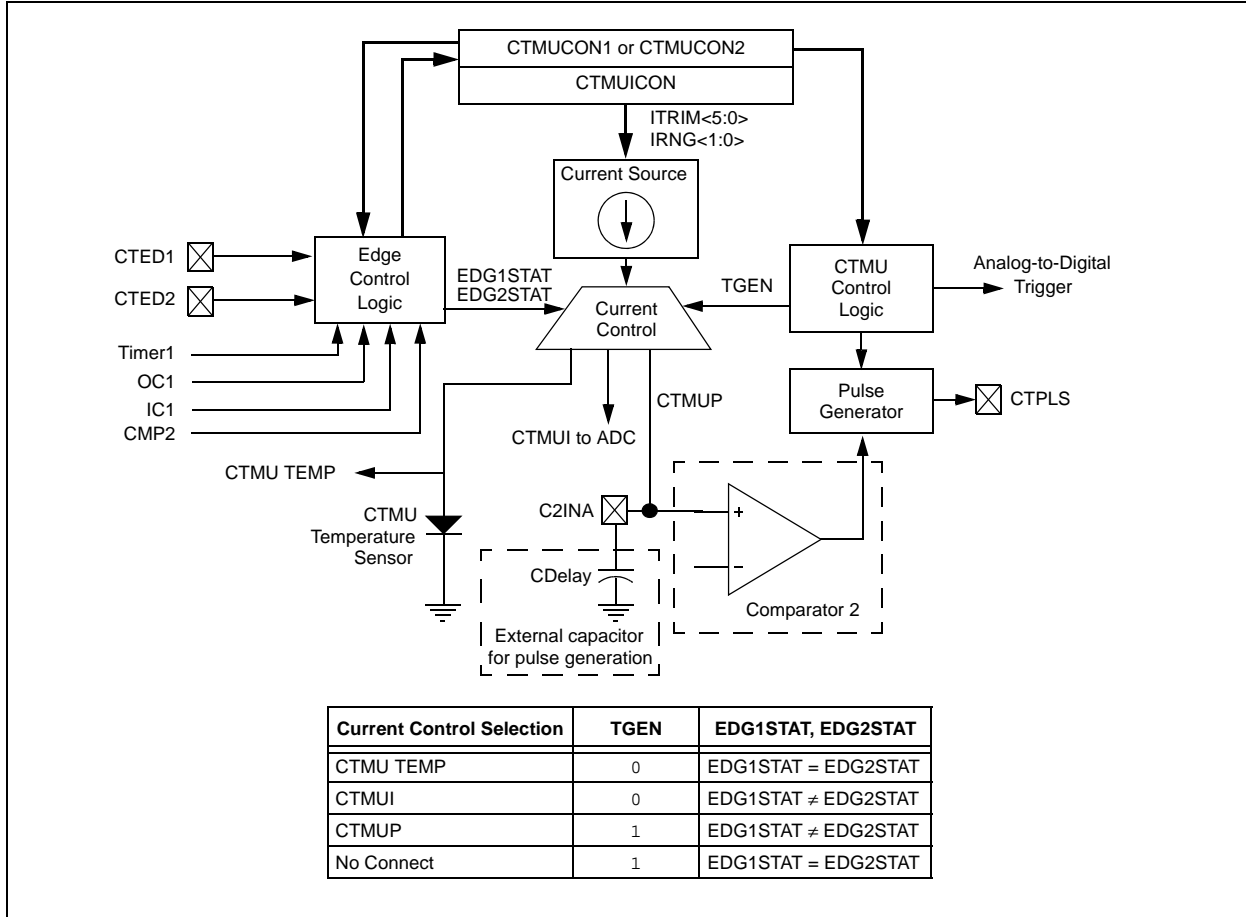
Together with other on-chip analog modules, the CTMU can be used to precisely measure time, measure capacitance, measure relative changes in capacitance or generate output pulses that are independent of the system clock.

The CTMU module is ideal for interfacing with capacitive-based sensors. The CTMU is controlled through three registers: CTMUCON1, CTMUCON2 and CTMUICON. CTMUCON1 enables the module, the Edge delay generation, sequencing of edges and controls the current source and the output trigger. CTMUCON2 controls the edge source selection, edge source polarity selection and edge sampling mode. The CTMUICON register controls the selection and trim of the current source.

Figure 22-1 shows the CTMU block diagram.

# PIC24FJ16MC101/102

FIGURE 22-1: CTMU BLOCK DIAGRAM



## REGISTER 22-1: CTMUCON1: CTMU CONTROL REGISTER 1

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CTMUEN	—	CTMUSIDL	TGEN <sup>(1)</sup>	EDGEN	EDGSEQEN	IDISSEN <sup>(2)</sup>	CTTRIG
bit 15							bit 8

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

### Legend:

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

- bit 15            **CTMUEN:** CTMU Enable bit  
                   1 = Module is enabled  
                   0 = Module is disabled
- bit 14            **Unimplemented:** Read as '0'
- bit 13            **CTMUSIDL:** Stop in Idle Mode bit  
                   1 = Discontinue module operation when device enters Idle mode  
                   0 = Continue module operation in Idle mode
- bit 12            **TGEN:** Time Generation Enable bit<sup>(1)</sup>  
                   1 = Enables edge delay generation  
                   0 = Disables edge delay generation
- bit 11            **EDGEN:** Edge Enable bit  
                   1 = Edges are not blocked  
                   0 = Edges are blocked
- bit 10            **EDGSEQEN:** Edge Sequence Enable bit  
                   1 = Edge 1 event must occur before Edge 2 event can occur  
                   0 = No edge sequence is needed
- bit 9             **IDISSEN:** Analog Current Source Control bit<sup>(2)</sup>  
                   1 = Analog current source output is grounded  
                   0 = Analog current source output is not grounded
- bit 8             **CTTRIG:** Trigger Control bit  
                   1 = Trigger output is enabled  
                   0 = Trigger output is disabled
- bit 7-0           **Unimplemented:** Read as '0'

- Note 1:** If TGEN = 1, the peripheral inputs and outputs must be configured to an available RPN pin. For more information, see [Section 10.4 "Peripheral Pin Select"](#).
- 2:** The ADC module Sample & Hold capacitor is not automatically discharged between sample/conversion cycles. Software using the ADC as part of a capacitance measurement, must discharge the ADC capacitor before conducting the measurement. The IDISSEN bit, when set to '1', performs this function. The ADC must be sampling while the IDISSEN bit is active to connect the discharge sink to the capacitor array.

# PIC24FJ16MC101/102

## REGISTER 22-2: CTMUCON2: CTMU CONTROL REGISTER 2

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
EDG1MOD	EDG1POL	EDG1SEL<3:0>			EDG2STAT	EDG1STAT	
bit 15						bit 8	

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0
EDG2MOD	EDG2POL	EDG2SEL<3:0>			—	—	
bit 7						bit 0	

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 15 **EDG1MOD:** Edge 1 Edge Sampling Selection bit

1 = Edge 1 is edge sensitive

0 = Edge 1 is level sensitive

bit 14 **EDG1POL:** Edge 1 Polarity Select bit

1 = Edge 1 programmed for a positive edge response

0 = Edge 1 programmed for a negative edge response

bit 13-10 **EDG1SEL<3:0>:** Edge 1 Source Select bits

1xxx = Reserved

01xx = Reserved

0011 = CTED1 pin

0010 = CTED2 pin

0001 = OC1 module

0000 = Timer1 module

bit 9 **EDG2STAT:** Edge 2 Status bit

Indicates the status of Edge 2 and can be written to control the edge source.

1 = Edge 2 has occurred

0 = Edge 2 has not occurred

bit 8 **EDG1STAT:** Edge 1 Status bit

Indicates the status of Edge 1 and can be written to control the edge source.

1 = Edge 1 has occurred

0 = Edge 1 has not occurred

bit 7 **EDG2MOD:** Edge 2 Edge Sampling Selection bit

1 = Edge 2 is edge sensitive

0 = Edge 2 is level sensitive

bit 6 **EDG2POL:** Edge 2 Polarity Select bit

1 = Edge 2 programmed for a positive edge response

0 = Edge 2 programmed for a negative edge response

bit 5-2 **EDG2SEL<3:0>:** Edge 2 Source Select bits

1xxx = Reserved

01xx = Reserved

0011 = CTED2 pin

0010 = CTED1 pin

0001 = Comparator 2 module

0000 = IC1 module

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

## REGISTER 22-3: CTMUICON: CTMU CURRENT CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ITRIM<5:0>						IRNG<1:0>	
bit 15						bit 8	

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

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

bit 15-10    **ITRIM<5:0>**: Current Source Trim bits

011111 = Nominal current output specified by IRNG<1:0> + 62%

011110 = Nominal current output specified by IRNG<1:0> + 60%

•

•

•

000001 = Nominal current output specified by IRNG<1:0> + 2%

000000 = Nominal current output specified by IRNG<1:0>

111111 = Nominal current output specified by IRNG<1:0> – 2%

•

•

•

100010 = Nominal current output specified by IRNG<1:0> – 62%

100001 = Nominal current output specified by IRNG<1:0> – 64%

bit 9-8    **IRNG<1:0>**: Current Source Range Select bits

11 = 100 × Base Current<sup>(1)</sup>

10 = 10 × Base Current

01 = Base current level (0.55 μA nominal)

00 = Reserved

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

**Note 1:** This setting must be used for the CTMU temperature sensor.

# PIC24FJ16MC101/102

---

---

NOTES:

## 23.0 SPECIAL FEATURES

**Note 1:** This data sheet summarizes the features of the PIC24FJ16MC101/102 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 (WDT)”** (DS39697) and **Section 33. “Programming and Diagnostics”** (DS39716) in the “PIC24F Family Reference Manual”, which are available from the Microchip web site ([www.microchip.com](http://www.microchip.com)).

**2:** It is important to note that the specifications in **Section 26.0 “Electrical Characteristics”** of this data sheet, supercede any specifications that may be provided in PIC24F Family Reference Manual sections.

**3:** 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.

PIC24FJ16MC101/102 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
- In-Circuit Serial Programming™ (ICSP™)
- In-Circuit emulation

## 23.1 Configuration Bits

The Configuration Shadow register bits can be configured (read as ‘0’), or left unprogrammed (read as ‘1’), to select various device configurations. These read-only bits are mapped starting at program memory location 0xF80000. A detailed explanation of the various bit functions is provided in [Table 23-3](#).

Note that address 0xF80000 is beyond the user program memory space and belongs to the configuration memory space (0x800000-0xFFFFF) which can only be accessed using table reads.

In PIC24FJ16MC101/102 devices, the configuration bytes are implemented as volatile memory. This means that configuration data must be programmed each time the device is powered up. Configuration data is stored in the two words at the top of the on-chip program memory space, known as the Flash Configuration Words. Their specific locations are shown in [Table 23-2](#). These are packed representations of the actual device Configuration bits, whose actual locations are distributed among several locations in configuration space. The configuration data is automatically loaded from the Flash Configuration Words to the proper Configuration registers during device Resets.

**Note:** Configuration data is reloaded on all types of device Resets.

When creating applications for these devices, users should always specifically allocate the location of the Flash Configuration Word for configuration data. This is to make certain that program code is not stored in this address when the code is compiled.

The upper byte of all Flash Configuration Words in program memory should always be ‘1111 1111’. This makes them appear to be NOP instructions in the remote event that their locations are ever executed by accident. Since Configuration bits are not implemented in the corresponding locations, writing ‘1’s to these locations has no effect on device operation.

**Note:** Performing a page erase operation on the last page of program memory clears the Flash Configuration Words, enabling code protection as a result. Therefore, users should avoid performing page erase operations on the last page of program memory.

The Configuration Shadow register map is shown in [Table 23-1](#).

**TABLE 23-1: CONFIGURATION SHADOW REGISTER MAP**

File Name	Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
FGS	F80004	—	—	—	—	—	—	GCP	GWRP
FOSCSEL	F80006	IESO	PWMLOCK	—	WDTWIN<1:0>		FNOSC<2:0>		
FOSC	F80008	FCKSM<1:0>		IOL1WAY	—	—	OSCI0FNC	POSCMD<1:0>	
FWDT	F8000A	FWDTEN	WINDIS	PLLKEN	WDTPRE	WDTPOST<3:0>			
FPOR	F8000C	PWMPIN	HPOL	LPOL	ALT12C1	—	—	—	—
FICD	F8000E	Reserved <sup>(1)</sup>	—	Reserved <sup>(2)</sup>	Reserved <sup>(2)</sup>	—	—	ICS<1:0>	

**Legend:** — = unimplemented, read as '1'.

**Note 1:** This bit is reserved for use by development tools and must be programmed as '1'.

**Note 2:** This bit is reserved; program as '0'.

The Configuration Flash Words map is shown in [Table 23-2](#).

**TABLE 23-2: CONFIGURATION FLASH WORDS**

File Name	Addr.	Bits 23-16	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
CONFIG2	002BFC	—	IESO	PWMLOCK	PWMPIN	WDTWIN<1:0>		FNOSC<2:0>		FCKSM<1:0>		OSCI0FNC	IOL1WAY	LPOL	ALT12C1	POSCMD<1:0>		
CONFIG1	002BFE	—	Reserved <sup>(2)</sup>	Reserved <sup>(2)</sup>	GCP	GWRP	Reserved <sup>(3)</sup>	HPOL	ICS<1:0>		FWDTEN	WINDIS	PLLKEN	WDTPRE	WDTPOST<3:0>			

**Legend:** — = unimplemented, read as '1'.

**Note 1:** During a Power-on Reset (POR), the contents of these Flash locations are transferred to the Configuration Shadow registers.

**Note 2:** This bit is reserved; program as '0'.

**Note 3:** This bit is reserved for use by development tools and must be programmed as '1'.



**TABLE 23-3: PIC24F CONFIGURATION BITS DESCRIPTION**

Bit Field	RTSP Effect	Description
GCP	Immediate	General Segment Code-Protect bit 1 = User program memory is not code-protected 0 = Code protection is enabled for the entire program memory space
GWRP	Immediate	General Segment Write-Protect bit 1 = User program memory is not write-protected 0 = User program memory is write-protected
IESO	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
PWMLOCK	Immediate	PWM Lock Enable bit 1 = Certain PWM registers may only be written after key sequence 0 = PWM registers may be written without key
WDTWIN<1:0>	Immediate	Watchdog Window Select bits 11 = WDT Window is 25% of WDT period 10 = WDT Window is 37.5% of WDT period 01 = WDT Window is 50% of WDT period 00 = WDT Window is 75% of WDT period
FNOSC<2:0>	Immediate	Oscillator Selection bits 111 = Fast RC Oscillator with divide-by-N (FRCDIVN) 110 = Reserved; do not use 101 = Low-Power RC Oscillator (LPRC) 100 = Secondary Oscillator (Sosc) 011 = Primary Oscillator with PLL module (MS + PLL, EC + PLL) 010 = Primary Oscillator (MS, HS, EC) 001 = Fast RC Oscillator with divide-by-N with PLL module (FRCDIVN + PLL) 000 = Fast RC Oscillator (FRC)
FCKSM<1:0>	If clock switch is enabled, RTSP effect is on any device Reset; otherwise, 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	Immediate	Peripheral pin select configuration 1 = Allow only one reconfiguration 0 = Allow multiple reconfigurations
OSCIOFNC	Immediate	OSC2 Pin Function bit (except in MS and HS modes) 1 = OSC2 is clock output 0 = OSC2 is general purpose digital I/O pin
POSCMD<1:0>	Immediate	Primary Oscillator Mode Select bits 11 = Primary oscillator disabled 10 = HS Crystal Oscillator mode (10 MHz - 32 MHz) 01 = MS Crystal Oscillator mode (3 MHz - 10 MHz) 00 = EC (External Clock) mode (DC - 32 MHz)
FWDTEN	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	Immediate	Watchdog Timer Window Enable bit 1 = Watchdog Timer in Non-Window mode 0 = Watchdog Timer in Window mode

# PIC24FJ16MC101/102

**TABLE 23-3: PIC24F CONFIGURATION BITS DESCRIPTION (CONTINUED)**

Bit Field	RTSP Effect	Description
WDTPRE	Immediate	Watchdog Timer Prescaler bit 1 = 1:128 0 = 1:32
WDTPOST<3:0>	Immediate	Watchdog Timer Postscaler bits 1111 = 1:32,768 1110 = 1:16,384 • • • 0001 = 1:2 0000 = 1:1
PLLKEN	Immediate	PLL Lock Enable bit 1 = Clock switch to PLL will wait until the PLL lock signal is valid 0 = Clock switch will not wait for the PLL lock signal
ALT2C	Immediate	Alternate I <sup>2</sup> C pins 1 = I <sup>2</sup> C™ mapped to SDA1/SCL1 pins 0 = I <sup>2</sup> C mapped to ASDA1/ASCL1 pins
ICS<1:0>	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
PWMPIN	Immediate	Motor Control PWM Module Pin Mode bit 1 = PWM module pins controlled by PORT register at device Reset (tri-stated) 0 = PWM module pins controlled by PWM module at device Reset (configured as output pins)
HPOL	Immediate	Motor Control PWM High Side Polarity bit 1 = PWM module high side output pins have active-high output polarity 0 = PWM module high side output pins have active-low output polarity
LPOL	Immediate	Motor Control PWM Low Side Polarity bit 1 = PWM module low side output pins have active-high output polarity 0 = PWM module low side output pins have active-low output polarity



# PIC24FJ16MC101/102

## 23.2 On-Chip Voltage Regulator

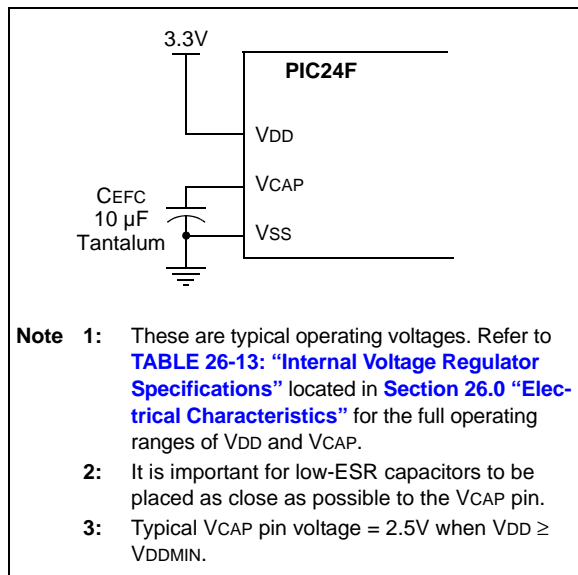
All of the PIC24FJ16MC101/102 devices power their core digital logic at a nominal 2.5V. This can create a conflict for designs that are required to operate at a higher typical voltage, such as 3.3V. To simplify system design, all devices in the PIC24FJ16MC101/102 family incorporate an on-chip regulator 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 23-1). This helps to maintain the stability of the regulator. The recommended value for the filter capacitor is provided in Table 26-13 located in Section 26.0 “Electrical Characteristics”.

**Note:** It is important for low-ESR capacitors to be placed as close as possible to the VCAP pin.

On a POR, it takes approximately 20  $\mu$ s for the on-chip voltage regulator to generate an output voltage. During this time, designated as TSTARTUP, code execution is disabled. TSTARTUP is applied every time the device resumes operation after any power-down.

**FIGURE 23-1: CONNECTIONS FOR THE ON-CHIP VOLTAGE REGULATOR<sup>(1,2,3)</sup>**



## 23.3 BOR: Brown-out Reset

The Brown-out Reset (BOR) module is based on an internal voltage reference circuit that monitors the regulated supply voltage VCAP. The main purpose of the BOR module is to generate a device Reset when a brown-out condition occurs. Brown-out conditions are generally caused by glitches on the AC mains (for example, missing portions of the AC cycle waveform due to bad power transmission lines, or voltage sags due to excessive current draw when a large inductive load is turned on).

A BOR generates a Reset pulse, which resets the device. The BOR selects the clock source, based on the device Configuration bit values (FNOSC<2:0> and POSCMD<1:0>).

If an oscillator mode is selected, the BOR activates the Oscillator Start-up Timer (OST). The system clock is held until OST expires. If the PLL is used, the clock is held until the LOCK bit (OSCCON<5>) is '1'.

Concurrently, the PWRT time-out (TPWRT) is applied before the internal Reset is released. If TPWRT = 0 and a crystal oscillator is being used, then a nominal delay of TFSCM = 100 is applied. The total delay in this case is TFSCM.

The BOR Status bit (RCON<1>) is set to indicate that a BOR has occurred. The BOR circuit continues to operate while in Sleep or Idle modes and resets the device should VDD fall below the BOR threshold voltage.

## 23.4 Watchdog Timer (WDT)

For PIC24FJ16MC101/102 devices, the WDT is driven by the LPRC oscillator. When the WDT is enabled, the clock source is also enabled.

### 23.4.1 PRESCALER/POSTSCALER

The nominal WDT clock source from LPRC is 32 kHz. This feeds a prescaler that can be configured for either 5-bit (divide-by-32) or 7-bit (divide-by-128) operation. The prescaler is set by the WDTPRE Configuration bit. With a 32 kHz input, the prescaler yields a nominal WDT time-out period (TWDT) of 1 ms in 5-bit mode, or 4 ms in 7-bit mode.

A variable postscaler divides down the WDT prescaler output and allows for a wide range of time-out periods. The postscaler is controlled by the WDTPOST<3:0> Configuration bits (FWDT<3:0>), which allow the selection of 16 settings, from 1:1 to 1:32,768. Using the prescaler and postscaler, time-out periods ranging from 1 ms to 131 seconds can be achieved.

The WDT, prescaler, and postscaler are reset:

- On any device Reset
- On the completion of a clock switch, whether invoked by software (i.e., setting the OSWEN bit after changing the NOSC bits) or by hardware (i.e., Fail-Safe Clock Monitor)
- When a 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.

### 23.4.2 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> and RCON<2>, respectively) will need to be cleared in software after the device wakes up.

### 23.4.3 ENABLING WDT

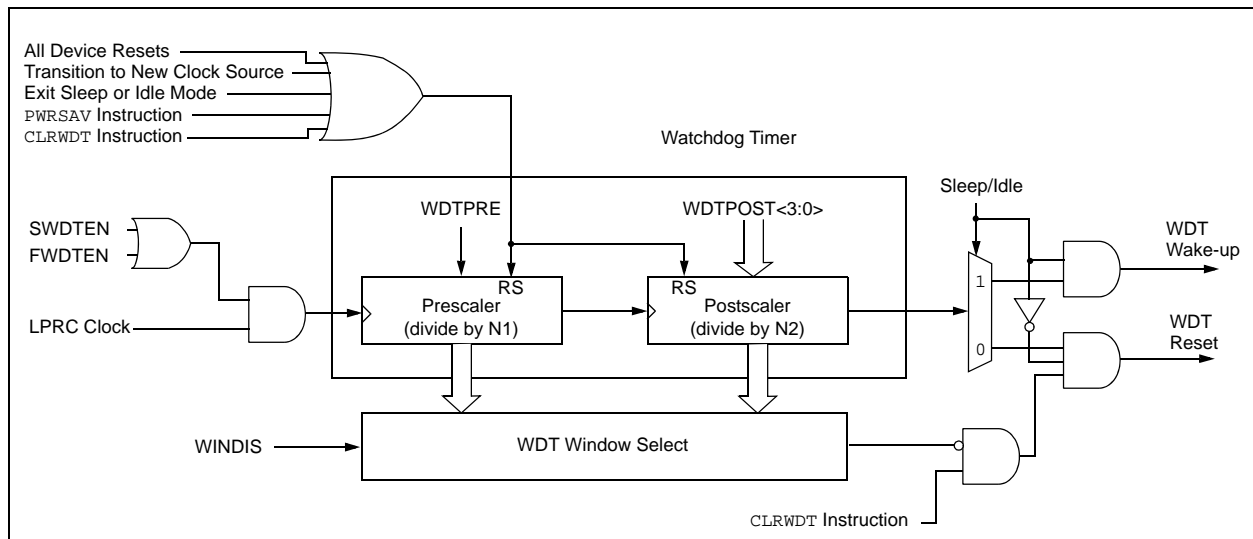
The WDT is enabled or disabled by the FWDTEN Configuration bit in the FWDT Configuration register. When the FWDTEN Configuration bit is set, the WDT is always enabled.

The WDT can be optionally controlled in software when the FWDTEN Configuration bit has been programmed to '0'. The WDT is enabled in software by setting the SWDTEN control bit (RCON<5>). The SWDTEN control bit is cleared on any device Reset. The software WDT option allows the user application to enable the WDT for critical code segments and disable the WDT during non-critical segments for maximum power savings.

**Note:** If the WINDIS bit (FWDT<6>) is cleared, the CLRWDT instruction should be executed by the application software only during the last 1/4 of the WDT period. This CLRWDT window can be determined by using a timer. If a CLRWDT instruction is executed before this window, a WDT Reset occurs.

The WDT flag bit, WDTO (RCON<4>), is not automatically cleared following a WDT time-out. To detect subsequent WDT events, the flag must be cleared in software.

**FIGURE 23-2: WDT BLOCK DIAGRAM**



## 23.5 In-Circuit Serial Programming

The PIC24FJ16MC101/102 devices 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 microcontroller just before shipping the product. Serial programming also allows the most recent firmware or a custom firmware to be programmed. Refer to the *"PIC24FJXXMC Family Flash Programming Specification"* (DS70512) 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

## 23.6 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 MCLR, VDD, VSS, and the PGECx/PGEDx pin pair. In addition, when the feature is enabled, some of the resources are not available for general use. These resources include the first 80 bytes of data RAM and two I/O pins.

## 24.0 INSTRUCTION SET SUMMARY

**Note 1:** This data sheet summarizes the features of the PIC24FJ16MC101/102 devices. However, it is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to the latest family reference sections of the “PIC24F Family Reference Manual”, which are available from the Microchip web site ([www.microchip.com](http://www.microchip.com)).

**2:** It is important to note that the specifications in **Section 26.0 “Electrical Characteristics”** of this data sheet, supercede any specifications that may be provided in PIC24F Family Reference Manual sections.

The PIC24F instruction set adds many enhancements to the previous PIC<sup>®</sup> MCU instruction sets, while maintaining an easy migration from previous PIC MCU instruction sets.

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
- Control operations

**Table 24-1** shows the general symbols used in describing the instructions.

The PIC24FXXXX instruction set summary in **Table 24-2** lists all the instructions, along with the status flags affected by each instruction.

Most word or byte-oriented W register instructions (including barrel shift instructions) have three operands:

- The first source operand, which is typically a register ‘Wb’ without any address modifier
- The second source operand, which is typically a register ‘Ws’ with or without an address modifier
- The destination of the result, which is typically a register ‘Wd’ with or without an address modifier

However, word or byte-oriented file register instructions have two operands:

- The file register specified by the value ‘f’
- The destination, which could 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 control instructions can use some of the following operands:

- A program memory address
- The mode of the table read and table write instructions

# PIC24FJ16MC101/102

Most instructions are a single word. Certain double-word instructions are designed to provide all the required information in these 48 bits. In the second word, the 8 MSBs are '0's. If this second word is executed as an instruction (by itself), it will execute as a 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 24-1: SYMBOLS USED IN OPCODE DESCRIPTIONS**

Field	Description
#text	Means literal defined by "text"
(text)	Means "content of text"
[text]	Means "the location addressed by text"
{ }	Optional field or operation
<n:m>	Register bit field
.b	Byte mode selection
.d	Double-Word mode selection
.S	Shadow register select
.w	Word mode selection (default)
Acc	One of two accumulators {A, B}
AWB	Accumulator write back destination address register $\in \{W13, [W13]+2\}$
bit4	4-bit bit selection field (used in word addressed instructions) $\in \{0..15\}$
C, DC, N, OV, Z	MCU Status bits: Carry, Digit Carry, Negative, Overflow, Sticky Zero
Expr	Absolute address, label or expression (resolved by the linker)
f	File register address $\in \{0x0000..0x1FFF\}$
lit1	1-bit unsigned literal $\in \{0,1\}$
lit4	4-bit unsigned literal $\in \{0..15\}$
lit5	5-bit unsigned literal $\in \{0..31\}$
lit8	8-bit unsigned literal $\in \{0..255\}$
lit10	10-bit unsigned literal $\in \{0..255\}$ for Byte mode, $\{0:1023\}$ for Word mode
lit14	14-bit unsigned literal $\in \{0..16384\}$
lit16	16-bit unsigned literal $\in \{0..65535\}$
lit23	23-bit unsigned literal $\in \{0..8388608\}$ ; LSb must be '0'
None	Field does not require an entry, can be blank
OA, OB, SA, SB	DSP Status bits: ACCA Overflow, ACCB Overflow, ACCA Saturate, ACCB Saturate
PC	Program Counter
Slit10	10-bit signed literal $\in \{-512..511\}$
Slit16	16-bit signed literal $\in \{-32768..32767\}$
Slit6	6-bit signed literal $\in \{-16..16\}$
Wb	Base W register $\in \{W0..W15\}$
Wd	Destination W register $\in \{Wd, [Wd], [Wd++] , [Wd--], [++Wd], [--Wd]\}$
Wdo	Destination W register $\in \{Wnd, [Wnd], [Wnd++] , [Wnd--], [++Wnd], [--Wnd], [Wnd+Wb]\}$
Wm,Wn	Dividend, Divisor working register pair (direct addressing)



**TABLE 24-1: SYMBOLS USED IN OPCODE DESCRIPTIONS (CONTINUED)**

Field	Description
Wm*Wm	Multiplicand and Multiplier working register pair for Square instructions $\in \{W4 * W4, W5 * W5, W6 * W6, W7 * W7\}$
Wn	One of 16 working registers $\in \{W0..W15\}$
Wnd	One of 16 destination working registers $\in \{W0...W15\}$
Wns	One of 16 source working registers $\in \{W0...W15\}$
WREG	W0 (working register used in file register instructions)
Ws	Source W register $\in \{Ws, [Ws], [Ws++] , [Ws--], [++Ws], [--Ws]\}$
Wso	Source W register $\in \{Wns, [Wns], [Wns++] , [Wns--], [++Wns], [--Wns], [Wns+Wb]\}$

# PIC24FJ16MC101/102

**TABLE 24-2: INSTRUCTION SET OVERVIEW**

Assembly Mnemonic	Assembly Syntax	Description	# of Words	# of Cycles	Status Flags Affected
ADD	ADD <i>Acc</i>	Add Accumulators	1	1	OA,OB,SA,SB
	ADD <i>f</i>	$f = f + WREG$	1	1	C,DC,N,OV,Z
	ADD <i>f, WREG</i>	$WREG = f + WREG$	1	1	C,DC,N,OV,Z
	ADD <i>#lit10, Wn</i>	$Wd = lit10 + Wd$	1	1	C,DC,N,OV,Z
	ADD <i>Wb, Ws, Wd</i>	$Wd = Wb + Ws$	1	1	C,DC,N,OV,Z
	ADD <i>Wb, #lit5, Wd</i>	$Wd = Wb + lit5$	1	1	C,DC,N,OV,Z
	ADD <i>Wso, #slit4, Acc</i>	16-bit Signed Add to Accumulator	1	1	OA,OB,SA,SB
ADDC	ADDC <i>f</i>	$f = f + WREG + (C)$	1	1	C,DC,N,OV,Z
	ADDC <i>f, WREG</i>	$WREG = f + WREG + (C)$	1	1	C,DC,N,OV,Z
	ADDC <i>#lit10, Wn</i>	$Wd = lit10 + Wd + (C)$	1	1	C,DC,N,OV,Z
	ADDC <i>Wb, Ws, Wd</i>	$Wd = Wb + Ws + (C)$	1	1	C,DC,N,OV,Z
	ADDC <i>Wb, #lit5, Wd</i>	$Wd = Wb + lit5 + (C)$	1	1	C,DC,N,OV,Z
AND	AND <i>f</i>	$f = f .AND. WREG$	1	1	N,Z
	AND <i>f, WREG</i>	$WREG = f .AND. WREG$	1	1	N,Z
	AND <i>#lit10, Wn</i>	$Wd = lit10 .AND. Wd$	1	1	N,Z
	AND <i>Wb, Ws, Wd</i>	$Wd = Wb .AND. Ws$	1	1	N,Z
	AND <i>Wb, #lit5, Wd</i>	$Wd = Wb .AND. lit5$	1	1	N,Z
ASR	ASR <i>f</i>	$f = \text{Arithmetic Right Shift } f$	1	1	C,N,OV,Z
	ASR <i>f, WREG</i>	$WREG = \text{Arithmetic Right Shift } f$	1	1	C,N,OV,Z
	ASR <i>Ws, Wd</i>	$Wd = \text{Arithmetic Right Shift } Ws$	1	1	C,N,OV,Z
	ASR <i>Wb, Wns, Wnd</i>	$Wnd = \text{Arithmetic Right Shift } Wb \text{ by } Wns$	1	1	N,Z
	ASR <i>Wb, #lit5, Wnd</i>	$Wnd = \text{Arithmetic Right Shift } Wb \text{ by } lit5$	1	1	N,Z
BCLR	BCLR <i>f, #bit4</i>	Bit Clear <i>f</i>	1	1	None
	BCLR <i>Ws, #bit4</i>	Bit Clear <i>Ws</i>	1	1	None
BRA	BRA <i>C, Expr</i>	Branch if Carry	1	1 (2)	None
	BRA <i>GE, Expr</i>	Branch if greater than or equal	1	1 (2)	None
	BRA <i>GEU, Expr</i>	Branch if unsigned greater than or equal	1	1 (2)	None
	BRA <i>GT, Expr</i>	Branch if greater than	1	1 (2)	None
	BRA <i>GTU, Expr</i>	Branch if unsigned greater than	1	1 (2)	None
	BRA <i>LE, Expr</i>	Branch if less than or equal	1	1 (2)	None
	BRA <i>LEU, Expr</i>	Branch if unsigned less than or equal	1	1 (2)	None
	BRA <i>LT, Expr</i>	Branch if less than	1	1 (2)	None
	BRA <i>LTU, Expr</i>	Branch if unsigned less than	1	1 (2)	None
	BRA <i>N, Expr</i>	Branch if Negative	1	1 (2)	None
	BRA <i>NC, Expr</i>	Branch if Not Carry	1	1 (2)	None
	BRA <i>NN, Expr</i>	Branch if Not Negative	1	1 (2)	None
	BRA <i>NOV, Expr</i>	Branch if Not Overflow	1	1 (2)	None
	BRA <i>NZ, Expr</i>	Branch if Not Zero	1	1 (2)	None
	BRA <i>OA, Expr</i>	Branch if Accumulator A overflow	1	1 (2)	None
	BRA <i>OB, Expr</i>	Branch if Accumulator B overflow	1	1 (2)	None
	BRA <i>OV, Expr</i>	Branch if Overflow	1	1 (2)	None
	BRA <i>SA, Expr</i>	Branch if Accumulator A saturated	1	1 (2)	None
	BRA <i>SB, Expr</i>	Branch if Accumulator B saturated	1	1 (2)	None
	BRA <i>Expr</i>	Branch Unconditionally	1	2	None
BRA <i>Z, Expr</i>	Branch if Zero	1	1 (2)	None	
BRA <i>Wn</i>	Computed Branch	1	2	None	
BSET	BSET <i>f, #bit4</i>	Bit Set <i>f</i>	1	1	None
	BSET <i>Ws, #bit4</i>	Bit Set <i>Ws</i>	1	1	None
BSW	BSW.C <i>Ws, Wb</i>	Write C bit to $Ws<Wb>$	1	1	None
	BSW.Z <i>Ws, Wb</i>	Write Z bit to $Ws<Wb>$	1	1	None

**TABLE 24-2: INSTRUCTION SET OVERVIEW (CONTINUED)**

Assembly Mnemonic	Assembly Syntax	Description	# of Words	# of Cycles	Status Flags Affected
BTG	BTG f, #bit4	Bit Toggle f	1	1	None
	BTG Ws, #bit4	Bit Toggle Ws	1	1	None
BTSC	BTSC f, #bit4	Bit Test f, Skip if Clear	1	1 (2 or 3)	None
	BTSC Ws, #bit4	Bit Test Ws, Skip if Clear	1	1 (2 or 3)	None
BTSS	BTSS f, #bit4	Bit Test f, Skip if Set	1	1 (2 or 3)	None
	BTSS Ws, #bit4	Bit Test Ws, Skip if Set	1	1 (2 or 3)	None
BTST	BTST f, #bit4	Bit Test f	1	1	Z
	BTST.C Ws, #bit4	Bit Test Ws to C	1	1	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
BTSTS	BTSTS f, #bit4	Bit Test then Set f	1	1	Z
	BTSTS. Ws, #bit4 C	Bit Test Ws to C, then Set C	1	1	C
	BTSTS. Ws, #bit4 Z	Bit Test Ws to Z, then Set Z	1	1	Z
CALL	CALL lit23	Call subroutine	2	2	None
	CALL Wn	Call indirect subroutine	1	2	None
CLR	CLR f	f = 0x0000	1	1	None
	CLR WREG	WREG = 0x0000	1	1	None
	CLR Ws	Ws = 0x0000	1	1	None
	CLR Acc, Wx, Wxd, Wy, Wyd, AWB	Clear Accumulator	1	1	OA,OB,SA,SB
CLRWDT	CLRWDT	Clear Watchdog Timer	1	1	WDTO,Sleep
COM	COM f	f = $\bar{f}$	1	1	N,Z
	COM f, WREG	WREG = $\bar{f}$	1	1	N,Z
	COM Ws, Wd	Wd = $\overline{Ws}$	1	1	N,Z
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
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
CPB	CPB f	Compare f with WREG, with Borrow	1	1	C,DC,N,OV,Z
	CPB Wb, #lit5	Compare Wb with lit5, with Borrow	1	1	C,DC,N,OV,Z
	CPB Wb, Ws	Compare Wb with Ws, with Borrow (Wb – Ws – C)	1	1	C,DC,N,OV,Z
CPSEQ	CPSEQ Wb, Wn	Compare Wb with Wn, skip if =	1	1 (2 or 3)	None
CPSGT	CPSGT Wb, Wn	Compare Wb with Wn, skip if >	1	1 (2 or 3)	None
CPSLT	CPSLT Wb, Wn	Compare Wb with Wn, skip if <	1	1 (2 or 3)	None
CPSNE	CPSNE Wb, Wn	Compare Wb with Wn, skip if ≠	1	1 (2 or 3)	None
DAW	DAW Wn	Wn = decimal adjust Wn	1	1	C
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

# PIC24FJ16MC101/102

**TABLE 24-2: INSTRUCTION SET OVERVIEW (CONTINUED)**

Assembly Mnemonic	Assembly Syntax	Description	# of Words	# of Cycles	Status Flags Affected
DEC2	DEC2 f	$f = f - 2$	1	1	C,DC,N,OV,Z
	DEC2 f, WREG	WREG = $f - 2$	1	1	C,DC,N,OV,Z
	DEC2 Ws, Wd	Wd = $Ws - 2$	1	1	C,DC,N,OV,Z
DISI	DISI #lit14	Disable Interrupts for k instruction cycles	1	1	None
DIV	DIV.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
EXCH	EXCH Wns, Wnd	Swap Wns with Wnd	1	1	None
FBCL	FBCL Ws, Wnd	Find Bit Change from Left (MSb) Side	1	1	C
FF1L	FF1L Ws, Wnd	Find First One from Left (MSb) Side	1	1	C
FF1R	FF1R Ws, Wnd	Find First One from Right (LSb) Side	1	1	C
GOTO	GOTO Expr	Go to address	2	2	None
	GOTO Wn	Go to indirect	1	2	None
INC	INC f	$f = f + 1$	1	1	C,DC,N,OV,Z
	INC f, WREG	WREG = $f + 1$	1	1	C,DC,N,OV,Z
	INC Ws, Wd	Wd = $Ws + 1$	1	1	C,DC,N,OV,Z
INC2	INC2 f	$f = f + 2$	1	1	C,DC,N,OV,Z
	INC2 f, WREG	WREG = $f + 2$	1	1	C,DC,N,OV,Z
	INC2 Ws, Wd	Wd = $Ws + 2$	1	1	C,DC,N,OV,Z
IOR	IOR f	$f = f .IOR. WREG$	1	1	N,Z
	IOR f, WREG	WREG = $f .IOR. WREG$	1	1	N,Z
	IOR #lit10, Wn	Wd = $lit10 .IOR. Wd$	1	1	N,Z
	IOR Wb, Ws, Wd	Wd = $Wb .IOR. Ws$	1	1	N,Z
	IOR Wb, #lit5, Wd	Wd = $Wb .IOR. lit5$	1	1	N,Z
LAC	LAC Wso, #Slit4, Acc	Load Accumulator	1	1	OA,OB,OAB,SA,SB,SAB
LNK	LNK #lit14	Link Frame Pointer	1	1	None
LSR	LSR f	$f = \text{Logical Right Shift } f$	1	1	C,N,OV,Z
	LSR f, WREG	WREG = Logical Right Shift f	1	1	C,N,OV,Z
	LSR Ws, Wd	Wd = Logical Right Shift Ws	1	1	C,N,OV,Z
	LSR Wb, Wns, Wnd	Wnd = Logical Right Shift Wb by Wns	1	1	N,Z
	LSR Wb, #lit5, Wnd	Wnd = Logical Right Shift Wb by lit5	1	1	N,Z
MOV	MOV f, Wn	Move f to Wn	1	1	None
	MOV 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	

**TABLE 24-2: INSTRUCTION SET OVERVIEW (CONTINUED)**

Assembly Mnemonic	Assembly Syntax	Description	# of Words	# of Cycles	Status Flags Affected
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
NEG	NEG Acc	Negate Accumulator	1	1	OA,OB,OAB,SA,SB,SAB
	NEG f	$f = \bar{f} + 1$	1	1	C,DC,N,OV,Z
	NEG f, WREG	WREG = $\bar{f} + 1$	1	1	C,DC,N,OV,Z
	NEG Ws, Wd	$Wd = \bar{Ws} + 1$	1	1	C,DC,N,OV,Z
NOP	NOP	No Operation	1	1	None
	NOPR	No Operation	1	1	None
POP	POP f	Pop f from Top-of-Stack (TOS)	1	1	None
	POP Wdo	Pop from Top-of-Stack (TOS) to Wdo	1	1	None
	POP.D Wnd	Pop from Top-of-Stack (TOS) to W(nd):W(nd + 1)	1	2	None
	POP.S	Pop Shadow Registers	1	1	All
PUSH	PUSH f	Push f to Top-of-Stack (TOS)	1	1	None
	PUSH Wso	Push Wso to Top-of-Stack (TOS)	1	1	None
	PUSH.D Wns	Push W(ns):W(ns + 1) to Top-of-Stack (TOS)	1	2	None
	PUSH.S	Push Shadow Registers	1	1	None
PWRSVAV	PWRSVAV #lit1	Go into Sleep or Idle mode	1	1	WDTO,Sleep
REPEAT	REPEAT #lit14	Repeat Next Instruction lit14 + 1 times	1	1	None
	REPEAT Wn	Repeat Next Instruction (Wn) + 1 times	1	1	None
RESET	RESET	Software device Reset	1	1	None
RETFIE	RETFIE	Return from interrupt	1	3 (2)	None
RETLW	RETLW #lit10, Wn	Return with literal in Wn	1	3 (2)	None
RETURN	RETURN	Return from Subroutine	1	3 (2)	None
RLC	RLC f	f = Rotate Left through Carry f	1	1	C,N,Z
	RLC f, WREG	WREG = Rotate Left through Carry f	1	1	C,N,Z
	RLC Ws, Wd	Wd = Rotate Left through Carry Ws	1	1	C,N,Z
RLNC	RLNC f	f = Rotate Left (No Carry) f	1	1	N,Z
	RLNC f, WREG	WREG = Rotate Left (No Carry) f	1	1	N,Z
	RLNC Ws, Wd	Wd = Rotate Left (No Carry) Ws	1	1	N,Z
RRC	RRC f	f = Rotate Right through Carry f	1	1	C,N,Z
	RRC f, WREG	WREG = Rotate Right through Carry f	1	1	C,N,Z
	RRC Ws, Wd	Wd = Rotate Right through Carry Ws	1	1	C,N,Z
RRNC	RRNC f	f = Rotate Right (No Carry) f	1	1	N,Z
	RRNC f, WREG	WREG = Rotate Right (No Carry) f	1	1	N,Z
	RRNC Ws, Wd	Wd = Rotate Right (No Carry) Ws	1	1	N,Z
SAC	SAC Acc, #Slit4, Wdo	Store Accumulator	1	1	None
	SAC.R Acc, #Slit4, Wdo	Store Rounded Accumulator	1	1	None

# PIC24FJ16MC101/102

**TABLE 24-2: INSTRUCTION SET OVERVIEW (CONTINUED)**

Assembly Mnemonic	Assembly Syntax	Description	# of Words	# of Cycles	Status Flags Affected
SE	SE Ws, Wnd	Wnd = sign-extended Ws	1	1	C,N,Z
SETM	SETM f	f = 0xFFFF	1	1	None
	SETM WREG	WREG = 0xFFFF	1	1	None
	SETM Ws	Ws = 0xFFFF	1	1	None
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
SL	SL f	f = Left Shift f	1	1	C,N,OV,Z
	SL f, WREG	WREG = Left Shift f	1	1	C,N,OV,Z
	SL Ws, Wd	Wd = Left Shift Ws	1	1	C,N,OV,Z
	SL Wb, Wns, Wnd	Wnd = Left Shift Wb by Wns	1	1	N,Z
	SL Wb, #lit5, Wnd	Wnd = Left Shift Wb by lit5	1	1	N,Z
SUB	SUB 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
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
SUBR	SUBR f	f = WREG - f	1	1	C,DC,N,OV,Z
	SUBR f, WREG	WREG = WREG - f	1	1	C,DC,N,OV,Z
	SUBR Wb, Ws, Wd	Wd = Ws - Wb	1	1	C,DC,N,OV,Z
	SUBR Wb, #lit5, Wd	Wd = lit5 - Wb	1	1	C,DC,N,OV,Z
SUBBR	SUBBR f	f = WREG - f - ( $\overline{C}$ )	1	1	C,DC,N,OV,Z
	SUBBR f, WREG	WREG = WREG - f - ( $\overline{C}$ )	1	1	C,DC,N,OV,Z
	SUBBR Wb, Ws, Wd	Wd = Ws - Wb - ( $\overline{C}$ )	1	1	C,DC,N,OV,Z
	SUBBR Wb, #lit5, Wd	Wd = lit5 - Wb - ( $\overline{C}$ )	1	1	C,DC,N,OV,Z
SWAP	SWAP.b Wn	Wn = nibble swap Wn	1	1	None
	SWAP Wn	Wn = byte swap Wn	1	1	None
TBLRDH	TBLRDH Ws, Wd	Read Prog<23:16> to Wd<7:0>	1	2	None
TBLRDL	TBLRDL Ws, Wd	Read Prog<15:0> to Wd	1	2	None
TBLWTH	TBLWTH Ws, Wd	Write Ws<7:0> to Prog<23:16>	1	2	None
TBLWTL	TBLWTL Ws, Wd	Write Ws to Prog<15:0>	1	2	None
ULNK	ULNK	Unlink Frame Pointer	1	1	None
XOR	XOR f	f = f .XOR. WREG	1	1	N,Z
	XOR f, WREG	WREG = f .XOR. WREG	1	1	N,Z
	XOR #lit10, Wn	Wd = lit10 .XOR. Wd	1	1	N,Z
	XOR Wb, Ws, Wd	Wd = Wb .XOR. Ws	1	1	N,Z
	XOR Wb, #lit5, Wd	Wd = Wb .XOR. lit5	1	1	N,Z
ZE	ZE Ws, Wnd	Wnd = Zero-extend Ws	1	1	C,Z,N

## 25.0 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

## 25.1 MPLAB Integrated Development Environment Software

The MPLAB IDE software brings an ease of software development previously unseen in the 8/16/32-bit microcontroller market. The MPLAB IDE is a Windows® operating system-based application that contains:

- A single graphical interface to all debugging tools
  - Simulator
  - Programmer (sold separately)
  - In-Circuit Emulator (sold separately)
  - In-Circuit Debugger (sold separately)
- A full-featured editor with color-coded context
- A multiple project manager
- Customizable data windows with direct edit of contents
- High-level source code debugging
- Mouse over variable inspection
- Drag and drop variables from source to watch windows
- Extensive on-line help
- Integration of select third party tools, such as IAR C Compilers

The MPLAB IDE allows you to:

- Edit your source files (either C or assembly)
- One-touch compile or assemble, and download to emulator and simulator tools (automatically updates all project information)
- Debug using:
  - Source files (C or assembly)
  - Mixed C and assembly
  - Machine code

MPLAB IDE supports multiple debugging tools in a single development paradigm, from the cost-effective simulators, through low-cost in-circuit debuggers, to full-featured emulators. This eliminates the learning curve when upgrading to tools with increased flexibility and power.

## 25.2 MPLAB C Compilers for Various Device Families

The MPLAB C Compiler code development systems are complete ANSI C compilers for Microchip's PIC18, PIC24 and PIC32 families of microcontrollers and the dsPIC30 and dsPIC33 families of digital signal controllers. These compilers provide powerful integration capabilities, superior code optimization and ease of use.

For easy source level debugging, the compilers provide symbol information that is optimized to the MPLAB IDE debugger.

## 25.3 HI-TECH C for Various Device Families

The HI-TECH C Compiler code development systems are complete ANSI C compilers for Microchip's PIC family of microcontrollers and the dsPIC family of digital signal controllers. These compilers provide powerful integration capabilities, omniscient code generation and ease of use.

For easy source level debugging, the compilers provide symbol information that is optimized to the MPLAB IDE debugger.

The compilers include a macro assembler, linker, pre-processor, and one-step driver, and can run on multiple platforms.

## 25.4 MPASM Assembler

The MPASM Assembler is a full-featured, universal macro assembler for PIC10/12/16/18 MCUs.

The MPASM Assembler generates relocatable object files for the MPLINK Object Linker, Intel® standard HEX files, MAP files to detail memory usage and symbol reference, absolute LST files that contain source lines and generated machine code and COFF files for debugging.

The MPASM Assembler features include:

- Integration into MPLAB IDE projects
- User-defined macros to streamline assembly code
- Conditional assembly for multi-purpose source files
- Directives that allow complete control over the assembly process

## 25.5 MPLINK Object Linker/ MPLIB Object Librarian

The MPLINK Object Linker combines relocatable objects created by the MPASM Assembler and the MPLAB C18 C Compiler. It can link relocatable objects from precompiled libraries, using directives from a linker script.

The MPLIB Object Librarian manages the creation and modification of library files of precompiled code. When a routine from a library is called from a source file, only the modules that contain that routine will be linked in with the application. This allows large libraries to be used efficiently in many different applications.

The object linker/library features include:

- Efficient linking of single libraries instead of many smaller files
- Enhanced code maintainability by grouping related modules together
- Flexible creation of libraries with easy module listing, replacement, deletion and extraction

## 25.6 MPLAB Assembler, Linker and Librarian for Various Device Families

MPLAB Assembler produces relocatable machine code from symbolic assembly language for PIC24, PIC32 and dsPIC devices. MPLAB C Compiler uses the assembler to produce its object file. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. Notable features of the assembler include:

- Support for the entire device instruction set
- Support for fixed-point and floating-point data
- Command line interface
- Rich directive set
- Flexible macro language
- MPLAB IDE compatibility



## 25.7 MPLAB SIM Software Simulator

The MPLAB SIM Software Simulator allows code development in a PC-hosted environment by simulating the PIC MCUs and dsPIC® DSCs on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a comprehensive stimulus controller. Registers can be logged to files for further run-time analysis. The trace buffer and logic analyzer display extend the power of the simulator to record and track program execution, actions on I/O, most peripherals and internal registers.

The MPLAB SIM Software Simulator fully supports symbolic debugging using the MPLAB C Compilers, and the MPASM and MPLAB Assemblers. The software simulator offers the flexibility to develop and debug code outside of the hardware laboratory environment, making it an excellent, economical software development tool.

## 25.8 MPLAB REAL ICE In-Circuit Emulator System

MPLAB REAL ICE In-Circuit Emulator System is Microchip's next generation high-speed emulator for Microchip Flash DSC and MCU devices. It debugs and programs PIC® 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.

## 25.9 MPLAB ICD 3 In-Circuit Debugger System

MPLAB ICD 3 In-Circuit Debugger System is Microchip's most cost effective high-speed hardware debugger/programmer for Microchip Flash Digital Signal Controller (DSC) and microcontroller (MCU) devices. It debugs and programs PIC® Flash microcontrollers and dsPIC® DSCs with the powerful, yet easy-to-use graphical user interface of MPLAB Integrated Development Environment (IDE).

The MPLAB ICD 3 In-Circuit Debugger probe is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with a connector compatible with the MPLAB ICD 2 or MPLAB REAL ICE systems (RJ-11). MPLAB ICD 3 supports all MPLAB ICD 2 headers.

## 25.10 PICkit 3 In-Circuit Debugger/Programmer and PICkit 3 Debug Express

The MPLAB PICkit 3 allows debugging and programming of PIC® 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 a Microchip debug (RJ-11) connector (compatible with MPLAB ICD 3 and MPLAB REAL ICE). The connector uses two device I/O pins and the reset line to implement in-circuit debugging and In-Circuit Serial Programming™.

The PICkit 3 Debug Express include the PICkit 3, demo board and microcontroller, hookup cables and CDROM with user's guide, lessons, tutorial, compiler and MPLAB IDE software.

## 25.11 PICkit 2 Development Programmer/Debugger and PICkit 2 Debug Express

The PICkit™ 2 Development Programmer/Debugger is a low-cost development tool with an easy to use interface for programming and debugging Microchip's Flash families of microcontrollers. The full featured Windows® programming interface supports baseline (PIC10F, PIC12F5xx, PIC16F5xx), midrange (PIC12F6xx, PIC16F), PIC18F, PIC24, dsPIC30, dsPIC33, and PIC32 families of 8-bit, 16-bit, and 32-bit microcontrollers, and many Microchip Serial EEPROM products. With Microchip's powerful MPLAB Integrated Development Environment (IDE) the PICkit™ 2 enables in-circuit debugging on most PIC® microcontrollers. In-Circuit-Debugging runs, halts and single steps the program while the PIC microcontroller is embedded in the application. When halted at a breakpoint, the file registers can be examined and modified.

The PICkit 2 Debug Express include the PICkit 2, demo board and microcontroller, hookup cables and CDROM with user's guide, lessons, tutorial, compiler and MPLAB IDE software.

## 25.12 MPLAB PM3 Device Programmer

The MPLAB PM3 Device Programmer is a universal, CE compliant device programmer with programmable voltage verification at VDDMIN and VDDMAX for maximum reliability. It features a large LCD display (128 x 64) for menus and error messages and a modular, detachable socket assembly to support various package types. The ICSP™ cable assembly is included as a standard item. In Stand-Alone mode, the MPLAB PM3 Device Programmer can read, verify and program PIC devices without a PC connection. It can also set code protection in this mode. The MPLAB PM3 connects to the host PC via an RS-232 or USB cable. The MPLAB PM3 has high-speed communications and optimized algorithms for quick programming of large memory devices and incorporates an MMC card for file storage and data applications.

## 25.13 Demonstration/Development Boards, Evaluation Kits, and Starter Kits

A wide variety of demonstration, development and evaluation boards for various PIC MCUs and dsPIC DSCs allows quick application development on fully functional systems. Most boards include prototyping areas for adding custom circuitry and provide application firmware and source code for examination and modification.

The boards support a variety of features, including LEDs, temperature sensors, switches, speakers, RS-232 interfaces, LCD displays, potentiometers and additional EEPROM memory.

The demonstration and development boards can be used in teaching environments, for prototyping custom circuits and for learning about various microcontroller applications.

In addition to the PICDEM™ and dsPICDEM™ demonstration/development board series of circuits, Microchip has a line of evaluation kits and demonstration software for analog filter design, KEELOQ® security ICs, CAN, IrDA®, PowerSmart battery management, SEEVAL® evaluation system, Sigma-Delta ADC, flow rate sensing, plus many more.

Also available are starter kits that contain everything needed to experience the specified device. This usually includes a single application and debug capability, all on one board.

Check the Microchip web page ([www.microchip.com](http://www.microchip.com)) for the complete list of demonstration, development and evaluation kits.

## 26.0 ELECTRICAL CHARACTERISTICS

**Note:** It is important to note that the specifications in this chapter of the data sheet, supercede any specifications that may be provided in PIC24F Family Reference Manual sections.

This section provides an overview of PIC24FJ16MC101/102 electrical characteristics. Additional information will be provided in future revisions of this document as it becomes available.

Absolute maximum ratings for the PIC24FJ16MC101/102 family are listed below. Exposure to these maximum rating conditions for extended periods may affect device reliability. Functional operation of the device at these or any other conditions above the parameters indicated in the operation listings of this specification is not implied.

### Absolute Maximum Ratings<sup>(1)</sup>

Ambient temperature under bias .....	-40°C to +125°C
Storage temperature .....	-65°C to +150°C
Voltage on VDD with respect to VSS .....	-0.3V to +4.0V
Voltage on any pin that is not 5V tolerant with respect to VSS <sup>(4)</sup> .....	-0.3V to (VDD + 0.3V)
Voltage on any 5V tolerant pin with respect to VSS when VDD ≥ 3.0V <sup>(4)</sup> .....	-0.3V to +5.6V
Voltage on any 5V tolerant pin with respect to VSS when VDD < 3.0V <sup>(4)</sup> .....	-0.3V to 3.6V
Maximum current out of VSS pin .....	300 mA
Maximum current into VDD pin <sup>(2)</sup> .....	250 mA
Maximum output current sunk by any I/O pin <sup>(3)</sup> .....	8 mA
Maximum output current sourced by any I/O pin <sup>(3)</sup> .....	8 mA
Maximum current sunk by all ports .....	200 mA
Maximum current sourced by all ports <sup>(2)</sup> .....	200 mA

**Note 1:** Stresses above those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only, and functional operation of the device at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

**2:** Maximum allowable current is a function of device maximum power dissipation (see [Table 26-2](#)).

**3:** An exception is the OSCO pin, which is able to source 12 mA and sink 10 mA.

**4:** See the “[Pin Diagrams](#)” section for 5V tolerant pins.

# PIC24FJ16MC101/102

## 26.1 DC Characteristics

**TABLE 26-1: OPERATING MIPS VS. VOLTAGE**

Characteristic	VDD Range (in Volts)	Temp Range (in °C)	Max MIPS
			PIC24FJ16MC101/102
DC5	3.0-3.6V	-40°C to +85°C	16
	3.0-3.6V	-40°C to +125°C	16

**TABLE 26-2: THERMAL OPERATING CONDITIONS**

Rating	Symbol	Min	Typ	Max	Unit
Industrial Temperature Devices					
Operating Junction Temperature Range	TJ	-40	—	+125	°C
Operating Ambient Temperature Range	TA	-40	—	+85	°C
Extended Temperature Devices					
Operating Junction Temperature Range	TJ	-40	—	+140	°C
Operating Ambient Temperature Range	TA	-40	—	+125	°C
Power Dissipation: Internal chip power dissipation: $P_{INT} = V_{DD} \times (I_{DD} - \sum I_{OH})$ I/O Pin Power Dissipation: $I/O = \sum (\{V_{DD} - V_{OH}\} \times I_{OH}) + \sum (V_{OL} \times I_{OL})$	PD	PINT + P/I/O			W
Maximum Allowed Power Dissipation	PDMAX	$(T_J - T_A)/\theta_{JA}$			W

**TABLE 26-3: THERMAL PACKAGING CHARACTERISTICS**

Characteristic	Symbol	Typ	Max	Unit	Notes
Package Thermal Resistance, 18-pin PDIP	$\theta_{JA}$	50	—	°C/W	1
Package Thermal Resistance, 20-pin PDIP	$\theta_{JA}$	50	—	°C/W	1
Package Thermal Resistance, 28-pin SPDIP	$\theta_{JA}$	50	—	°C/W	1
Package Thermal Resistance, 18-pin SOIC	$\theta_{JA}$	63	—	°C/W	1
Package Thermal Resistance, 20-pin SOIC	$\theta_{JA}$	63	—	°C/W	1
Package Thermal Resistance, 28-pin SOIC	$\theta_{JA}$	55	—	°C/W	1
Package Thermal Resistance, 20-pin SSOP	$\theta_{JA}$	90	—	°C/W	1
Package Thermal Resistance, 28-pin SSOP	$\theta_{JA}$	71	—	°C/W	1
Package Thermal Resistance, 28-pin QFN (6x6 mm)	$\theta_{JA}$	37	—	°C/W	1
Package Thermal Resistance, 36-pin TLA (5x5 mm)	$\theta_{JA}$	31.1	—	°C/W	1

**Note 1:** Junction to ambient thermal resistance, Theta-JA ( $\theta_{JA}$ ) numbers are achieved by package simulations.

**TABLE 26-4: DC TEMPERATURE AND VOLTAGE SPECIFICATIONS**

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	Symbol	Characteristic	Min	Typ <sup>(1)</sup>	Max	Units	Conditions
<b>Operating Voltage</b>							
DC10	<b>Supply Voltage</b>						
	VDD	—	3.0	—	3.6	V	Industrial and Extended
DC12	VDR	<b>RAM Data Retention Voltage<sup>(2)</sup></b>	1.8	—	—	V	—
DC16	VPOR	<b>VDD Start Voltage</b> to ensure internal Power-on Reset signal	—	—	VSS	V	—
DC17	SVDD	<b>VDD Rise Rate</b> to ensure internal Power-on Reset signal	0.024	—	—	V/ms	0-2.4V in 0.1s
DC18	VCORE	<b>VDD Core<sup>(3)</sup></b> Internal regulator voltage	2.25	—	2.75	V	Voltage is dependent on load, temperature and VDD

- Note 1:** Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.  
**Note 2:** This is the limit to which VDD may be lowered without losing RAM data.  
**Note 3:** These parameters are characterized by similarity, but are not tested in manufacturing.

**TABLE 26-5: ELECTRICAL CHARACTERISTICS: BOR**

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 <sup>(1)</sup>	Typ	Max	Units	Conditions
BO10	VBOR	BOR Event on VDD transition	2.40	2.48	2.55	V	—

- Note 1:** Parameters are for design guidance only and are not tested in manufacturing.

# PIC24FJ16MC101/102

**TABLE 26-6: DC CHARACTERISTICS: OPERATING CURRENT (IDD)**

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended			
Parameter No.	Typical <sup>(1)</sup>	Max	Units	Conditions		
<b>Operating Current (IDD)<sup>(2)</sup></b>						
DC20d	0.7	1.7	mA	-40°C	3.3V	LPRC (31 kHz) <sup>(3)</sup>
DC20a	0.7	1.7	mA	+25°C		
DC20b	1.0	1.7	mA	+85°C		
DC20c	1.3	1.7	mA	+125°C		
DC21d	1.9	2.6	mA	-40°C	3.3V	1 MIPS <sup>(3)</sup>
DC21a	1.9	2.6	mA	+25°C		
DC21b	1.9	2.6	mA	+85°C		
DC21c	2.0	2.6	mA	+125°C		
DC22d	6.5	8.5	mA	-40°C	3.3V	4 MIPS <sup>(3)</sup>
DC22a	6.5	8.5	mA	+25°C		
DC22b	6.5	8.5	mA	+85°C		
DC22c	6.5	8.5	mA	+125°C		
DC23d	12.2	16	mA	-40°C	3.3V	10 MIPS <sup>(3)</sup>
DC23a	12.2	16	mA	+25°C		
DC23b	12.2	16	mA	+85°C		
DC23c	12.2	16	mA	+125°C		
DC24d	16	21	mA	-40°C	3.3V	16 MIPS
DC24a	16	21	mA	+25°C		
DC24b	16	21	mA	+85°C		
DC24c	16	21	mA	+125°C		

**Note 1:** Data in "Typical" column is at 3.3V, 25°C unless otherwise stated.

**2:** IDD is primarily a function of the operating voltage and frequency. Other factors, such as I/O pin loading and switching rate, oscillator type, internal code execution pattern and temperature, also have an impact on the current consumption. The test conditions for all IDD measurements are as follows:

- Oscillator is configured in EC mode, OSC1 is driven with external square wave from rail-to-rail
- CLKO is configured as an I/O input pin in the Configuration word
- All I/O pins are configured as inputs and pulled to VSS
- $\overline{\text{MCLR}} = \text{VDD}$ , 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 (PMDx bits are all zeroed)
- CPU executing `while(1)` statement

**3:** These parameters are characterized, but not tested in manufacturing.

**TABLE 26-7: DC CHARACTERISTICS: IDLE CURRENT (IDLE)**

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended		
Parameter No.	Typical <sup>(1)</sup>	Max	Units	Conditions	
<b>Idle Current (IDLE): Core OFF Clock ON Base Current<sup>(2)</sup></b>					
DC40d	0.6	1.6	mA	-40°C	3.3V LPRC (31 kHz) <sup>(3)</sup>
DC40a	0.6	1.6	mA	+25°C	
DC40b	0.9	1.6	mA	+85°C	
DC40c	1.2	1.6	mA	+125°C	
DC41d	0.5	1.1	mA	-40°C	3.3V 1 MIPS <sup>(3)</sup>
DC41a	0.5	1.1	mA	+25°C	
DC41b	0.5	1.1	mA	+85°C	
DC41c	0.8	1.1	mA	+125°C	3.3V 4 MIPS <sup>(3)</sup>
DC42d	0.9	1.6	mA	-40°C	
DC42a	0.9	1.6	mA	+25°C	
DC42b	1.0	1.6	mA	+85°C	
DC42c	1.2	1.6	mA	+125°C	3.3V 10 MIPS <sup>(3)</sup>
DC43a	1.6	2.6	mA	+25°C	
DC43d	1.6	2.6	mA	-40°C	
DC43b	1.7	2.6	mA	+85°C	
DC43c	2	2.6	mA	+125°C	3.3V 16 MIPS <sup>(3)</sup>
DC44d	2.4	3.8	mA	-40°C	
DC44a	2.4	3.8	mA	+25°C	
DC44b	2.6	3.8	mA	+85°C	
DC44c	2.9	3.8	mA	+125°C	

**Note 1:** Data in "Typical" column is at 3.3V, 25°C unless otherwise stated.

**2:** Base Idle current is measured as follows:

- CPU core is off, oscillator is configured in EC mode, OSC1 is driven with external square wave from rail-to-rail
- CLKO is configured as an I/O input pin in the Configuration word
- External Secondary Oscillator (Sosc) is disabled (i.e., SOSCO and SOSCI pins are configured as digital I/O inputs)
- All I/O pins are configured as inputs and pulled to VSS
- $\overline{\text{MCLR}} = \text{VDD}$ , WDT and FSCM are disabled
- No peripheral modules are operating; however, every peripheral is being clocked (PMDx bits are all zeroed)
- The VREGS bit (RCON<8>) = 1

**3:** These parameters are characterized, but not tested in manufacturing.

# PIC24FJ16MC101/102

**TABLE 26-8: DC CHARACTERISTICS: POWER-DOWN CURRENT (IPD)**

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended			
Parameter No.	Typical <sup>(1)</sup>	Max	Units	Conditions		
<b>Power-Down Current (IPD)<sup>(2)</sup></b>						
DC60d	27	250	μA	-40°C	3.3V	Base Power-Down Current <sup>(3,4)</sup>
DC60a	32	250	μA	+25°C		
DC60b	43	250	μA	+85°C		
DC60c	150	500	μA	+125°C		
DC61d	420	600	μA	-40°C	3.3V	Watchdog Timer Current: ΔI <sub>WDT</sub> <sup>(3,5)</sup>
DC61a	420	600	μA	+25°C		
DC61b	530	750	μA	+85°C		
DC61c	620	900	μA	+125°C		

**Note 1:** Data in the Typical column is at 3.3V, 25°C unless otherwise stated.

**2:** IPD (Sleep) current is measured as follows:

- CPU core is off, oscillator is configured in EC mode, OSC1 is driven with external square wave from rail-to-rail
- CLKO is configured as an I/O input pin in the Configuration word
- External Secondary Oscillator (Sosc) is disabled (i.e., SOSCO and SOSCI pins are configured as digital I/O inputs)
- All I/O pins are configured as inputs and pulled to V<sub>SS</sub>
- MCLR = V<sub>DD</sub>, WDT and FSCM are disabled
- All peripheral modules are disabled (PMDx bits are all ones)
- VREGS bit (RCON<8>) = 1 (i.e., core regulator is set to stand-by while the device is in Sleep mode)
- On applicable devices, RTCC is disabled plus the VREGS bit (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 IPD current.

**4:** These currents are measured on the device containing the most memory in this family.

**5:** These parameters are characterized, but not tested in manufacturing.



**TABLE 26-9: DC CHARACTERISTICS: DOZE CURRENT (IDOZE)**

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended				
Parameter No.	Typical <sup>(1)</sup>	Max	Doze Ratio <sup>(2)</sup>	Units	Conditions		
DC70a	13.2	17.2	1:2	mA	+25°C	3.3V	16 MIPS
DC70f	4.7	6.2	1:64	mA			
DC70g	4.7	6.2	1:128	mA			
DC71a	13.2	17.2	1:2	mA	+85°C	3.3V	16 MIPS
DC71f	4.7	6.2	1:64	mA			
DC71g	4.7	6.2	1:128	mA			
DC72a	13.2	17.2	1:2	mA	+125°C	3.3V	16 MIPS
DC72f	4.7	6.2	1:64	mA			
DC72g	4.7	6.2	1:128	mA			
DC73a	13.2	17.2	1:2	mA	-40°C	3.3V	16 MIPS
DC73f	4.7	6.2	1:64	mA			
DC73g	4.7	6.2	1:128	mA			

**Note 1:** Data in the Typical column is at 3.3V, 25°C unless otherwise stated.

**2:** IDOZE is primarily a function of the operating voltage and frequency. Other factors, such as I/O pin loading and switching rate, oscillator type, internal code execution pattern and temperature, also have an impact on the current consumption. The test conditions for all IDOZE measurements are as follows:

- Oscillator is configured in EC mode, OSC1 is driven with external square wave from rail-to-rail
- CLKO is configured as an I/O input pin in the Configuration word
- All I/O pins are configured as inputs and pulled to Vss
- MCLR = VDD, WDT and FSCM are disabled
- CPU, SRAM, program memory and data memory are operational
- No peripheral modules are operating; however, every peripheral is being clocked (PMDx bits are all zeroes)
- CPU executing `while(1)` statement

# PIC24FJ16MC101/102

**TABLE 26-10: DC CHARACTERISTICS: I/O PIN INPUT SPECIFICATIONS**

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended				
Param No.	Symbol	Characteristic	Min	Typ <sup>(1)</sup>	Max	Units	Conditions
	VIL	<b>Input Low Voltage</b>					
DI10		I/O pins	VSS	—	0.2 VDD	V	
DI15		MCLR	VSS	—	0.2 VDD	V	
DI16		I/O pins with OSC1 or SOSCI	VSS	—	0.2 VDD	V	
DI18		SDA, SCL	VSS	—	0.3 VDD	V	SMBus disabled
DI19		SDA, SCL	VSS	—	0.8	V	SMBus enabled
	VIH	<b>Input High Voltage</b>					
DI20		I/O pins not 5V tolerant <sup>(4)</sup>	0.7 VDD	—	VDD	V	
		I/O pins 5V tolerant <sup>(4)</sup>	0.7 VDD	—	5.5	V	
DI28		SDAx, SCLx	0.7 VDD	—	VDD	V	SMBus disabled
DI29		SDAx, SCLx	2.1	—	VDD	V	SMBus enabled
	ICNPU	<b>CNx Pull-up Current</b>					
DI30			50	250	400	μA	VDD = 3.3V, VPIN = VSS
	IIL	<b>Input Leakage Current<sup>(2,3)</sup></b>					
DI50a		MCLR pin	-2	—	+2	μA	VSS ≤ VPIN ≤ VDD, Pin at high-impedance
DI50b		All pins except MCLR and OSCO	-2	—	+2	μA	VSS ≤ VPIN ≤ VDD, Pin at high-impedance
DI50c		OSCO pin	-4	—	+4	μA	VSS ≤ VPIN ≤ VDD, Pin at high-impedance

**Note 1:** Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.

**2:** The leakage current on the MCLR pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.

**3:** Negative current is defined as current sourced by the pin.

**4:** See “Pin Diagrams” for a list of 5V tolerant pins.

**TABLE 26-11: DC CHARACTERISTICS: I/O PIN OUTPUT SPECIFICATIONS**

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended				
Param No.	Symbol	Characteristic	Min	Typ	Max	Units	Conditions
	VOL	<b>Output Low Voltage</b>					
DO10b		All I/O pins except OSCO	—	—	0.4	V	IOL = 8 mA, VDD = 3.3V
DO10c		OSCO pin	—	—	0.4	V	IOL = 10 mA, VDD = 3.3V
	VOH	<b>Output High Voltage</b>					
DO20b		All I/O pins except OSCO	2.4	—	—	V	IOL = -8 mA, VDD = 3.3V
DO20c		OSCO pin	2.4	—	—	V	IOL = -12 mA, VDD = 3.3V

**TABLE 26-12: DC CHARACTERISTICS: PROGRAM MEMORY**

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended				
Param No.	Symbol	Characteristic <sup>(3)</sup>	Min	Typ <sup>(1)</sup>	Max	Units	Conditions
<b>Program Flash Memory</b>							
D130a	EP	Cell Endurance	10,000	—	—	E/W	-40° C to +125° C
D131	VPR	VDD for Read	VMIN	—	3.6	V	VMIN = Minimum operating voltage
D132B	VPEW	VDD for Self-Timed Write	VMIN	—	3.6	V	VMIN = Minimum operating voltage
D134	TRETD	Characteristic Retention	20	—	—	Year	Provided no other specifications are violated
D135	IDDP	Supply Current during Programming	—	10	—	mA	
D137a	TPE	Page Erase Time	20.1	—	26.5	ms	TPE = 168517 FRC cycles, TA = +100°C, See <b>Note 2</b>
D137b	TPE	Page Erase Time	19.5	—	27.3	ms	TPE = 168517 FRC cycles, TA = +125°C, See <b>Note 2</b>
D138a	TWW	Word Write Cycle Time	47.9	—	48.8	µs	TWW = 355 FRC cycles, TA = +100°C, See <b>Note 2</b>
D138b	TWW	Word Write Cycle Time	47.4	—	49.3	µs	TWW = 355 FRC cycles, TA = +125°C, See <b>Note 2</b>

- Note 1:** Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.
- Note 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 26-18](#)) and the value of the FRC Oscillator Tuning register (see [Register 8-3](#)). For complete details on calculating the Minimum and Maximum time see [Section 5.3 “Programming Operations”](#).
- Note 3:** These parameters are ensured by design, but are not characterized or tested in manufacturing.

**TABLE 26-13: INTERNAL VOLTAGE REGULATOR 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	Characteristics	Min	Typ	Max	Units	Comments
—	CEFC	External Filter Capacitor Value <sup>(1)</sup>	4.7	10	—	µF	Capacitor must be low series resistance (< 5 ohms)

- Note 1:** Typical VCAP voltage = 2.5V when VDD ≥ VDDMIN.

# PIC24FJ16MC101/102

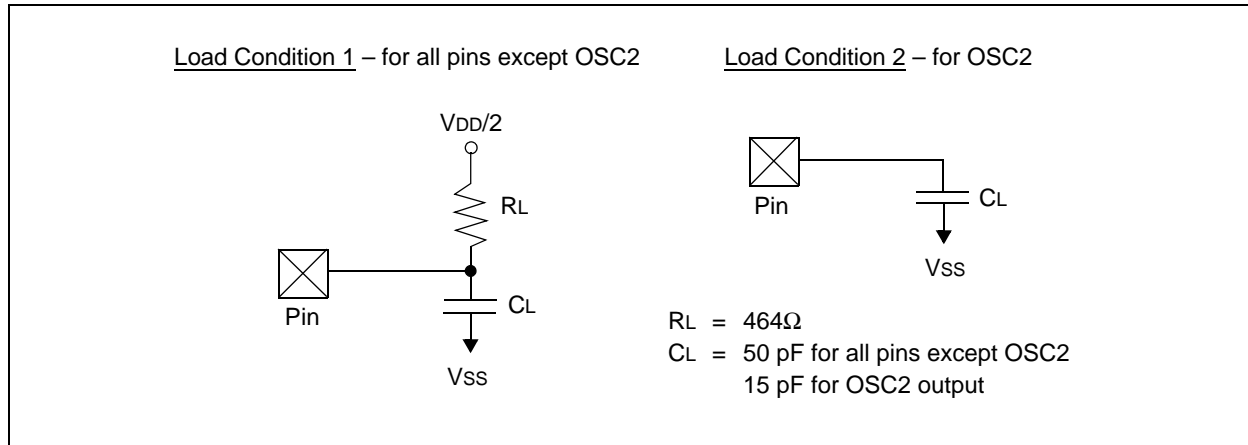
## 26.2 AC Characteristics and Timing Parameters

This section defines PIC24FJ16MC101/102 AC characteristics and timing parameters.

**TABLE 26-14: TEMPERATURE AND VOLTAGE SPECIFICATIONS – AC**

<b>AC CHARACTERISTICS</b>	<b>Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)</b>
	Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended Operating voltage $V_{DD}$ range as described in <a href="#">Section 26.1 “DC Characteristics”</a> .

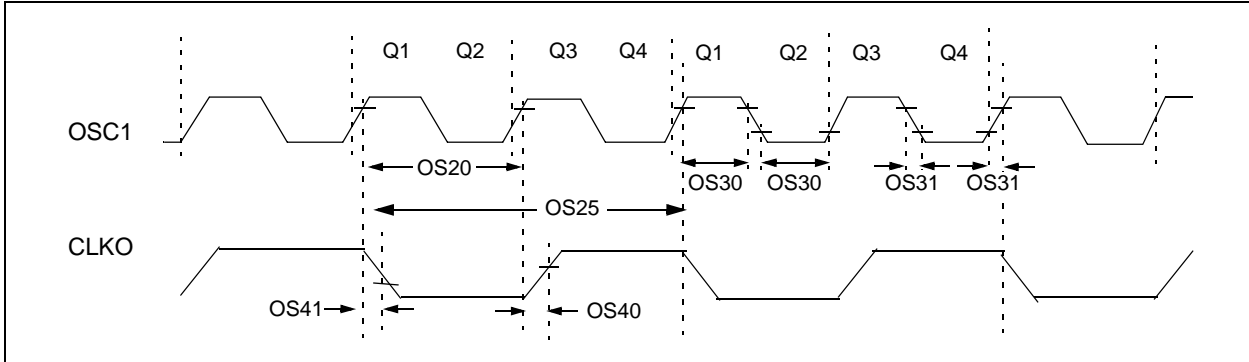
**FIGURE 26-1: LOAD CONDITIONS FOR DEVICE TIMING SPECIFICATIONS**



**TABLE 26-15: CAPACITIVE LOADING REQUIREMENTS ON OUTPUT PINS**

Param No.	Symbol	Characteristic	Min	Typ	Max	Units	Conditions
DO50	Cosc2	OSC2/SOSC2 pin	—	—	15	pF	In MS and HS modes when external clock is used to drive OSC1
DO56	Cio	All I/O pins and OSC2	—	—	50	pF	EC mode
DO58	CB	SCLx, SDAx	—	—	400	pF	In I <sup>2</sup> C™ mode

**FIGURE 26-2: EXTERNAL CLOCK TIMING**



**TABLE 26-16: EXTERNAL CLOCK TIMING REQUIREMENTS**

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	Symb	Characteristic	Min	Typ <sup>(1)</sup>	Max	Units	Conditions
OS10	FIN	External CLKI Frequency (External clocks allowed only in EC and ECPLL modes)	DC	—	32	MHz	EC
		Oscillator Crystal Frequency	3.0	—	10	MHz	MS
			10	—	32	MHz	HS
		31	—	33	kHz	SOSC	
OS20	Tosc	Tosc = 1/Fosc	31.25	—	DC	ns	—
OS25	Tcy	Instruction Cycle Time <sup>(2,4)</sup>	62.5	—	DC	ns	—
OS30	TosL, TosH	External Clock in (OSC1) <sup>(5)</sup> High or Low Time	0.45 x Tosc	—	—	ns	EC
OS31	TosR, TosF	External Clock in (OSC1) <sup>(5)</sup> Rise or Fall Time	—	—	20	ns	EC
OS40	TckR	CLKO Rise Time <sup>(3,5)</sup>	—	6	10	ns	—
OS41	TckF	CLKO Fall Time <sup>(3,5)</sup>	—	6	10	ns	—
OS42	GM	External Oscillator Transconductance <sup>(4)</sup>	14	16	18	mA/V	VDD = 3.3V TA = +25°C

**Note 1:** Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.

- 2:** Instruction cycle period (Tcy) equals two times the input oscillator time-base period. All specified values are based on characterization data for that particular oscillator type under standard operating conditions with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at “min.” values with an external clock applied to the OSC1/CLKI pin. When an external clock input is used, the “max.” cycle time limit is “DC” (no clock) for all devices.
- 3:** Measurements are taken in EC mode. The CLKO signal is measured on the OSC2 pin.
- 4:** These parameters are characterized by similarity, but are tested in manufacturing at FIN = 32 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 not tested in manufacturing.

# PIC24FJ16MC101/102

**TABLE 26-17: PLL CLOCK TIMING 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 <sup>(1)</sup>	Max	Units	Conditions
OS50	FPLLI	PLL Voltage Controlled Oscillator (VCO) Input Frequency Range <sup>(2)</sup>	3.0	—	8	MHz	ECPLL and MSPLL modes
OS51	Fsys	On-Chip VCO System Frequency <sup>(3)</sup>	12	—	32	MHz	—
OS52	TLOCK	PLL Start-up Time (Lock Time) <sup>(3)</sup>	—	—	2	ms	—
OS53	DCLK	CLKO Stability (Jitter) <sup>(3)</sup>	-2	1	+2	%	—

**Note 1:** Data in “Typ” column is at 3.3V, 25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

**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. The effective jitter for individual time bases or communication clocks used by the user application, are derived from dividing the CLKO stability specification by the square root of “N” (where “N” is equal to FOSC divided by the peripheral data rate clock). For example, if FOSC = 32 MHz and the SPI bit rate is 5 MHz, the effective jitter of the SPI clock is equal to:

$$\frac{DCLK}{\sqrt{\frac{32}{5}}} = \frac{2\%}{2.53} = 0.79\%$$

**TABLE 26-18: AC CHARACTERISTICS: INTERNAL FAST RC (FRC) ACCURACY**

AC CHARACTERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended				
Param No.	Characteristic	Min	Typ	Max	Units	Conditions
Internal FRC Accuracy @ 7.3728 MHz <sup>(1)</sup>						
F20a	FRC	1.5	±0.25	1.5	%	-40°C ≤TA ≤+85°C
F20b	FRC	-2	±0.25	+2	%	-40°C ≤TA ≤+125°C

**Note 1:** Frequency calibrated at 25°C and 3.3V. TUN bits may be used to compensate for temperature drift.

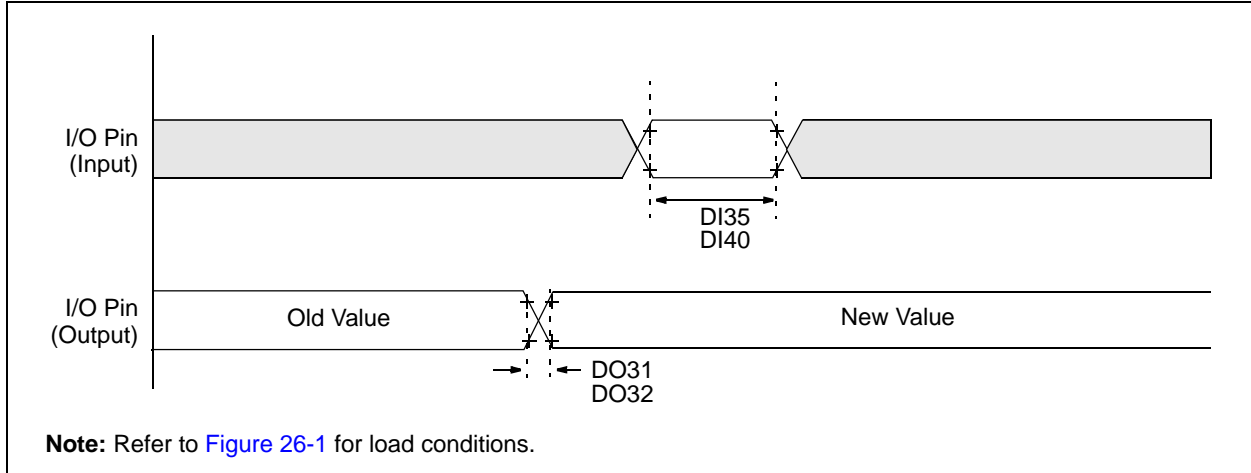
**TABLE 26-19: INTERNAL LOW-POWER RC (LPRC) ACCURACY**

AC CHARACTERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended				
Param No.	Characteristic	Min	Typ	Max	Units	Conditions
LPRC @ 32.768 kHz <sup>(1,2)</sup>						
F21a	LPRC	-20	±10	+20	%	-40°C ≤TA ≤+85°C
F21b	LPRC	-30	±10	+30	%	-40°C ≤TA ≤+125°C

**Note 1:** Change of LPRC frequency as VDD changes.

**2:** LPRC accuracy impacts the Watchdog Timer Time-out Period (TWDT1). See [Section 23.4 “Watchdog Timer \(WDT\)”](#) for more information.

**FIGURE 26-3: CLKO AND I/O TIMING CHARACTERISTICS**



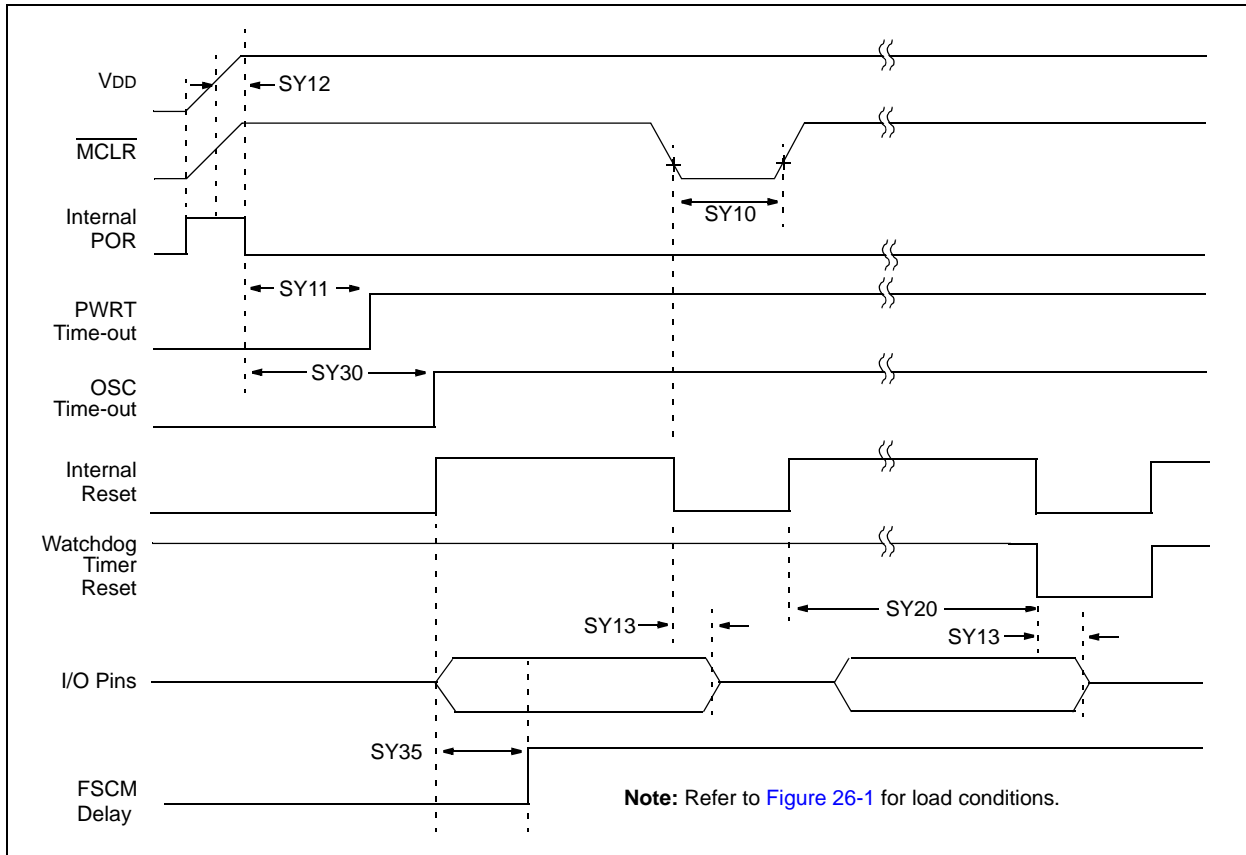
**TABLE 26-20: I/O TIMING REQUIREMENTS**

AC CHARACTERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended					
Param No.	Symbol	Characteristic <sup>(2)</sup>	Min	Typ <sup>(1)</sup>	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	—	—	T <sub>CY</sub>	—

- Note 1:** Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.  
**Note 2:** These parameters are characterized, but are not tested in manufacturing.

# PIC24FJ16MC101/102

**FIGURE 26-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING CHARACTERISTICS**



**TABLE 26-21: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING REQUIREMENTS**

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	Symbol	Characteristic <sup>(1)</sup>	Min	Typ <sup>(2)</sup>	Max	Units	Conditions
SY10	TMCL	MCLR Pulse Width (low)	2	—	—	μs	-40°C to +85°C
SY11	TPWRT	Power-up Timer Period <sup>(1)</sup>	—	64	—	ms	-40°C to +85°C
SY12	TPOR	Power-on Reset Delay <sup>(3)</sup>	3	10	30	μs	-40°C to +85°C
SY13	TIOZ	I/O High-Impedance from MCLR Low or Watchdog Timer Reset <sup>(1)</sup>	—	—	1.2	μs	—
SY20	TWDT1	Watchdog Timer Time-out Period <sup>(1)</sup>	—	—	—	ms	See Section 23.4 “Watchdog Timer (WDT)” and LPRC parameter F21a (Table 26-19).
SY30	TOST	Oscillator Start-up Time	—	1024 * TOSC	—	—	TOSC = OSC1 period
SY35	TFSCM	Fail-Safe Clock Monitor Delay <sup>(1)</sup>	—	500	900	μs	-40°C to +85°C

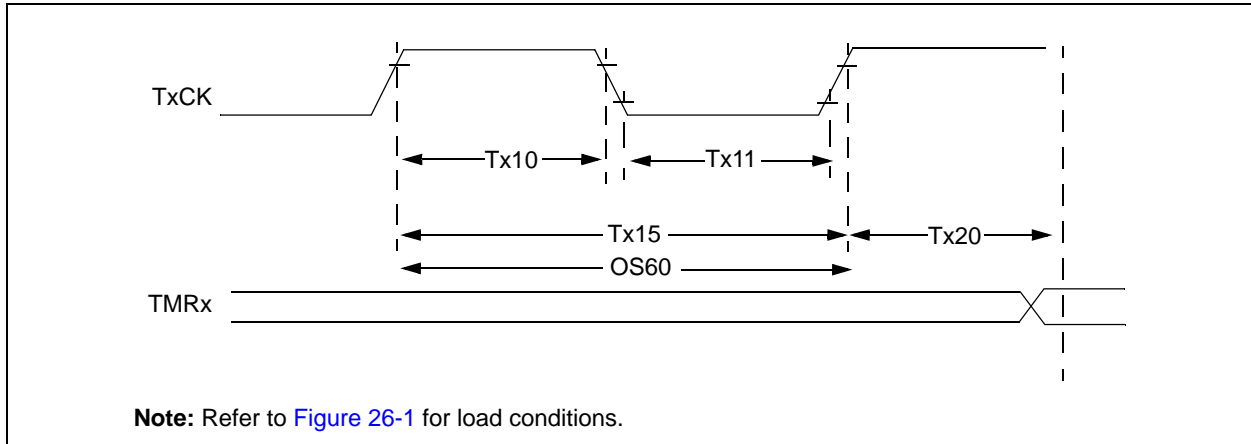
**Note 1:** These parameters are characterized but not tested in manufacturing.

**Note 2:** Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.

**Note 3:** These parameters are characterized, but are not tested in manufacturing.



**FIGURE 26-5: TIMER1, 2 AND 3 EXTERNAL CLOCK TIMING CHARACTERISTICS**



**Note:** Refer to Figure 26-1 for load conditions.

**TABLE 26-22: TIMER1 EXTERNAL CLOCK TIMING REQUIREMENTS<sup>(1)</sup>**

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended					
Param No.	Symbol	Characteristic <sup>(2)</sup>	Min	Typ	Max	Units	Conditions	
TA10	TtxH	TxCK High Time	Synchronous mode	Greater of: 20 or (Tcy + 20)/N	—	—	ns	Must also meet parameter TA15 N = prescaler value (1, 8, 64, 256)
		Asynchronous	35	—	—	ns		
TA11	TtxL	TxCK Low Time	Synchronous mode	Greater of: 20 ns or (Tcy + 20)/N	—	—	ns	Must also meet parameter TA15 N = prescaler value (1, 8, 64, 256)
		Asynchronous	10	—	—	ns		
TA15	TtxP	TxCK Input Period	Synchronous mode	Greater of: 40 or (2 Tcy + 40)/N	—	—	ns	N = prescale value (1, 8, 64, 256)
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	0.75 Tcy + 40	—	1.75 Tcy + 40	ns	—	

**Note 1:** Timer1 is a Type A.

**Note 2:** These parameters are characterized by similarity, but are not tested in manufacturing.

# PIC24FJ16MC101/102

**TABLE 26-23: 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 <sup>(1)</sup>		Min	Typ	Max	Units	Conditions
TB10	TtxH	TxCK High Time	Synchronous mode	Greater of: 20 or (Tcy + 20)/N	—	—	ns	Must also meet parameter TB15 N = prescale value (1, 8, 64, 256)
TB11	TtxL	TxCK Low Time	Synchronous mode	Greater of: 20 or (Tcy + 20)/N	—	—	ns	Must also meet parameter TB15 N = prescale value (1, 8, 64, 256)
TB15	TtxP	TxCK Input Period	Synchronous mode	Greater of: 40 or (2 Tcy + 40)/N	—	—	ns	N = prescale value (1, 8, 64, 256)
TB20	TCKEXTMRL	Delay from External TxCK Clock Edge to Timer Increment		0.75 Tcy + 40	—	1.75 Tcy + 40	ns	—

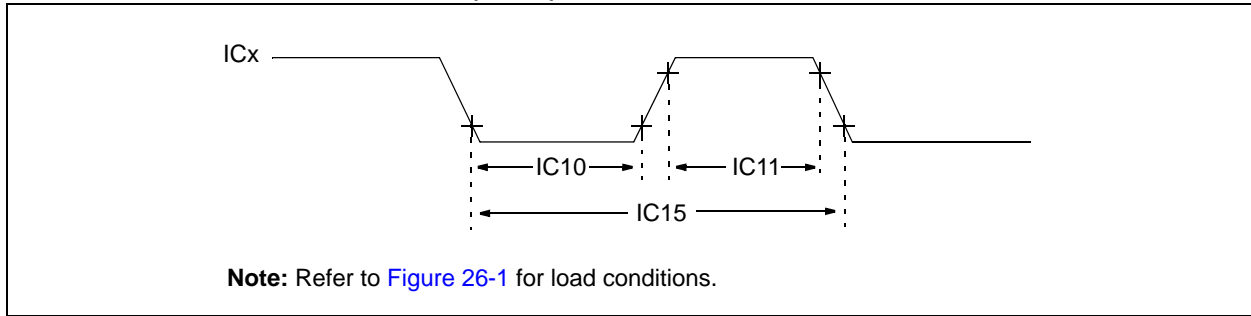
**Note 1:** These parameters are characterized, but are not tested in manufacturing.

**TABLE 26-24: 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 <sup>(1)</sup>		Min	Typ	Max	Units	Conditions
TC10	TtxH	TxCK High Time	Synchronous	Tcy + 20	—	—	ns	Must also meet parameter TC15
TC11	TtxL	TxCK Low Time	Synchronous	Tcy + 20	—	—	ns	Must also meet parameter TC15
TC15	TtxP	TxCK Input Period	Synchronous, with prescaler	2 Tcy + 40	—	—	ns	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 26-6: INPUT CAPTURE (CAPx) TIMING CHARACTERISTICS**



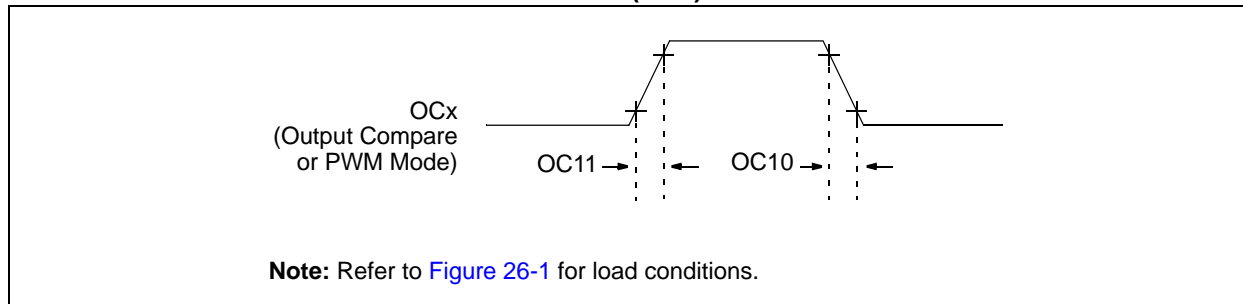
**TABLE 26-25: INPUT CAPTURE TIMING REQUIREMENTS**

AC CHARACTERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended					
Param No.	Symbol	Characteristic <sup>(1)</sup>		Min	Max	Units	Conditions
IC10	TccL	ICx Input Low Time	No Prescaler	$0.5 T_{CY} + 20$	—	ns	—
			With Prescaler	10	—	ns	
IC11	TccH	ICx Input High Time	No Prescaler	$0.5 T_{CY} + 20$	—	ns	—
			With Prescaler	10	—	ns	
IC15	TccP	ICx Input Period		$(T_{CY} + 40)/N$	—	ns	N = prescale value (1, 4, 16)

**Note 1:** These parameters are characterized by similarity, but are not tested in manufacturing.

# PIC24FJ16MC101/102

**FIGURE 26-7: OUTPUT COMPARE MODULE (OCx) TIMING CHARACTERISTICS**

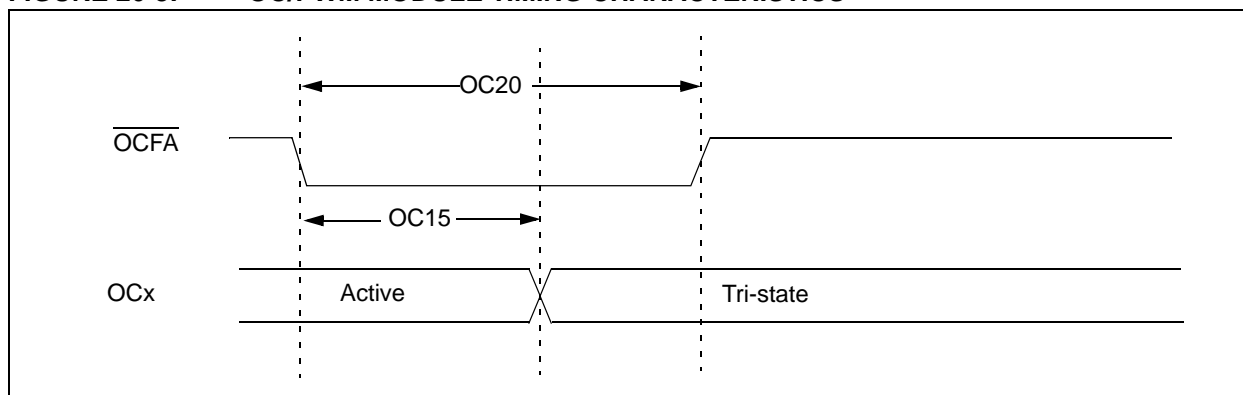


**TABLE 26-26: OUTPUT COMPARE MODULE TIMING REQUIREMENTS**

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 <sup>(1)</sup>	Min	Typ	Max	Units	Conditions
OC10	TccF	OCx Output Fall Time	—	—	—	ns	See parameter <a href="#">DO32</a>
OC11	TccR	OCx Output Rise Time	—	—	—	ns	See parameter <a href="#">DO31</a>

**Note 1:** These parameters are characterized by similarity, but are not tested in manufacturing.

**FIGURE 26-8: OC/PWM MODULE TIMING CHARACTERISTICS**

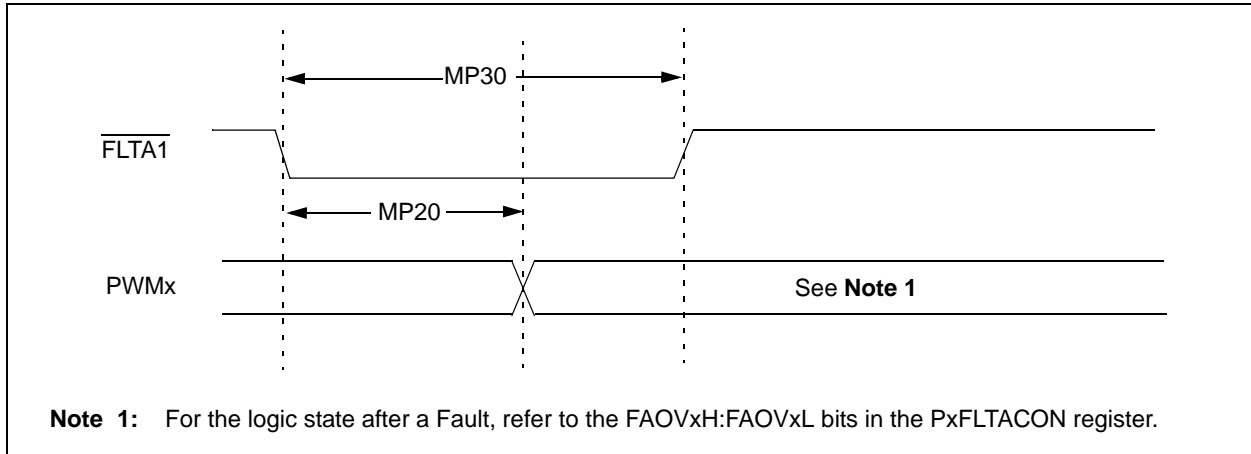


**TABLE 26-27: SIMPLE OC/PWM MODE TIMING REQUIREMENTS**

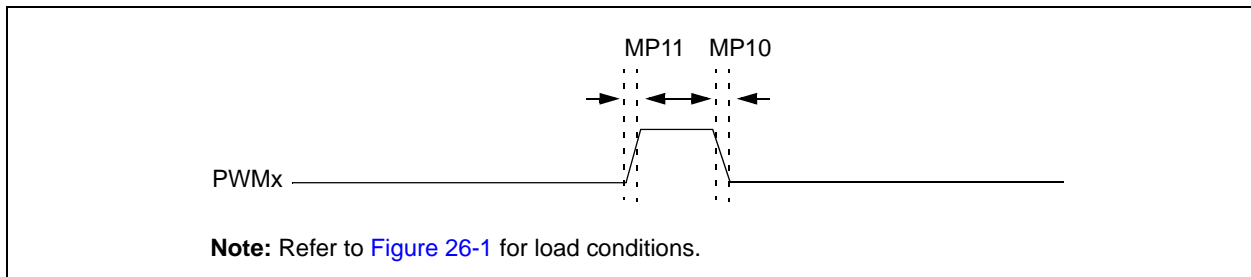
AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended				
Param No.	Symbol	Characteristic <sup>(1)</sup>	Min	Typ	Max	Units	Conditions
OC15	TFD	Fault Input to PWM I/O Change	—	—	T <sub>CY</sub> + 20 ns	ns	—
OC20	TFLT	Fault Input Pulse Width	T <sub>CY</sub> + 20 ns	—	—	ns	—

**Note 1:** These parameters are characterized by similarity, but are not tested in manufacturing.

**FIGURE 26-9: MOTOR CONTROL PWM MODULE FAULT TIMING CHARACTERISTICS**



**FIGURE 26-10: MOTOR CONTROL PWM MODULE TIMING CHARACTERISTICS**



**TABLE 26-28: MOTOR CONTROL PWM MODULE TIMING REQUIREMENTS**

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended				
Param No.	Symbol	Characteristic <sup>(1)</sup>	Min	Typ	Max	Units	Conditions
MP10	$T_{\text{FPWM}}$	PWM Output Fall Time	—	—	—	ns	See parameter <a href="#">DO32</a>
MP11	$T_{\text{RPWM}}$	PWM Output Rise Time	—	—	—	ns	See parameter <a href="#">DO31</a>
MP20	$T_{\text{FD}}$	Fault Input $\downarrow$ to PWM I/O Change	—	—	50	ns	—
MP30	$T_{\text{FH}}$	Minimum Pulse Width	50	—	—	ns	—

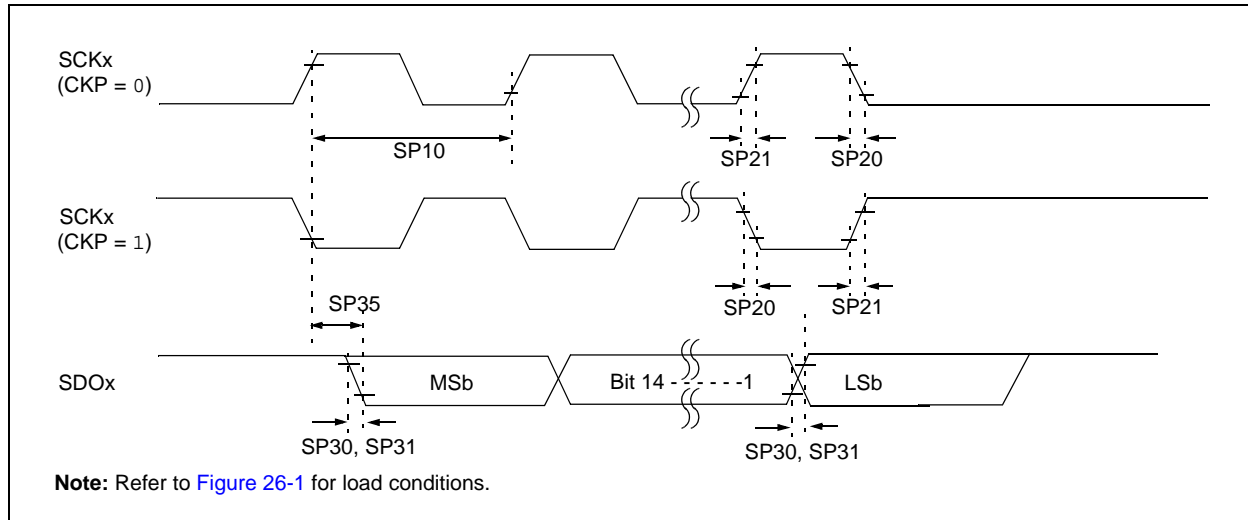
**Note 1:** These parameters are characterized by similarity, but are not tested in manufacturing.

# PIC24FJ16MC101/102

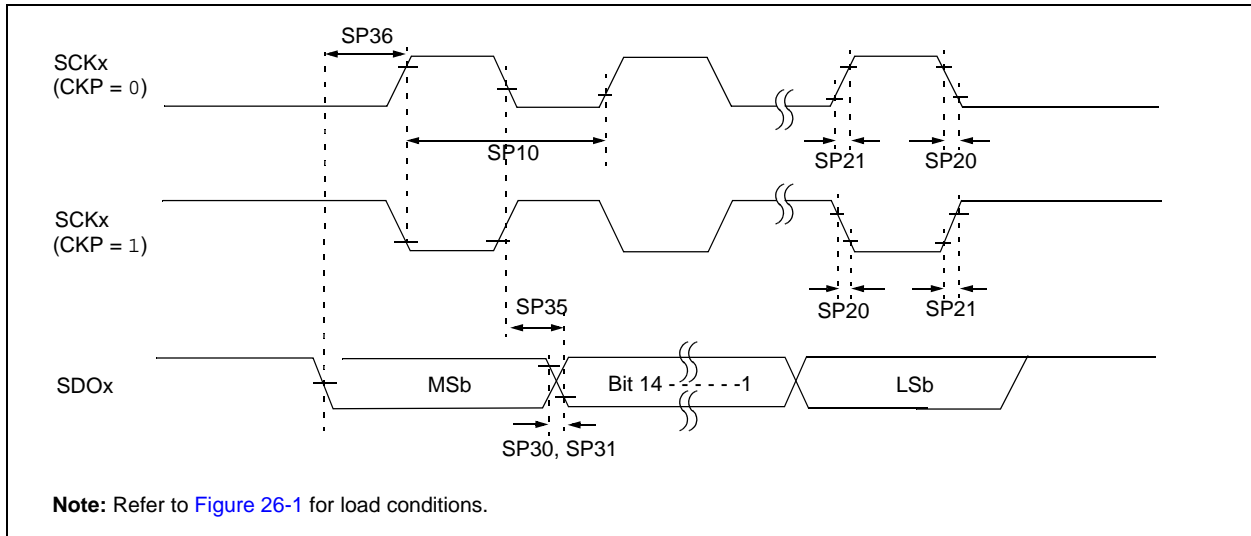
**TABLE 26-29: SPIx MAXIMUM DATA/CLOCK RATE SUMMARY**

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 26-30	—	—	0,1	0,1	0,1
10 MHz	—	Table 26-31	—	1	0,1	1
10 MHz	—	Table 26-32	—	0	0,1	1
15 MHz	—	—	Table 26-33	1	0	0
11 MHz	—	—	Table 26-34	1	1	0
15 MHz	—	—	Table 26-35	0	1	0
11 MHz	—	—	Table 26-36	0	0	0

**FIGURE 26-11: SPIx MASTER MODE (HALF-DUPLEX, TRANSMIT ONLY CKE = 0) TIMING CHARACTERISTICS**



**FIGURE 26-12: SPIx MASTER MODE (HALF-DUPLEX, TRANSMIT ONLY) TIMING CHARACTERISTICS**



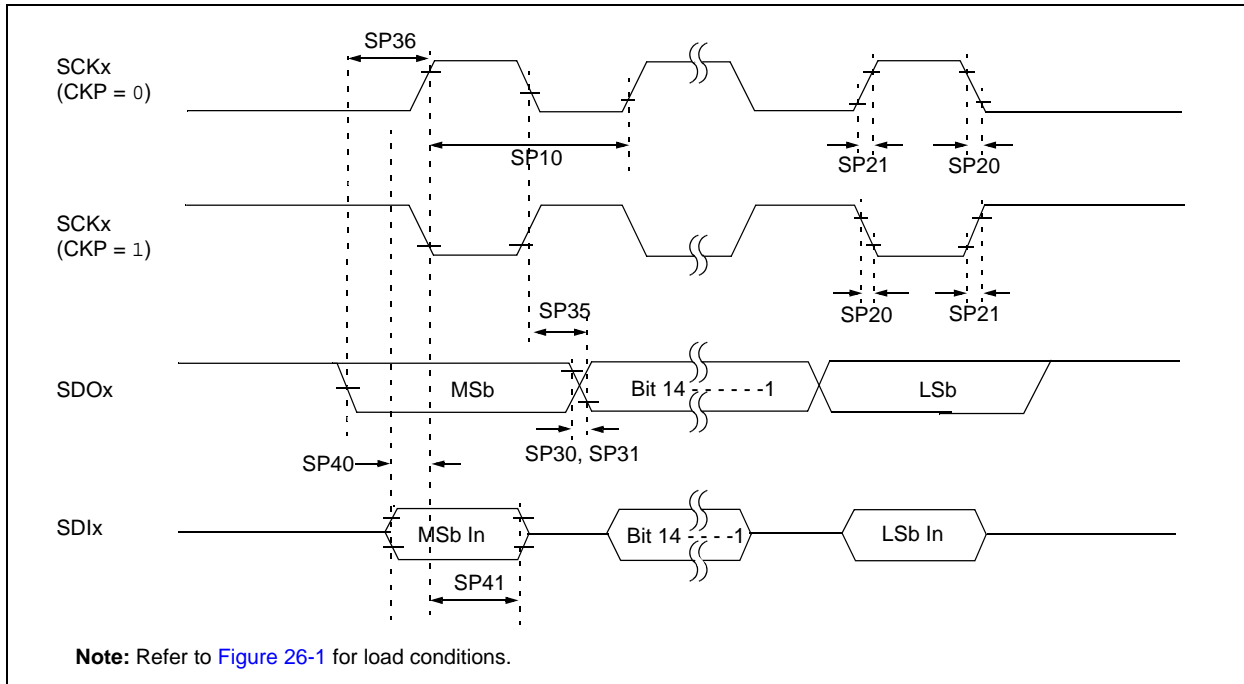
**TABLE 26-30: SPIx MASTER MODE (HALF-DUPLEX, TRANSMIT ONLY) TIMING REQUIREMENTS**

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	Symbol	Characteristic <sup>(1)</sup>	Min	Typ <sup>(2)</sup>	Max	Units	Conditions
SP10	TscP	Maximum SCK Frequency	—	—	15	MHz	See <a href="#">Note 3</a>
SP20	TscF	SCKx Output Fall Time	—	—	—	ns	See parameter <a href="#">DO32</a> and <a href="#">Note 4</a>
SP21	TscR	SCKx Output Rise Time	—	—	—	ns	See parameter <a href="#">DO31</a> and <a href="#">Note 4</a>
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See parameter <a href="#">DO32</a> and <a href="#">Note 4</a>
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See parameter <a href="#">DO31</a> and <a href="#">Note 4</a>
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	6	20	ns	—
SP36	TdiV2sch, TdiV2scl	SDOx Data Output Setup to First SCKx Edge	30	—	—	ns	—

- Note 1:** These parameters are characterized, but are not tested in manufacturing.
- Note 2:** Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.
- Note 3:** The minimum clock period for SCKx is 66.7 ns. Therefore, the clock generated in Master mode must not violate this specification.
- Note 4:** Assumes 50 pF load on all SPIx pins.

# PIC24FJ16MC101/102

**FIGURE 26-13: SPIx MASTER MODE (FULL-DUPLEX, CKE = 1, CKP = x, SMP = 1) TIMING CHARACTERISTICS**



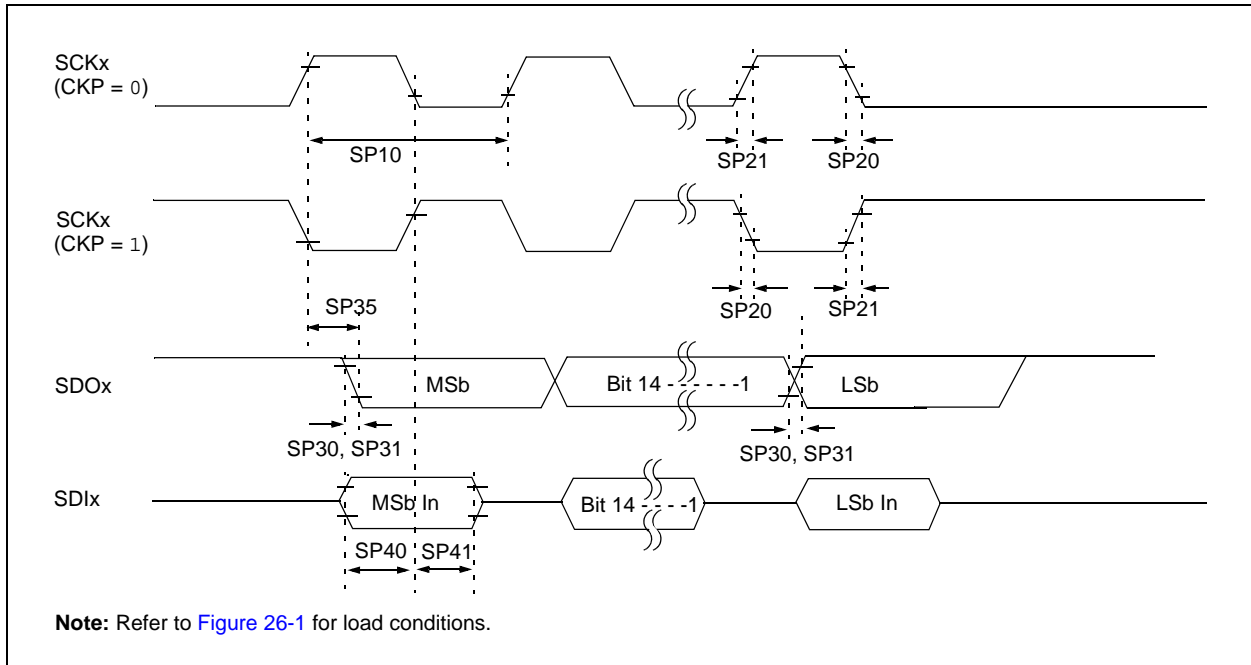
**TABLE 26-31: SPIx MASTER MODE (FULL-DUPLEX, CKE = 1, CKP = x, SMP = 1) TIMING REQUIREMENTS**

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 <sup>(1)</sup>	Min	Typ <sup>(2)</sup>	Max	Units	Conditions
SP10	TscP	Maximum SCK Frequency	—	—	10	MHz	See <b>Note 3</b>
SP20	TscF	SCKx Output Fall Time	—	—	—	ns	See parameter <b>DO32</b> and <b>Note 4</b>
SP21	TscR	SCKx Output Rise Time	—	—	—	ns	See parameter <b>DO31</b> and <b>Note 4</b>
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See parameter <b>DO32</b> and <b>Note 4</b>
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See parameter <b>DO31</b> and <b>Note 4</b>
SP35	Tsch2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	6	20	ns	—
SP36	TdoV2sc, TdoV2scL	SDOx Data Output Setup to First SCKx Edge	30	—	—	ns	—
SP40	TdiV2sch, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	30	—	—	ns	—
SP41	Tsch2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	30	—	—	ns	—

- Note 1:** These parameters are characterized, but are not tested in manufacturing.  
**Note 2:** Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.  
**Note 3:** The minimum clock period for SCKx is 100 ns. The clock generated in Master mode must not violate this specification.  
**Note 4:** Assumes 50 pF load on all SPIx pins.



**FIGURE 26-14: SPIx MASTER MODE (FULL-DUPLEX, CKE = 0, CKP = x, SMP = 1) TIMING CHARACTERISTICS**



**TABLE 26-32: SPIx MASTER MODE (FULL-DUPLEX, CKE = 0, CKP = x, SMP = 1) TIMING REQUIREMENTS**

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended				
Param No.	Symbol	Characteristic <sup>(1)</sup>	Min	Typ <sup>(2)</sup>	Max	Units	Conditions
SP10	TscP	Maximum SCK Frequency	—	—	10	MHz	-40°C to +125°C and see <b>Note 3</b>
SP20	TscF	SCKx Output Fall Time	—	—	—	ns	See parameter <b>DO32</b> and <b>Note 4</b>
SP21	TscR	SCKx Output Rise Time	—	—	—	ns	See parameter <b>DO31</b> and <b>Note 4</b>
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See parameter <b>DO32</b> and <b>Note 4</b>
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See parameter <b>DO31</b> and <b>Note 4</b>
SP35	Tsch2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	6	20	ns	—
SP36	TdoV2scH, TdoV2scL	SDOx Data Output Setup to First SCKx Edge	30	—	—	ns	—
SP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	30	—	—	ns	—
SP41	Tsch2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	30	—	—	ns	—

**Note 1:** These parameters are characterized, but are not tested in manufacturing.

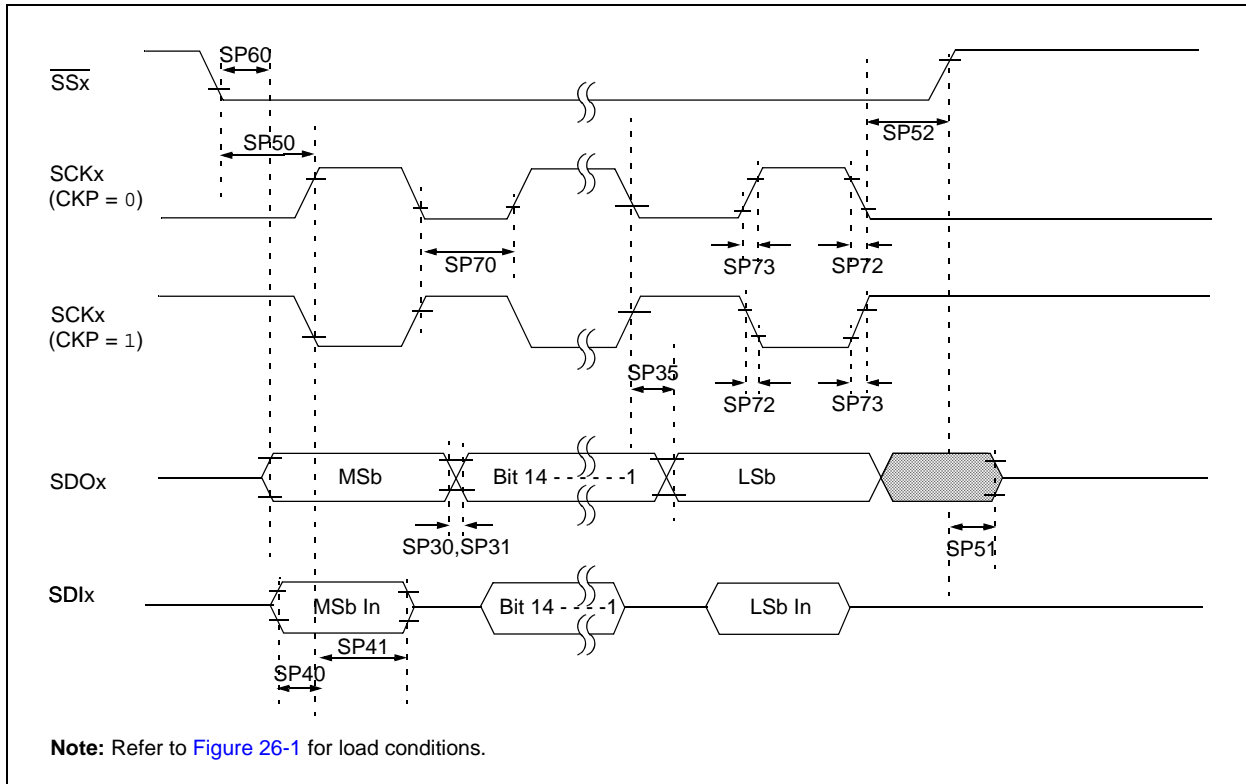
**Note 2:** Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.

**Note 3:** The minimum clock period for SCKx is 100 ns. The clock generated in Master mode must not violate this specification.

**Note 4:** Assumes 50 pF load on all SPIx pins.

# PIC24FJ16MC101/102

**FIGURE 26-15: SPIx SLAVE MODE (FULL-DUPLEX, CKE = 1, CKP = 0, SMP = 0) TIMING CHARACTERISTICS**



**TABLE 26-33: SPIx SLAVE MODE (FULL-DUPLEX, CKE = 1, CKP = 0, SMP = 0) TIMING REQUIREMENTS**

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	Symbol	Characteristic <sup>(1)</sup>	Min	Typ <sup>(2)</sup>	Max	Units	Conditions
SP70	TscP	Maximum SCK Input Frequency	—	—	15	MHz	See <b>Note 3</b>
SP72	TscF	SCKx Input Fall Time	—	—	—	ns	See parameter <a href="#">DO32</a> and <b>Note 4</b>
SP73	TscR	SCKx Input Rise Time	—	—	—	ns	See parameter <a href="#">DO31</a> and <b>Note 4</b>
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See parameter <a href="#">DO32</a> and <b>Note 4</b>
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See parameter <a href="#">DO31</a> and <b>Note 4</b>
SP35	Tsch2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	6	20	ns	—
SP36	TdoV2scH, TdoV2scL	SDOx Data Output Setup to First SCKx Edge	30	—	—	ns	—
SP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	30	—	—	ns	—
SP41	Tsch2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	30	—	—	ns	—
SP50	TssL2scH, TssL2scL	$\overline{SSx} \downarrow$ to SCKx $\uparrow$ or SCKx Input	120	—	—	ns	—
SP51	TssH2doZ	$\overline{SSx} \uparrow$ to SDOx Output High-Impedance <sup>(4)</sup>	10	—	50	ns	—
SP52	Tsch2ssH TscL2ssH	$\overline{SSx}$ after SCKx Edge	1.5 TCY + 40	—	—	ns	See <b>Note 4</b>
SP60	TssL2doV	SDOx Data Output Valid after $\overline{SSx}$ Edge	—	—	50	ns	—

**Note 1:** These parameters are characterized, but are not tested in manufacturing.

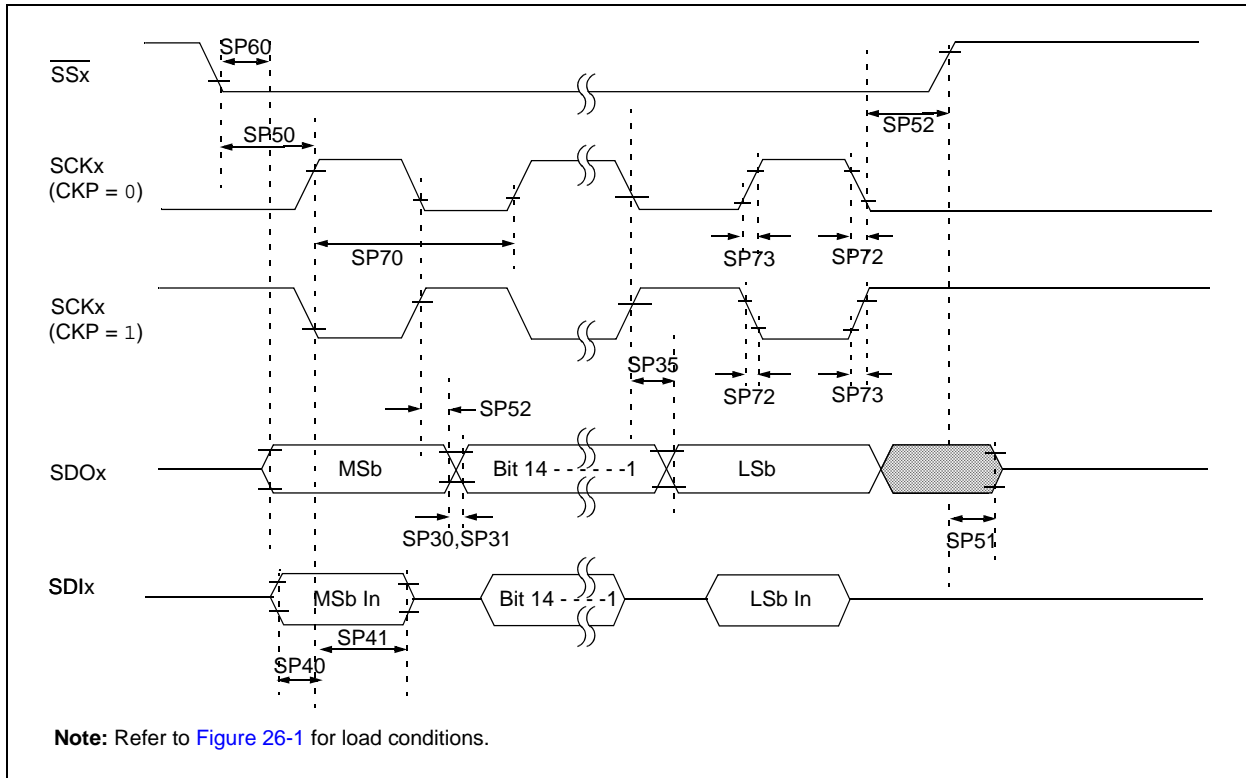
**2:** Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.

**3:** The minimum clock period for SCKx is 66.7 ns. Therefore, the SCK clock generated by the Master must not violate this specification.

**4:** Assumes 50 pF load on all SPIx pins.

# PIC24FJ16MC101/102

**FIGURE 26-16: SPIx SLAVE MODE (FULL-DUPLEX, CKE = 1, CKP = 1, SMP = 0) TIMING CHARACTERISTICS**



**TABLE 26-34: SPIx SLAVE MODE (FULL-DUPLEX, CKE = 1, CKP = 1, SMP = 0) TIMING REQUIREMENTS**

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	Symbol	Characteristic <sup>(1)</sup>	Min	Typ <sup>(2)</sup>	Max	Units	Conditions
SP70	TscP	Maximum SCK Input Frequency	—	—	11	MHz	See <b>Note 3</b>
SP72	TscF	SCKx Input Fall Time	—	—	—	ns	See parameter <b>DO32</b> and <b>Note 4</b>
SP73	TscR	SCKx Input Rise Time	—	—	—	ns	See parameter <b>DO31</b> and <b>Note 4</b>
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See parameter <b>DO32</b> and <b>Note 4</b>
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See parameter <b>DO31</b> and <b>Note 4</b>
SP35	Tsch2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	6	20	ns	—
SP36	TdoV2scH, TdoV2scL	SDOx Data Output Setup to First SCKx Edge	30	—	—	ns	—
SP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	30	—	—	ns	—
SP41	Tsch2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	30	—	—	ns	—
SP50	TssL2scH, TssL2scL	$\overline{SSx}$ ↓ to SCKx ↑ or SCKx Input	120	—	—	ns	—
SP51	TssH2doZ	$\overline{SSx}$ ↑ to SDOx Output High-Impedance <sup>(4)</sup>	10	—	50	ns	—
SP52	Tsch2ssH TscL2ssH	$\overline{SSx}$ after SCKx Edge	1.5 TCY + 40	—	—	ns	See <b>Note 4</b>
SP60	TssL2doV	SDOx Data Output Valid after $\overline{SSx}$ Edge	—	—	50	ns	—

**Note 1:** These parameters are characterized, but are not tested in manufacturing.

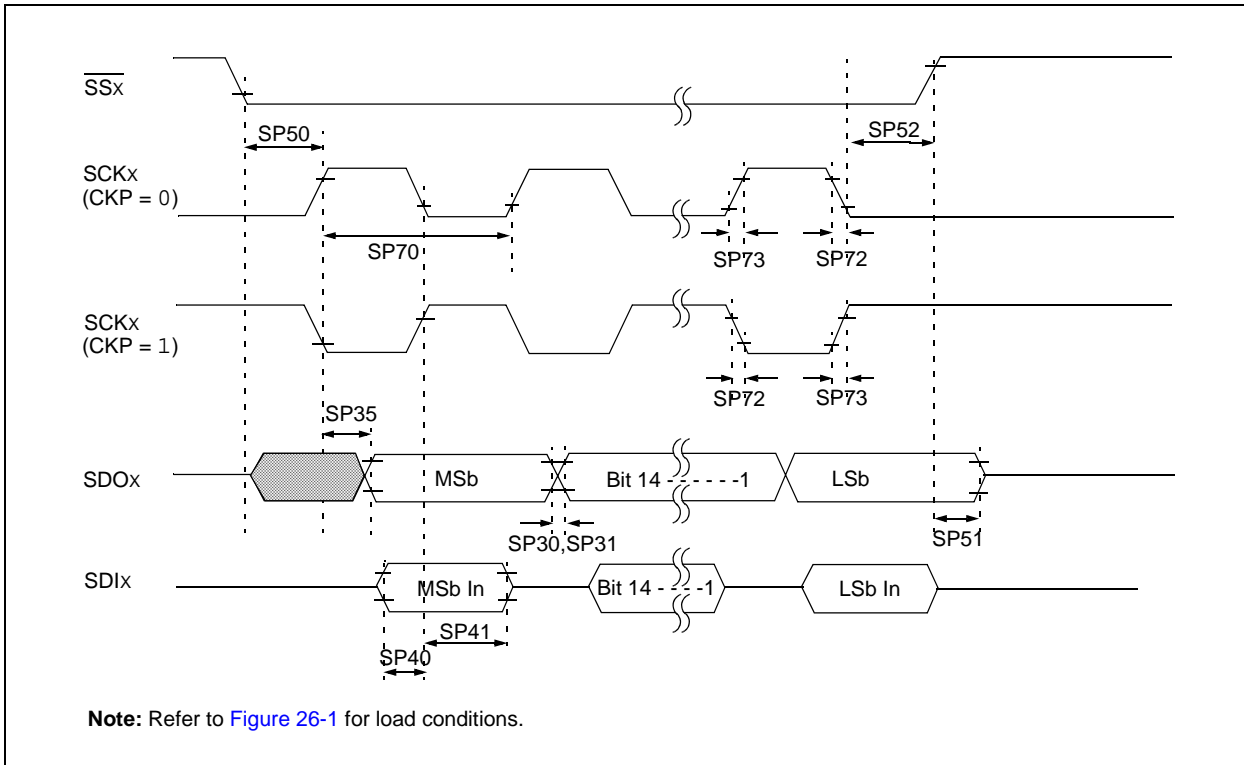
**2:** Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.

**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.

# PIC24FJ16MC101/102

**FIGURE 26-17: SPIx SLAVE MODE (FULL-DUPLEX CKE = 0, CKP = 1, SMP = 0) TIMING CHARACTERISTICS**



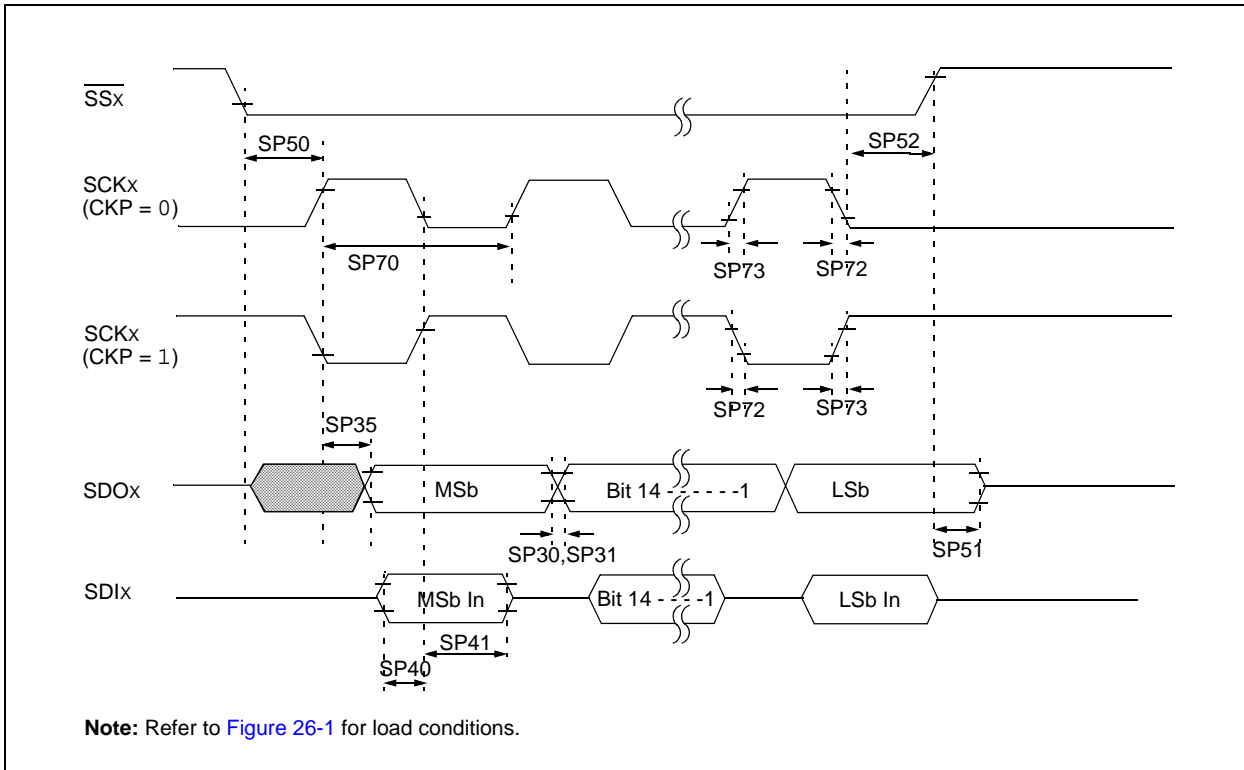
**TABLE 26-35: SPIx SLAVE MODE (FULL-DUPLEX, CKE = 0, CKP = 1, SMP = 0) TIMING REQUIREMENTS**

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	Symbol	Characteristic <sup>(1)</sup>	Min	Typ <sup>(2)</sup>	Max	Units	Conditions
SP70	TscP	Maximum SCK Input Frequency	—	—	15	MHz	See <b>Note 3</b>
SP72	TscF	SCKx Input Fall Time	—	—	—	ns	See parameter <b>DO32</b> and <b>Note 4</b>
SP73	TscR	SCKx Input Rise Time	—	—	—	ns	See parameter <b>DO31</b> and <b>Note 4</b>
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See parameter <b>DO32</b> and <b>Note 4</b>
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See parameter <b>DO31</b> and <b>Note 4</b>
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	6	20	ns	—
SP36	TdoV2scH, TdoV2scL	SDOx Data Output Setup to First SCKx Edge	30	—	—	ns	—
SP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	30	—	—	ns	—
SP41	Tsch2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	30	—	—	ns	—
SP50	TssL2scH, TssL2scL	$\overline{SSx} \downarrow$ to SCKx $\uparrow$ or SCKx Input	120	—	—	ns	—
SP51	TssH2doZ	$\overline{SSx} \uparrow$ to SDOx Output High-Impedance <sup>(4)</sup>	10	—	50	ns	—
SP52	Tsch2ssH TscL2ssH	$\overline{SSx}$ after SCKx Edge	1.5 Tcy + 40	—	—	ns	See <b>Note 4</b>

- Note 1:** These parameters are characterized, but are not tested in manufacturing.  
**Note 2:** Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.  
**Note 3:** The minimum clock period for SCKx is 66.7 ns. Therefore, the SCK clock generated by the Master must not violate this specification.  
**Note 4:** Assumes 50 pF load on all SPIx pins.

# PIC24FJ16MC101/102

**FIGURE 26-18: SPIx SLAVE MODE (FULL-DUPLEX, CKE = 0, CKP = 0, SMP = 0) TIMING CHARACTERISTICS**





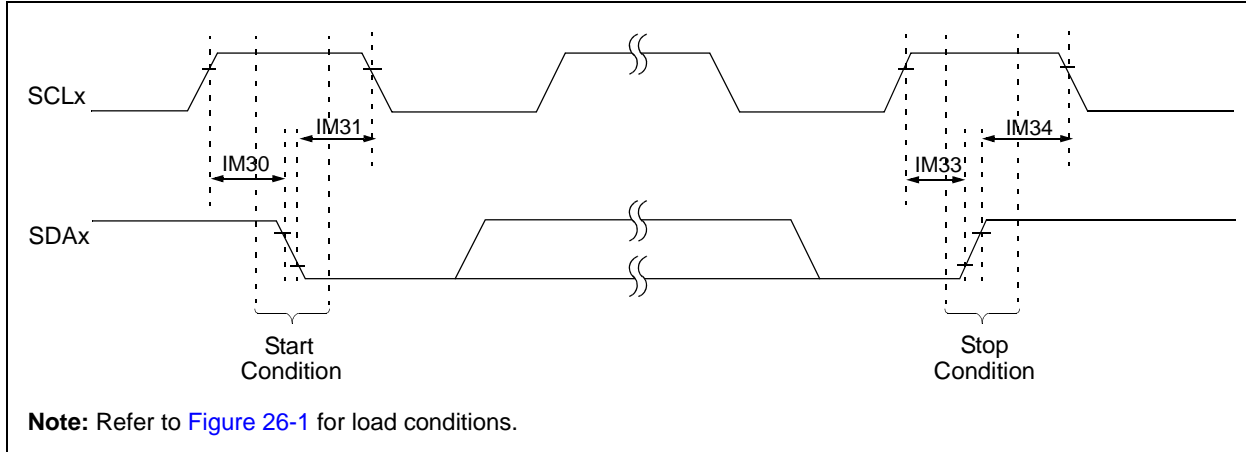
**TABLE 26-36: SPIx SLAVE MODE (FULL-DUPLEX, CKE = 0, CKP = 0, SMP = 0) TIMING REQUIREMENTS**

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	Symbol	Characteristic <sup>(1)</sup>	Min	Typ <sup>(2)</sup>	Max	Units	Conditions
SP70	TscP	Maximum SCK Input Frequency	—	—	11	MHz	See <b>Note 3</b>
SP72	TscF	SCKx Input Fall Time	—	—	—	ns	See parameter <b>DO32</b> and <b>Note 4</b>
SP73	TscR	SCKx Input Rise Time	—	—	—	ns	See parameter <b>DO31</b> and <b>Note 4</b>
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See parameter <b>DO32</b> and <b>Note 4</b>
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See parameter <b>DO31</b> and <b>Note 4</b>
SP35	Tsch2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	6	20	ns	—
SP36	TdoV2scH, TdoV2scL	SDOx Data Output Setup to First SCKx Edge	30	—	—	ns	—
SP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	30	—	—	ns	—
SP41	Tsch2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	30	—	—	ns	—
SP50	TssL2scH, TssL2scL	$\overline{SSx}$ ↓ to SCKx ↑ or SCKx Input	120	—	—	ns	—
SP51	TssH2doZ	$\overline{SSx}$ ↑ to SDOx Output High-Impedance <sup>(4)</sup>	10	—	50	ns	—
SP52	Tsch2ssH TscL2ssH	$\overline{SSx}$ after SCKx Edge	1.5 TCY + 40	—	—	ns	See <b>Note 4</b>

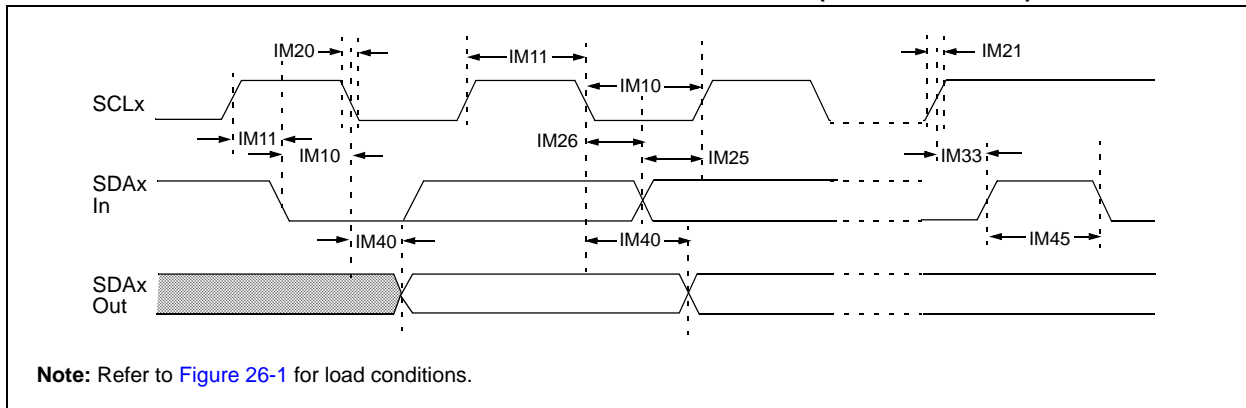
- Note 1:** These parameters are characterized, but are not tested in manufacturing.
- Note 2:** Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.
- Note 3:** The minimum clock period for SCKx is 91 ns. Therefore, the SCK clock generated by the Master must not violate this specification.
- Note 4:** Assumes 50 pF load on all SPIx pins.

# PIC24FJ16MC101/102

**FIGURE 26-19: I2Cx BUS START/STOP BITS TIMING CHARACTERISTICS (MASTER MODE)**



**FIGURE 26-20: I2Cx BUS DATA TIMING CHARACTERISTICS (MASTER MODE)**



**TABLE 26-37: I2Cx BUS DATA TIMING REQUIREMENTS (MASTER 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 <sup>(1)</sup>	Max	Units	Conditions
IM10	TLO:SCL	Clock Low Time	100 kHz mode	Tcy/2 (BRG + 1)	—	μs	—
			400 kHz mode	Tcy/2 (BRG + 1)	—	μs	—
			1 MHz mode <sup>(2)</sup>	Tcy/2 (BRG + 1)	—	μs	—
IM11	THI:SCL	Clock High Time	100 kHz mode	Tcy/2 (BRG + 1)	—	μs	—
			400 kHz mode	Tcy/2 (BRG + 1)	—	μs	—
			1 MHz mode <sup>(2)</sup>	Tcy/2 (BRG + 1)	—	μs	—
IM20	TF:SCL	SDAx and SCLx Fall Time	100 kHz mode	—	300	ns	Cb is specified to be from 10 to 400 pF
			400 kHz mode	20 + 0.1 Cb	300	ns	
			1 MHz mode <sup>(2)</sup>	—	100	ns	
IM21	TR:SCL	SDAx and SCLx Rise Time	100 kHz mode	—	1000	ns	Cb is specified to be from 10 to 400 pF
			400 kHz mode	20 + 0.1 Cb	300	ns	
			1 MHz mode <sup>(2)</sup>	—	300	ns	
IM25	TSU:DAT	Data Input Setup Time	100 kHz mode	250	—	ns	—
			400 kHz mode	100	—	ns	
			1 MHz mode <sup>(2)</sup>	40	—	ns	
IM26	THD:DAT	Data Input Hold Time	100 kHz mode	0	—	μs	—
			400 kHz mode	0	0.9	μs	
			1 MHz mode <sup>(2)</sup>	0.2	—	μs	
IM30	TSU:STA	Start Condition Setup Time	100 kHz mode	Tcy/2 (BRG + 1)	—	μs	Only relevant for Repeated Start condition
			400 kHz mode	Tcy/2 (BRG + 1)	—	μs	
			1 MHz mode <sup>(2)</sup>	Tcy/2 (BRG + 1)	—	μs	
IM31	THD:STA	Start Condition Hold Time	100 kHz mode	Tcy/2 (BRG + 1)	—	μs	After this period the first clock pulse is generated
			400 kHz mode	Tcy/2 (BRG + 1)	—	μs	
			1 MHz mode <sup>(2)</sup>	Tcy/2 (BRG + 1)	—	μs	
IM33	TSU:STO	Stop Condition Setup Time	100 kHz mode	Tcy/2 (BRG + 1)	—	μs	—
			400 kHz mode	Tcy/2 (BRG + 1)	—	μs	
			1 MHz mode <sup>(2)</sup>	Tcy/2 (BRG + 1)	—	μs	
IM34	THD:STO	Stop Condition Hold Time	100 kHz mode	Tcy/2 (BRG + 1)	—	ns	—
			400 kHz mode	Tcy/2 (BRG + 1)	—	ns	
			1 MHz mode <sup>(2)</sup>	Tcy/2 (BRG + 1)	—	ns	
IM40	TAA:SCL	Output Valid From Clock	100 kHz mode	—	3500	ns	—
			400 kHz mode	—	1000	ns	
			1 MHz mode <sup>(2)</sup>	—	400	ns	
IM45	TBF:SDA	Bus Free Time	100 kHz mode	4.7	—	μs	Time the bus must be free before a new transmission can start
			400 kHz mode	1.3	—	μs	
			1 MHz mode <sup>(2)</sup>	0.5	—	μs	
IM50	Cb	Bus Capacitive Loading	—	400	pF	—	
IM51	TPGD	Pulse Gobbler Delay	65	390	ns	See Note 3	

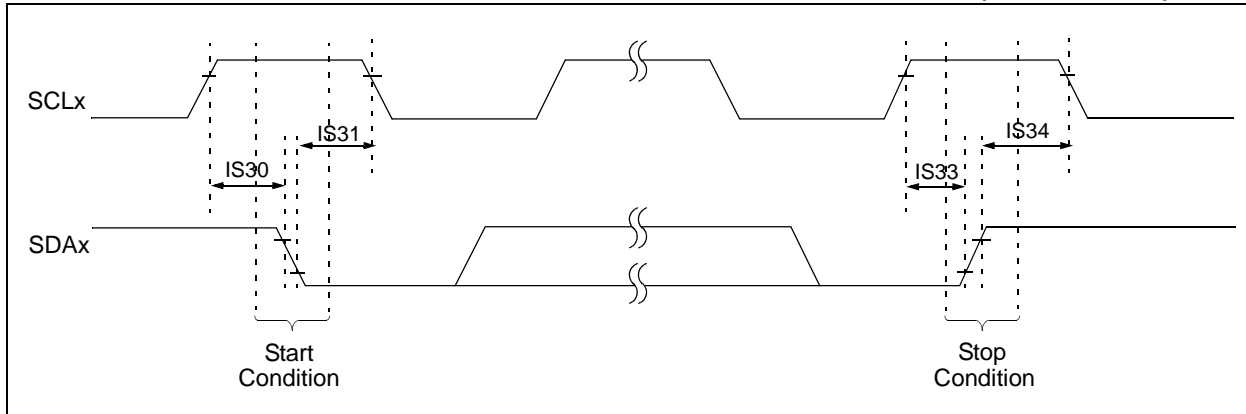
**Note 1:** BRG is the value of the I<sup>2</sup>C Baud Rate Generator. Refer to **Section 19. “Inter-Integrated Circuit (I<sup>2</sup>C™)”** (DS70195) in the “PIC24F Family Reference Manual”. Please see the Microchip web site for the latest PIC24F Family Reference Manual sections.

**2:** Maximum pin capacitance = 10 pF for all I2Cx pins (for 1 MHz mode only).

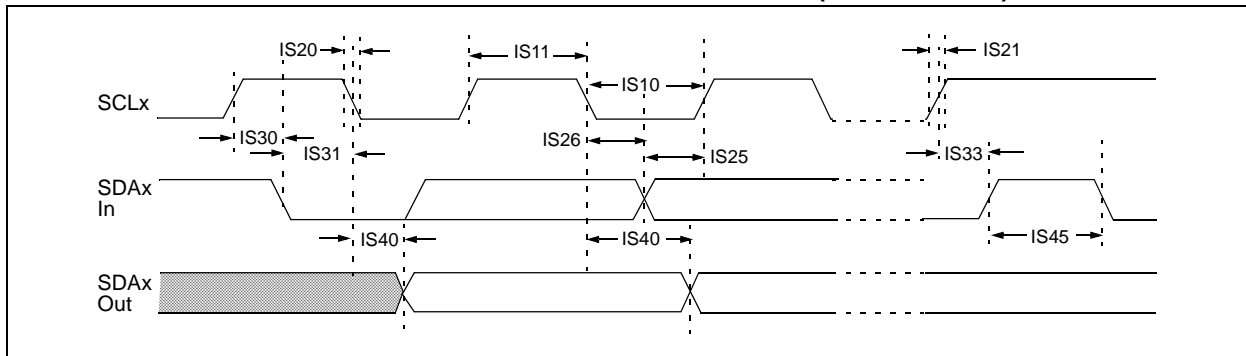
**3:** Typical value for this parameter is 130 ns.

# PIC24FJ16MC101/102

**FIGURE 26-21: I2Cx BUS START/STOP BITS TIMING CHARACTERISTICS (SLAVE MODE)**



**FIGURE 26-22: I2Cx BUS DATA TIMING CHARACTERISTICS (SLAVE MODE)**



**TABLE 26-38: I2Cx BUS DATA TIMING REQUIREMENTS (SLAVE MODE)**

AC CHARACTERISTICS				Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended			
Param.	Symbol	Characteristic		Min	Max	Units	Conditions
IS10	TLO:SCL	Clock Low Time	100 kHz mode	4.7	—	μs	Device must operate at a minimum of 1.5 MHz
			400 kHz mode	1.3	—	μs	Device must operate at a minimum of 10 MHz
			1 MHz mode <sup>(1)</sup>	0.5	—	μs	—
IS11	THI:SCL	Clock High Time	100 kHz mode	4.0	—	μs	Device must operate at a minimum of 1.5 MHz
			400 kHz mode	0.6	—	μs	Device must operate at a minimum of 10 MHz
			1 MHz mode <sup>(1)</sup>	0.5	—	μs	—
IS20	TF:SCL	SDAx and SCLx Fall Time	100 kHz mode	—	300	ns	CB is specified to be from 10 to 400 pF
			400 kHz mode	20 + 0.1 CB	300	ns	
			1 MHz mode <sup>(1)</sup>	—	100	ns	
IS21	TR:SCL	SDAx and SCLx Rise Time	100 kHz mode	—	1000	ns	CB is specified to be from 10 to 400 pF
			400 kHz mode	20 + 0.1 CB	300	ns	
			1 MHz mode <sup>(1)</sup>	—	300	ns	
IS25	TSU:DAT	Data Input Setup Time	100 kHz mode	250	—	ns	—
			400 kHz mode	100	—	ns	
			1 MHz mode <sup>(1)</sup>	100	—	ns	
IS26	THD:DAT	Data Input Hold Time	100 kHz mode	0	—	μs	—
			400 kHz mode	0	0.9	μs	
			1 MHz mode <sup>(1)</sup>	0	0.3	μs	
IS30	TSU:STA	Start Condition Setup Time	100 kHz mode	4.7	—	μs	Only relevant for Repeated Start condition
			400 kHz mode	0.6	—	μs	
			1 MHz mode <sup>(1)</sup>	0.25	—	μs	
IS31	THD:STA	Start Condition Hold Time	100 kHz mode	4.0	—	μs	After this period, the first clock pulse is generated
			400 kHz mode	0.6	—	μs	
			1 MHz mode <sup>(1)</sup>	0.25	—	μs	
IS33	TSU:STO	Stop Condition Setup Time	100 kHz mode	4.7	—	μs	—
			400 kHz mode	0.6	—	μs	
			1 MHz mode <sup>(1)</sup>	0.6	—	μs	
IS34	THD:STO	Stop Condition Hold Time	100 kHz mode	4000	—	ns	—
			400 kHz mode	600	—	ns	
			1 MHz mode <sup>(1)</sup>	250	—	ns	
IS40	TAA:SCL	Output Valid From Clock	100 kHz mode	0	3500	ns	—
			400 kHz mode	0	1000	ns	
			1 MHz mode <sup>(1)</sup>	0	350	ns	
IS45	TBF:SDA	Bus Free Time	100 kHz mode	4.7	—	μs	Time the bus must be free before a new transmission can start
			400 kHz mode	1.3	—	μs	
			1 MHz mode <sup>(1)</sup>	0.5	—	μs	
IS50	CB	Bus Capacitive Loading		—	400	pF	—

**Note 1:** Maximum pin capacitance = 10 pF for all I2Cx pins (for 1 MHz mode only).

# PIC24FJ16MC101/102

**TABLE 26-39: 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
<b>Device Supply</b>							
AD01	AVDD	Module VDD Supply <sup>(2,4)</sup>	Greater of VDD – 0.3 or 2.9	—	Lesser of VDD + 0.3 or 3.6	V	—
AD02	AVSS	Module VSS Supply <sup>(2,5)</sup>	VSS – 0.3	—	VSS + 0.3	V	—
AD09	IAD	Operating Current	—	7.0	9.0	mA	See <b>Note 1</b>
<b>Analog Input</b>							
AD12	VINH	Input Voltage Range VINH <sup>(2)</sup>	VINL	—	AVDD	V	This voltage reflects Sample and Hold Channels 0, 1, 2, and 3 (CH0-CH3), positive input
AD13	VINL	Input Voltage Range VINL <sup>(2)</sup>	AVSS	—	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 <sup>(3)</sup>	—	—	200	Ω	—

- 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.
- 4:** This pin may not be available on all devices, in which case, this pin will be connected to VDD internally. See the “[Pin Diagrams](#)” section for availability.
- 5:** This pin may not be available on all devices, in which case, this pin will be connected to VSS internally. See the “[Pin Diagrams](#)” section for availability.

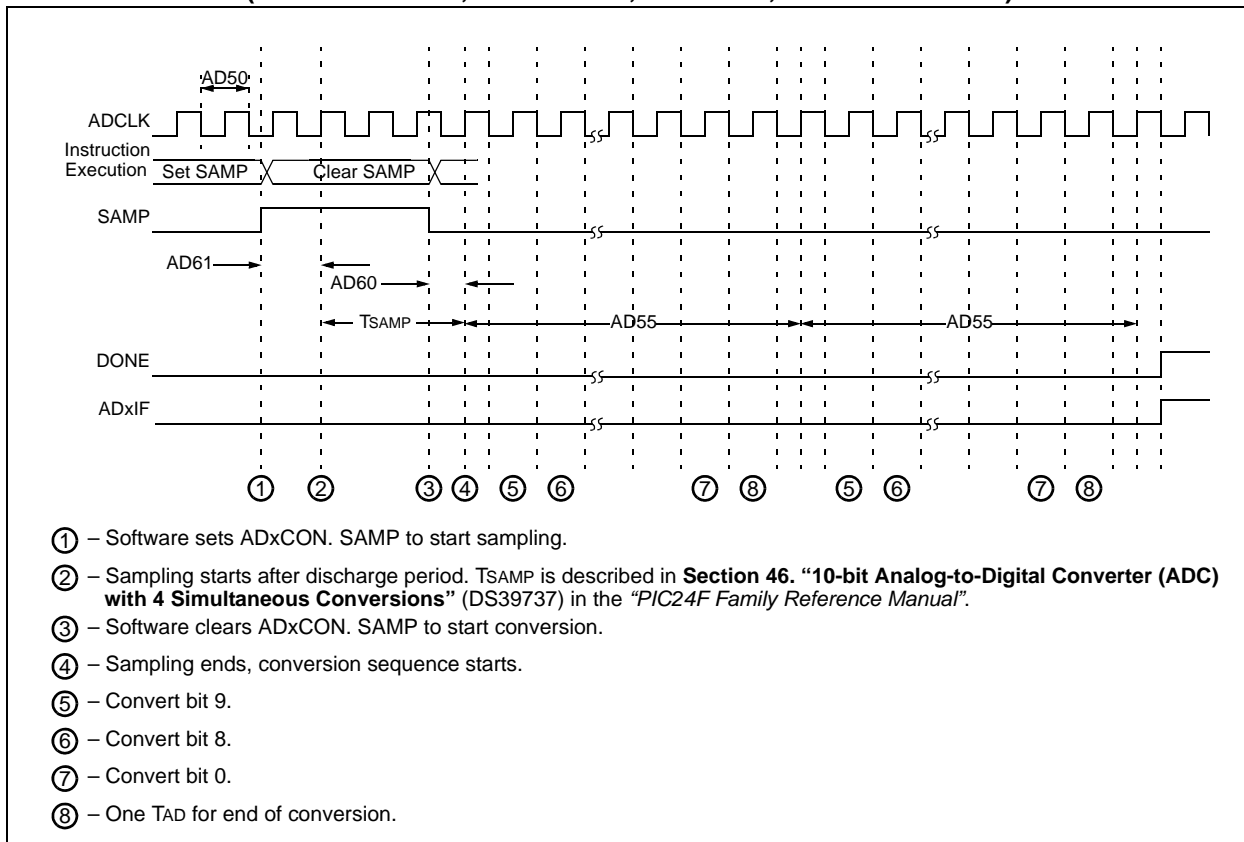
**TABLE 26-40: 10-BIT ADC MODULE SPECIFICATIONS**

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
<b>10-bit ADC Accuracy – Measurements with AVDD/AVSS<sup>(3)</sup></b>							
AD20b	Nr	Resolution	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	3	7	15	LSb	VINL = AVSS = 0V, AVDD = 3.6V
AD24b	EOFF	Offset Error	1.5	3	7	LSb	VINL = AVSS = 0V, AVDD = 3.6V
AD25b	—	Monotonicity	—	—	—	—	Guaranteed <sup>(1)</sup>
<b>Dynamic Performance (10-bit Mode)<sup>(2)</sup></b>							
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	—

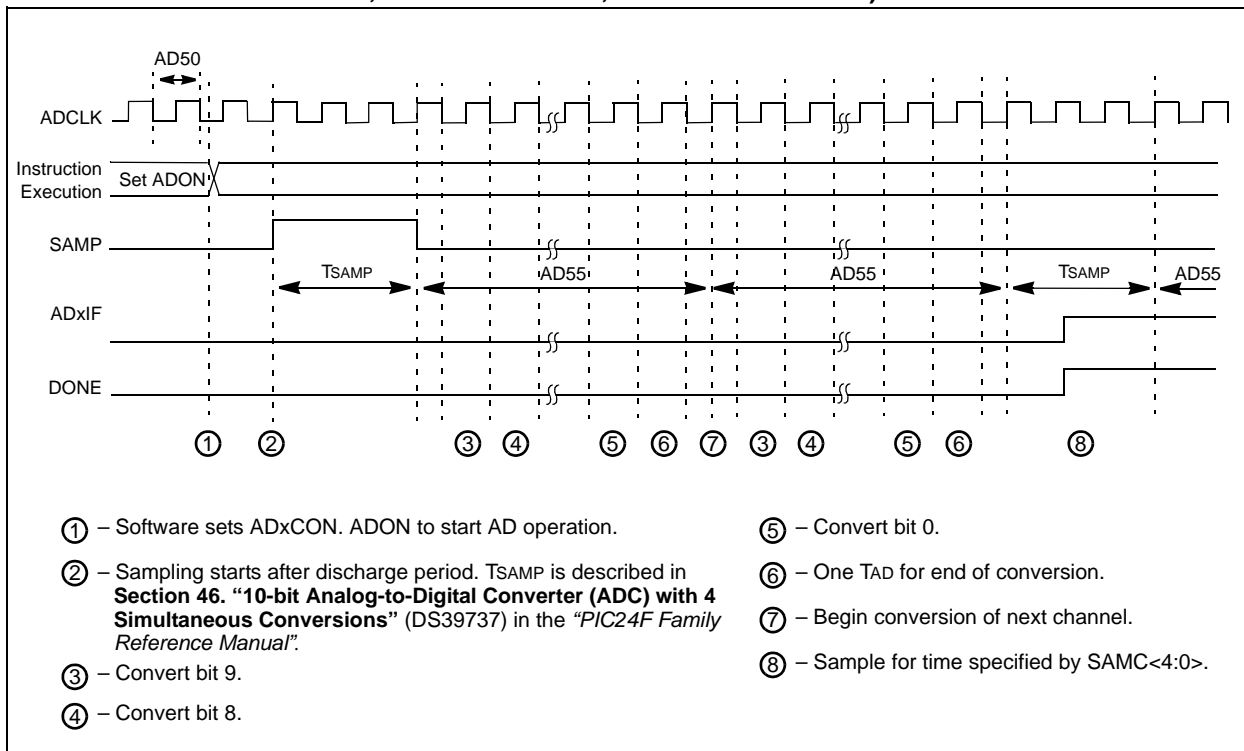
- Note 1:** The analog-to-digital conversion result never decreases with an increase in the input voltage, and has no missing codes.
- 2:** These parameters are characterized by similarity, but are not tested in manufacturing.
- 3:** These parameters are characterized, but are tested at 20 ksp/s only.

# PIC24FJ16MC101/102

**FIGURE 26-23: ADC CONVERSION TIMING CHARACTERISTICS (CHPS<1:0> = 01, SIMSAM = 0, ASAM = 0, SSRC<2:0> = 000)**



**FIGURE 26-24: ADC CONVERSION TIMING CHARACTERISTICS (CHPS<1:0> = 01, SIMSAM = 0, ASAM = 1, SSRC<2:0> = 111, SAMC<4:0> = 00001)**





**TABLE 26-41: 10-BIT ADC CONVERSION 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 <sup>(1)</sup>	Max.	Units	Conditions
<b>Clock Parameters<sup>(2)</sup></b>							
AD50	TAD	ADC Clock Period	76	—	—	ns	—
AD51	tRC	ADC Internal RC Oscillator Period	—	250	—	ns	—
<b>Conversion Rate</b>							
AD55	tCONV	Conversion Time	—	12 TAD	—	—	—
AD56	FCNV	Throughput Rate	—	—	1.1	MSPS	—
AD57	TSAMP	Sample Time	2.0 TAD	—	—	—	—
<b>Timing Parameters</b>							
AD60	tPCS	Conversion Start from Sample Trigger <sup>(1)</sup>	2.0 TAD	—	3.0 TAD	—	Auto-Convert Trigger (SSRC<2:0> = 111) not selected
AD61	tPSS	Sample Start from Setting Sample (SAMP) bit <sup>(1)</sup>	2.0 TAD	—	3.0 TAD	—	—
AD62	tCSS	Conversion Completion to Sample Start (ASAM = 1) <sup>(1)</sup>	—	0.5 TAD	—	—	—
AD63	tDPU	Time to Stabilize Analog Stage from ADC Off to ADC On <sup>(1)</sup>	—	—	20	μs	—

**Note 1:** These parameters are characterized but not tested in manufacturing.

**2:** Because the sample caps will eventually lose charge, clock rates below 10 kHz may affect linearity performance, especially at elevated temperatures.

# PIC24FJ16MC101/102

**TABLE 26-42: COMPARATOR TIMING 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
300	TRESP	Response Time <sup>(1,2)</sup>	—	150	400	ns	—
301	TMC2OV	Comparator Mode Change to Output Valid <sup>(1)</sup>	—	—	10	μs	—
302	TON2OV	Comparator Enabled to Output Valid <sup>(1)</sup>	—	—	10	μs	—

**Note 1:** Parameters are characterized but not tested.

**Note 2:** Response time measured with one comparator input at (VDD - 1.5)/2, while the other input transitions from VSS to VDD.

**TABLE 26-43: COMPARATOR MODULE 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	Max.	Units	Conditions
D300	VIOFF	Input Offset Voltage <sup>(1)</sup>	—	±10	—	mV	—
D301	VICM	Input Common Mode Voltage <sup>(1)</sup>	0	—	AVDD-1.5V	V	—
D302	CMRR	Common Mode Rejection Ratio <sup>(1)</sup>	-54	—	—	dB	—
D305	IVREF	Internal Voltage Reference <sup>(1)</sup>	1.116	1.24	1.364	V	—

**Note 1:** Parameters are characterized but not tested.

**TABLE 26-44: COMPARATOR REFERENCE VOLTAGE SETTling TIME 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
VR310	TSET	Settling Time <sup>(1)</sup>	—	—	10	μs	—

**Note 1:** Setting time measured while CVRR = 1 and CVR3:CVR0 bits transition from '0000' to '1111'.

**TABLE 26-45: COMPARATOR REFERENCE VOLTAGE SPECIFICATIONS**

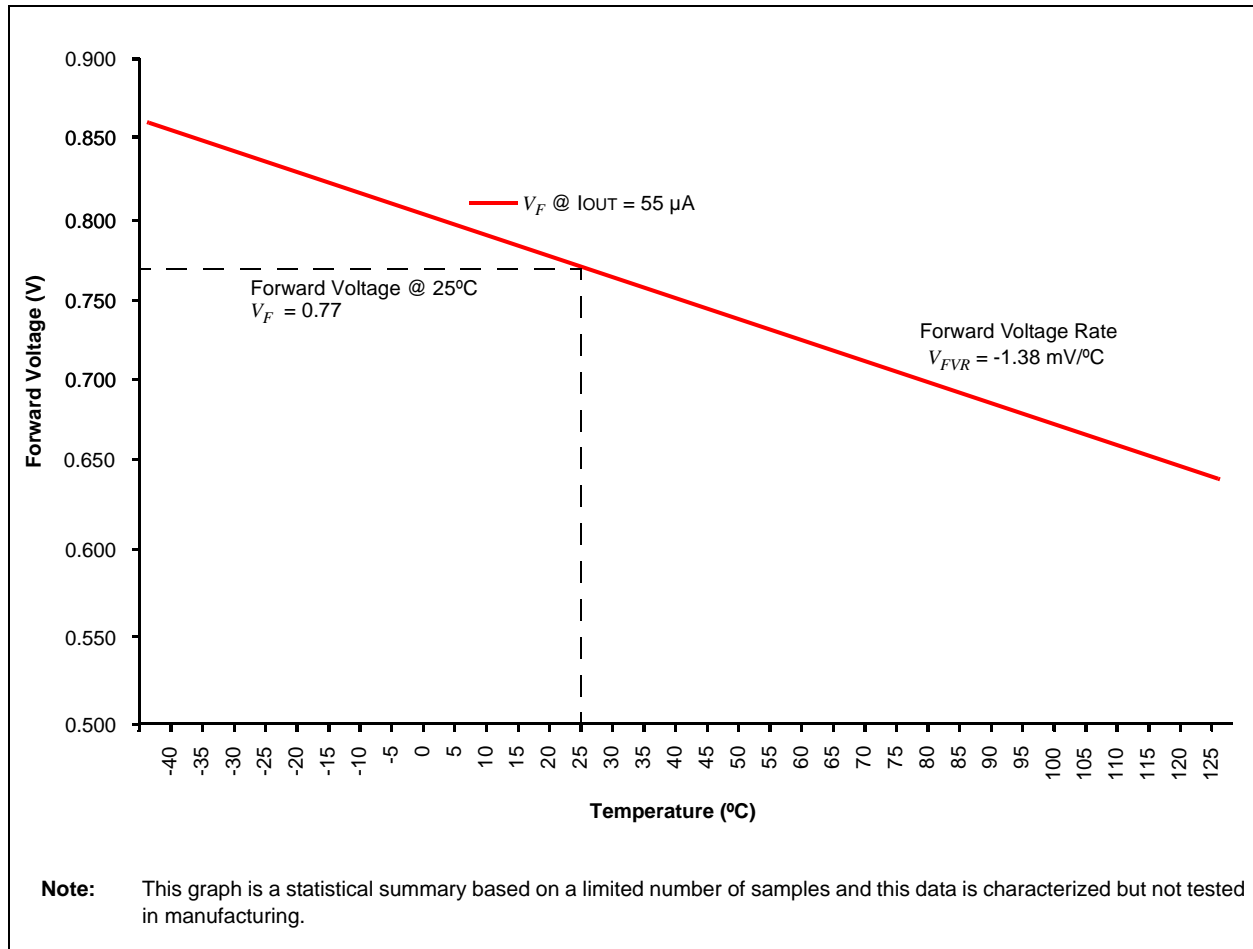
DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	Symbol	Characteristic	Min.	Typ	Max.	Units	Conditions
VRD310	CVRES	Resolution	CVRSRC/24	—	CVRSRC/32	LSb	—
VRD311	CVRAA	Absolute Accuracy	—	—	0.5	LSb	—
VRD312	CVRUR	Unit Resistor Value (R)	—	2k	—	Ω	—

**TABLE 26-46: CTMU CURRENT SOURCE 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.	Max.	Units	Conditions
<b>CTMU CURRENT SOURCE</b>							
CTMUI1	IOUT1	Base Range <sup>(1)</sup>	—	550	—	na	IRNG<1:0> bits (CTMUICON<9:8>) = 01
CTMUI2	IOUT2	10x Range <sup>(1)</sup>	—	5.5	—	μA	IRNG<1:0> bits (CTMUICON<9:8>) = 10
CTMUI3	IOUT3	100x Range <sup>(1)</sup>	—	55	—	μA	IRNG<1:0> bits (CTMUICON<9:8>) = 11
<b>Internal Diode</b>							
CTMUFV1	VF	Forward Voltage <sup>(2)</sup>	—	0.77	—	V	IRNG<1:0> bits (CTMUICON<9:8>) = 0b11 @ 25°C
CTMUFV2	VFVR	Forward Voltage Rate <sup>(2)</sup>	—	-1.38	—	mV/°C	IRNG<1:0> bits (CTMUICON<9:8>) = 0b11

- Note 1:** Nominal value at center point of current trim range (ITRIM<5:0> bits (CTMUICON<15:10>) = 0b000000).  
**Note 2:** ADC module configured for conversion speed of 500 ksp. Parameters are characterized but not tested in manufacturing.

**FIGURE 26-25: FORWARD VOLTAGE VERSUS TEMPERATURE**



# PIC24FJ16MC101/102

---

---

NOTES:

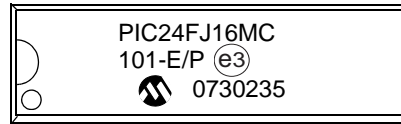
## 27.0 PACKAGING INFORMATION

### 27.1 Package Marking Information

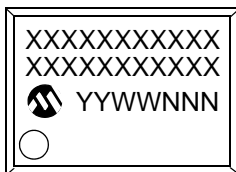
20-Lead PDIP



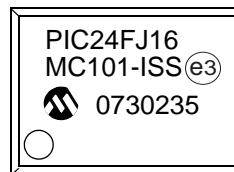
Example



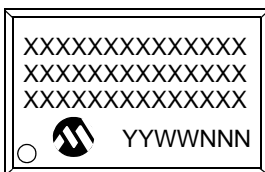
20-Lead SSOP



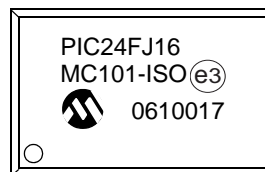
Example



20-Lead SOIC



Example

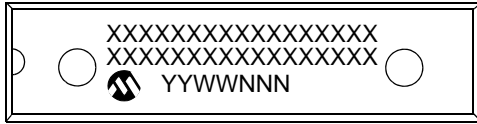


<b>Legend:</b>	XX...X	Customer-specific information
	Y	Year code (last digit of calendar year)
	YY	Year code (last 2 digits of calendar year)
	WW	Week code (week of January 1 is week '01')
	NNN	Alphanumeric traceability code
	(e3)	Pb-free JEDEC designator for Matte Tin (Sn)
	* (e3)	This package is Pb-free. The Pb-free JEDEC designator ((e3)) can be found on the outer packaging for this package.
<b>Note:</b>	If the full Microchip part number cannot be marked on one line, it is carried over to the next line, thus limiting the number of available characters for customer-specific information.	

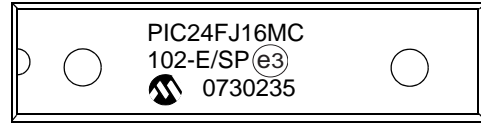
# PIC24FJ16MC101/102

## 27.1 Package Marking Information (Continued)

28-Lead SPDIP



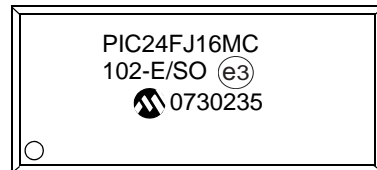
Example



28-Lead SOIC



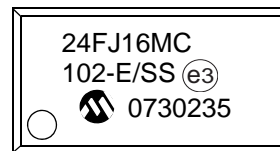
Example



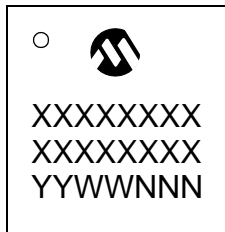
28-Lead SSOP



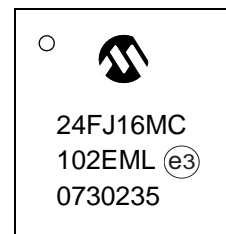
Example



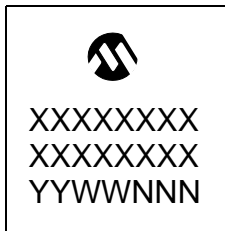
28-Lead QFN



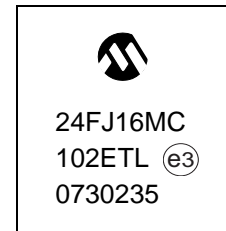
Example



36-Lead TLA



Example



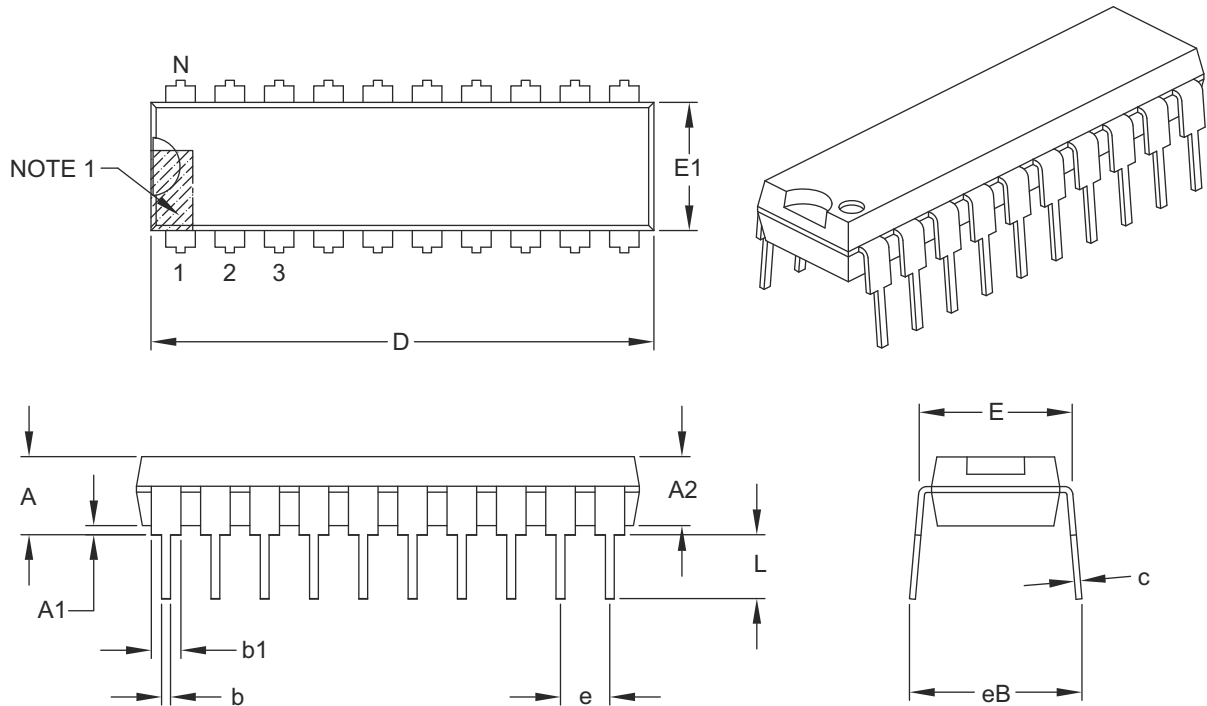
<b>Legend:</b>	XX...X	Customer-specific information
	Y	Year code (last digit of calendar year)
	YY	Year code (last 2 digits of calendar year)
	WW	Week code (week of January 1 is week '01')
	NNN	Alphanumeric traceability code
	(e3)	Pb-free JEDEC designator for Matte Tin (Sn)
	*	This package is Pb-free. The Pb-free JEDEC designator ((e3)) can be found on the outer packaging for this package.

**Note:** If the full Microchip part number cannot be marked on one line, it is carried over to the next line, thus limiting the number of available characters for customer-specific information.

## 27.2 Package Details

### 20-Lead Plastic Dual In-Line (P) – 300 mil Body [PDIP]

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



Dimension Limits	Units	INCHES		
		MIN	NOM	MAX
Number of Pins	N	20		
Pitch	e	.100 BSC		
Top to Seating Plane	A	–	–	.210
Molded Package Thickness	A2	.115	.130	.195
Base to Seating Plane	A1	.015	–	–
Shoulder to Shoulder Width	E	.300	.310	.325
Molded Package Width	E1	.240	.250	.280
Overall Length	D	.980	1.030	1.060
Tip to Seating Plane	L	.115	.130	.150
Lead Thickness	c	.008	.010	.015
Upper Lead Width	b1	.045	.060	.070
Lower Lead Width	b	.014	.018	.022
Overall Row Spacing §	eB	–	–	.430

**Notes:**

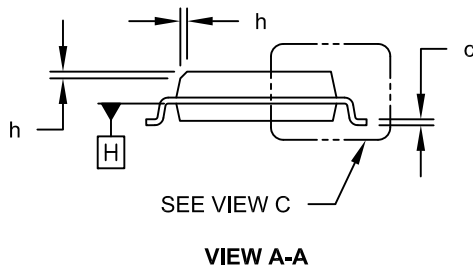
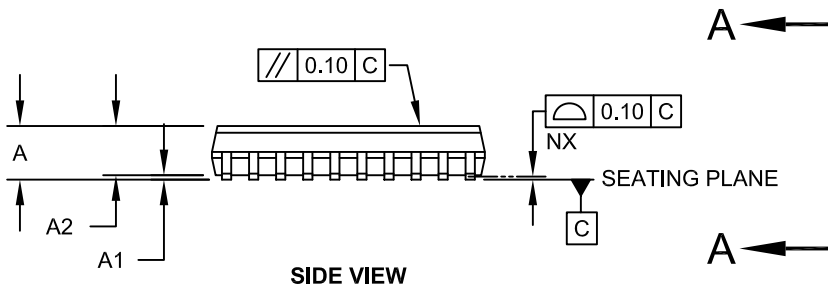
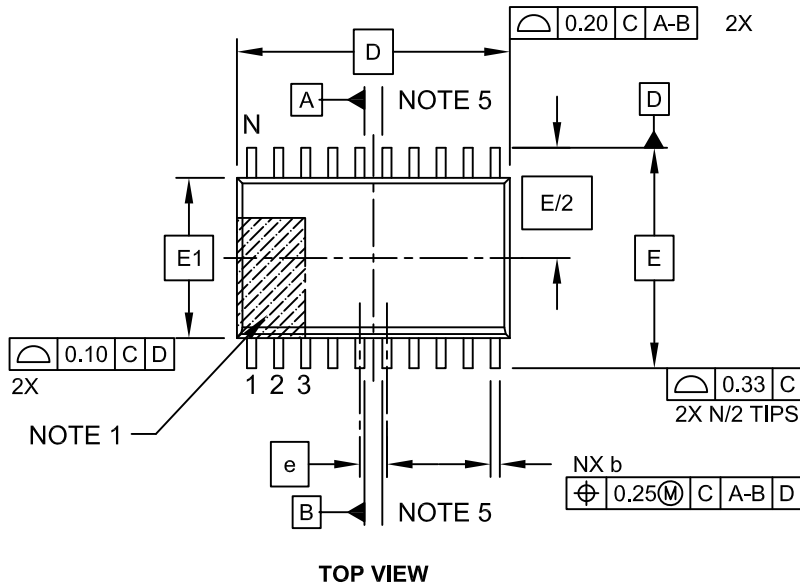
1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. § Significant Characteristic.
3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" per side.
4. Dimensioning and tolerancing per ASME Y14.5M.  
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-019B

# PIC24FJ16MC101/102

## 20-Lead Plastic Small Outline (SO) - Wide, 7.50 mm Body [SOIC]

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

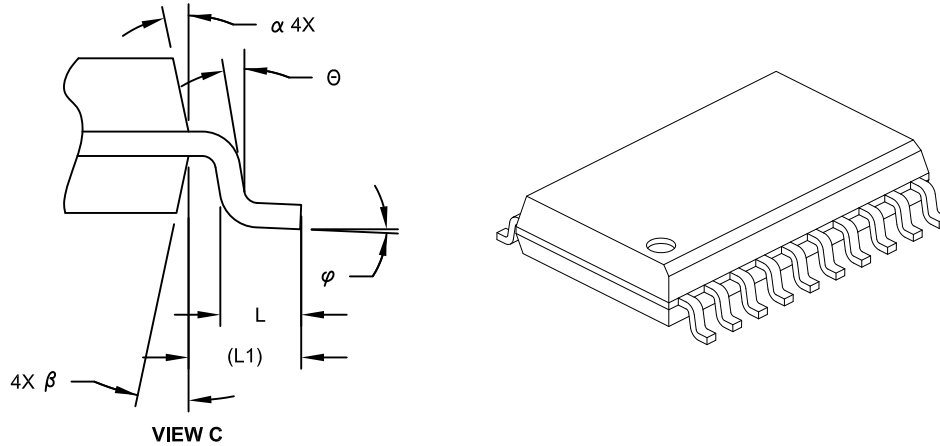


Microchip Technology Drawing C04-094C Sheet 1 of 2



## 20-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>



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Number of Pins	N	20		
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	12.80 BSC		
Chamfer (Optional)	h	0.25	-	0.75
Foot Length	L	0.40	-	1.27
Footprint	L1	1.40 REF		
Lead Angle	$\Theta$	0°	-	-
Foot Angle	$\varphi$	0°	-	8°
Lead Thickness	c	0.20	-	0.33
Lead Width	b	0.31	-	0.51
Mold Draft Angle Top	$\alpha$	5°	-	15°
Mold Draft Angle Bottom	$\beta$	5°	-	15°

**Notes:**

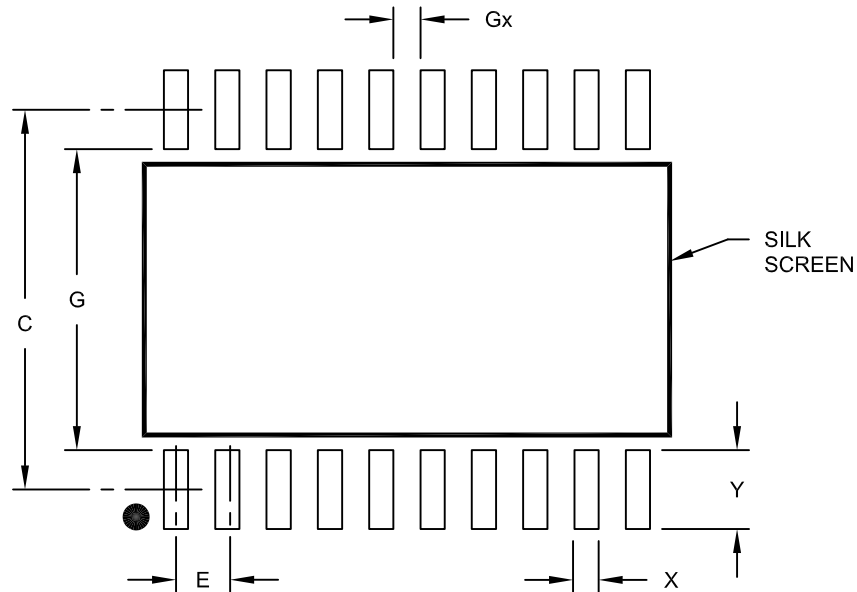
- Pin 1 visual index feature may vary, but must be located within the hatched area.
- § Significant Characteristic
- Dimension D does not include mold flash, protrusions or gate burrs, which shall not exceed 0.15 mm per end. Dimension E1 does not include interlead flash or protrusion, which shall not exceed 0.25 mm per side.
- 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.
- Datums A & B to be determined at Datum H.

Microchip Technology Drawing No. C04-094C Sheet 2 of 2

# PIC24FJ16MC101/102

20-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

Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Contact Pitch	E	1.27 BSC		
Contact Pad Spacing	C		9.40	
Contact Pad Width (X20)	X			0.60
Contact Pad Length (X20)	Y			1.95
Distance Between Pads	Gx	0.67		
Distance Between Pads	G	7.45		

Notes:

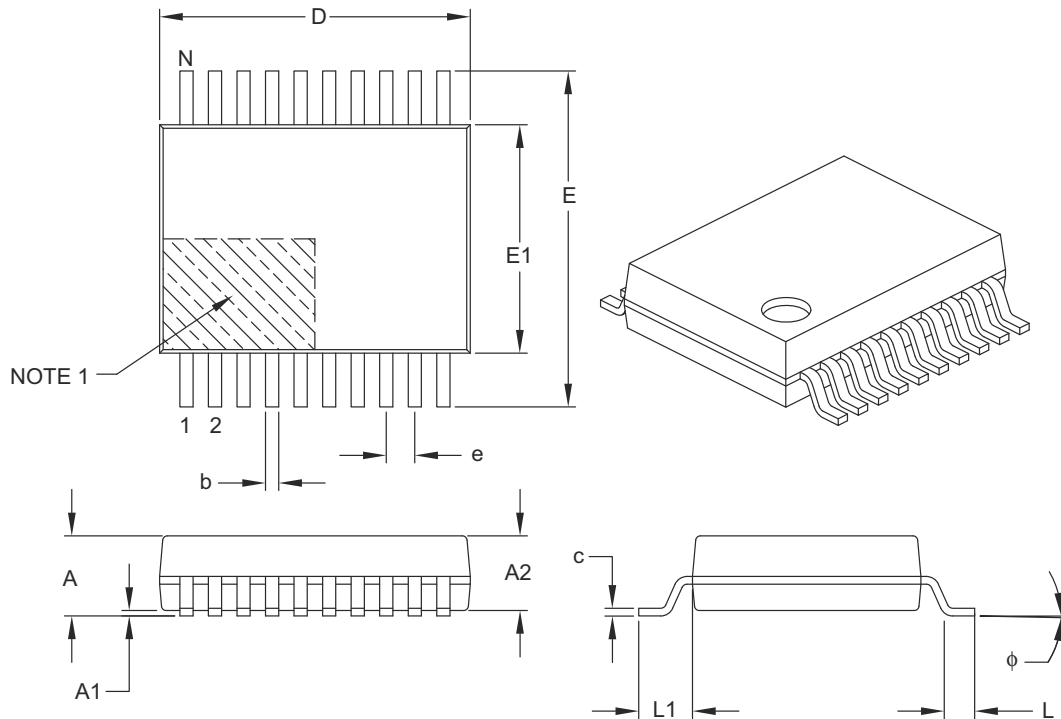
1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2094A

## 20-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>



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Number of Pins	N	20		
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	6.90	7.20	7.50
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:**

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.20 mm per side.
- Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

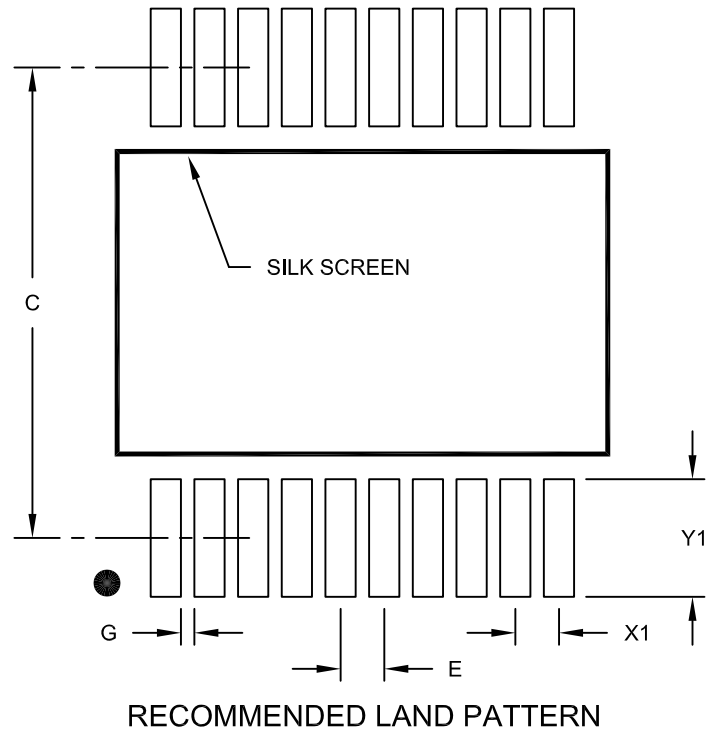
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-072B

# PIC24FJ16MC101/102

20-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>



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Contact Pitch	E	0.65 BSC		
Contact Pad Spacing	C		7.20	
Contact Pad Width (X20)	X1			0.45
Contact Pad Length (X20)	Y1			1.75
Distance Between Pads	G	0.20		

**Notes:**

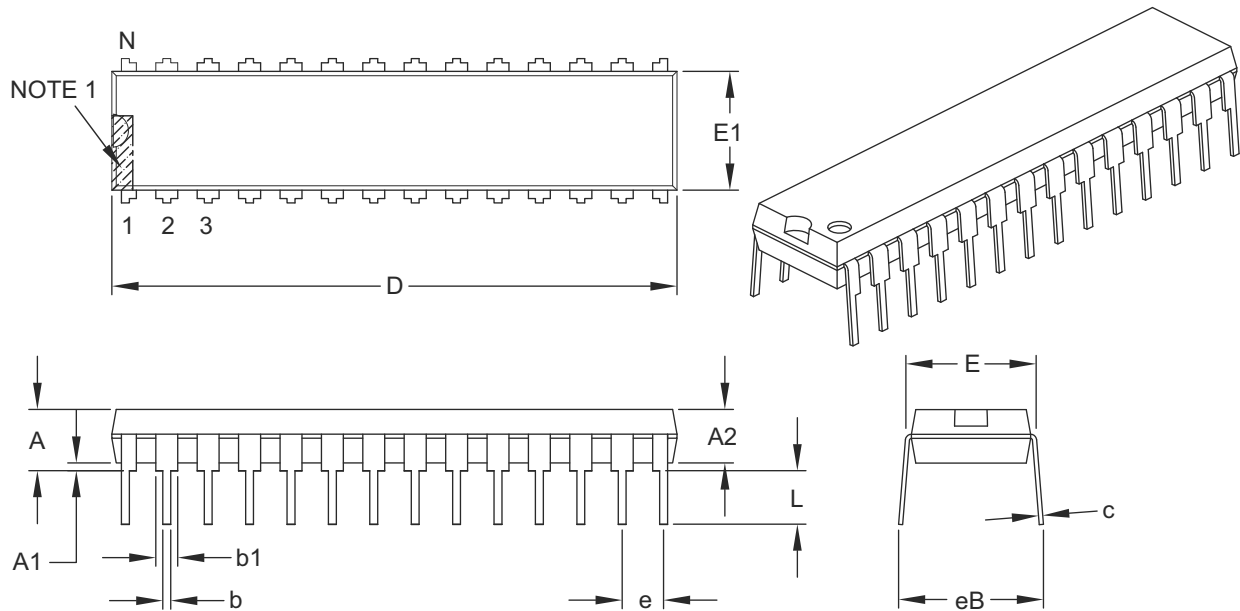
1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2072A

## 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		
		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.

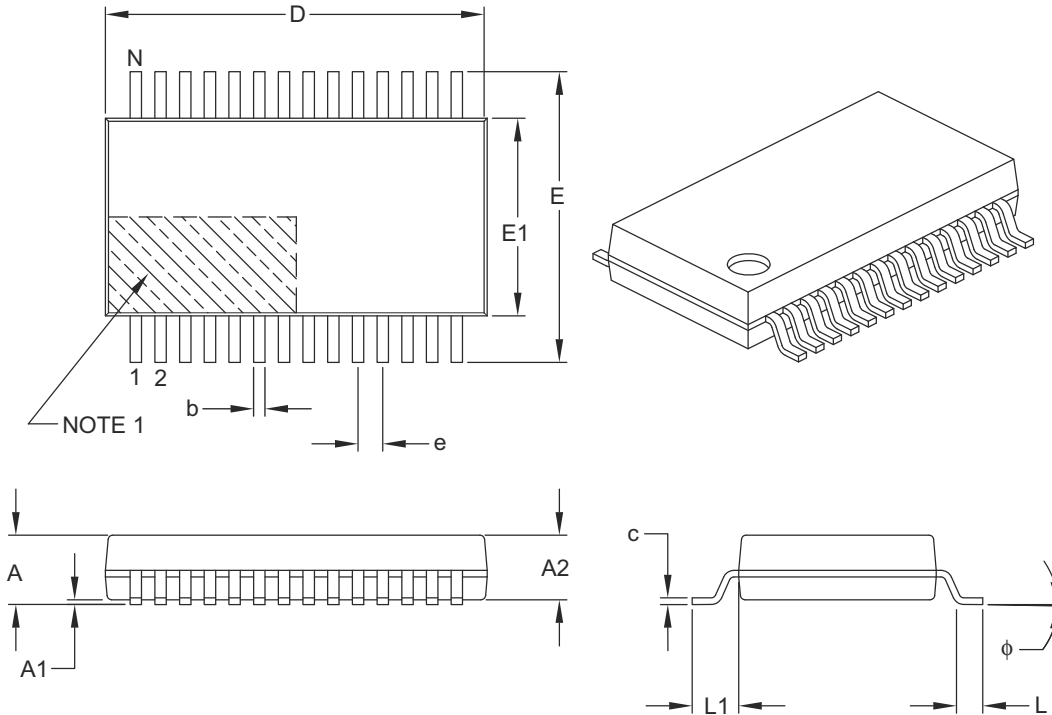
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-070B

# PIC24FJ16MC101/102

## 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:**

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.20 mm per side.
- Dimensioning and tolerancing per ASME Y14.5M.

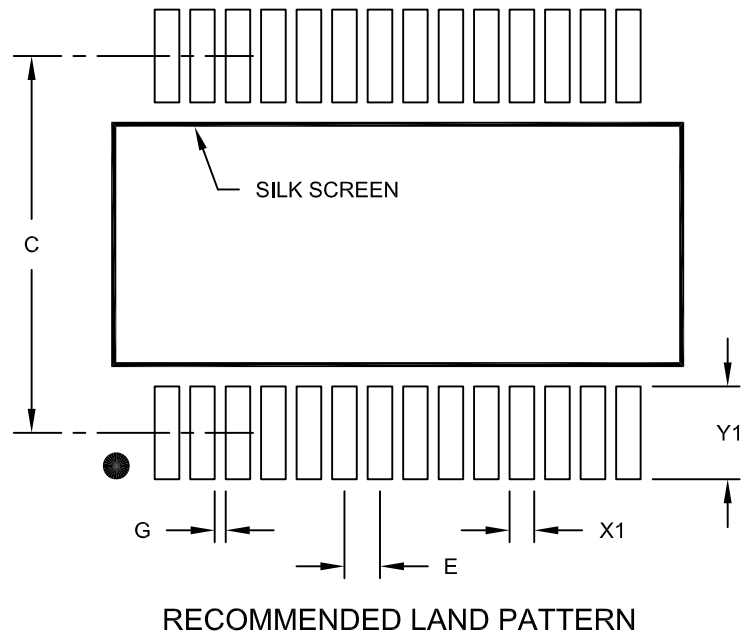
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-073B

## 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>



Dimension Limits	Units	MILLIMETERS		
		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

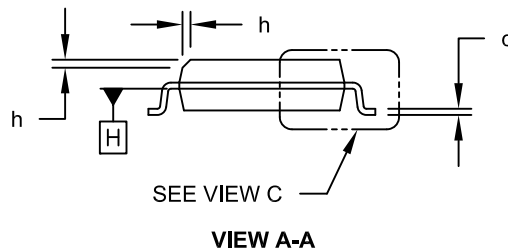
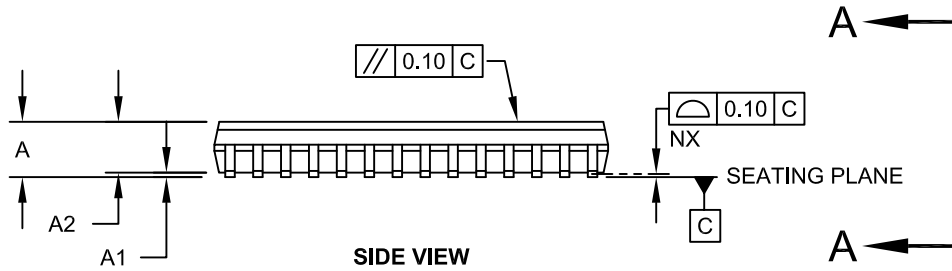
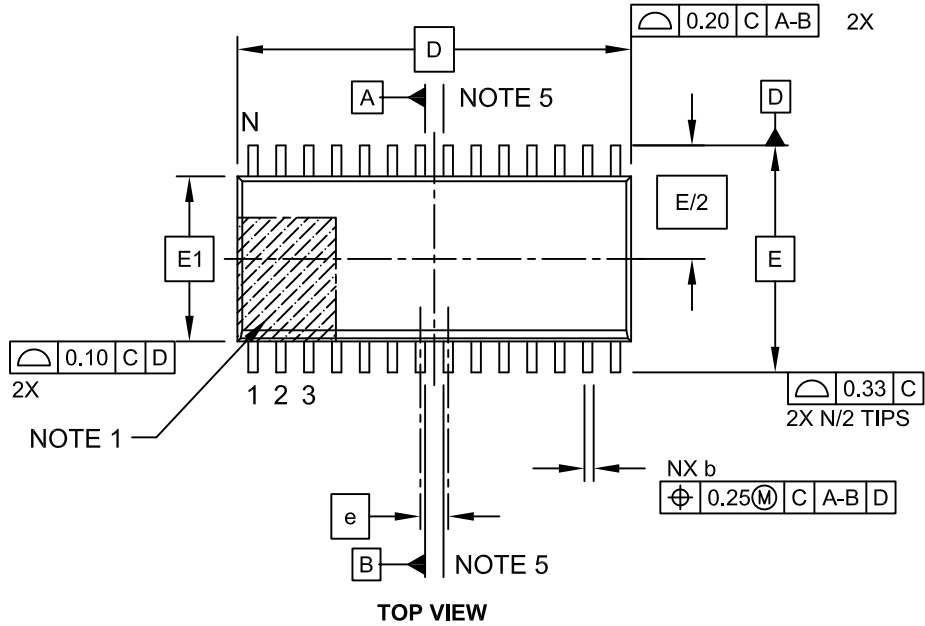
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2073A

# PIC24FJ16MC101/102

## 28-Lead Plastic Small Outline (SO) - Wide, 7.50 mm Body [SOIC]

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

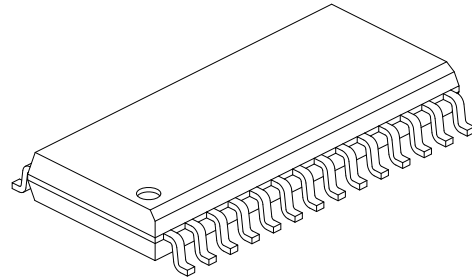
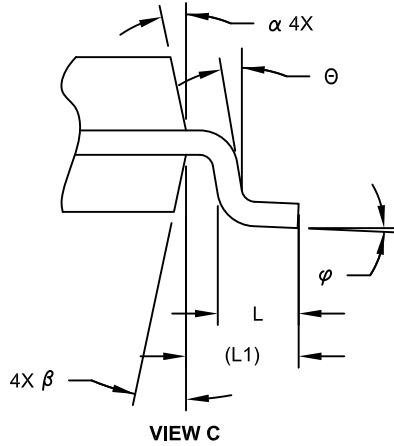


Microchip Technology Drawing C04-052C Sheet 1 of 2



## 28-Lead Plastic Small Outline (SO) - Wide, 7.50 mm Body [SOIC]

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



Dimension	Units	MILLIMETERS		
		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	$\theta$	0°	-	-
Foot Angle	$\varphi$	0°	-	8°
Lead Thickness	c	0.18	-	0.33
Lead Width	b	0.31	-	0.51
Mold Draft Angle Top	$\alpha$	5°	-	15°
Mold Draft Angle Bottom	$\beta$	5°	-	15°

**Notes:**

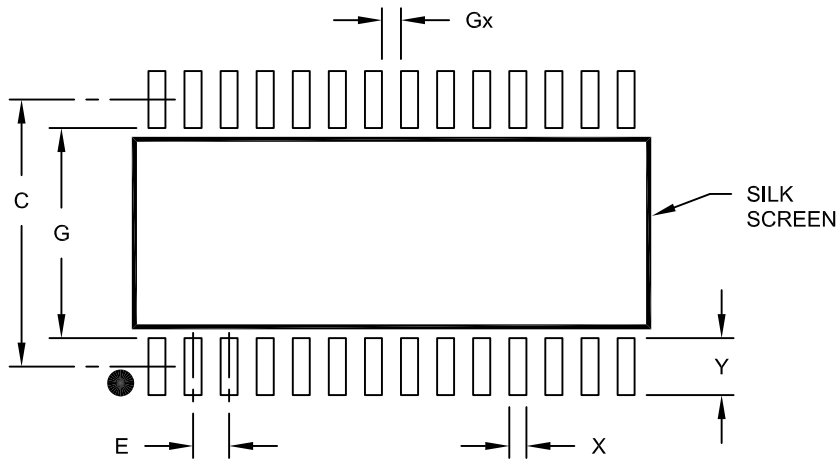
- Pin 1 visual index feature may vary, but must be located within the hatched area.
- § Significant Characteristic
- Dimension D does not include mold flash, protrusions or gate burrs, which shall not exceed 0.15 mm per end. Dimension E1 does not include interlead flash or protrusion, which shall not exceed 0.25 mm per side.
- 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.
- Datums A & B to be determined at Datum H.

Microchip Technology Drawing C04-052C Sheet 2 of 2

# PIC24FJ16MC101/102

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

Dimension Limits	Units	MILLIMETERS		
		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:**

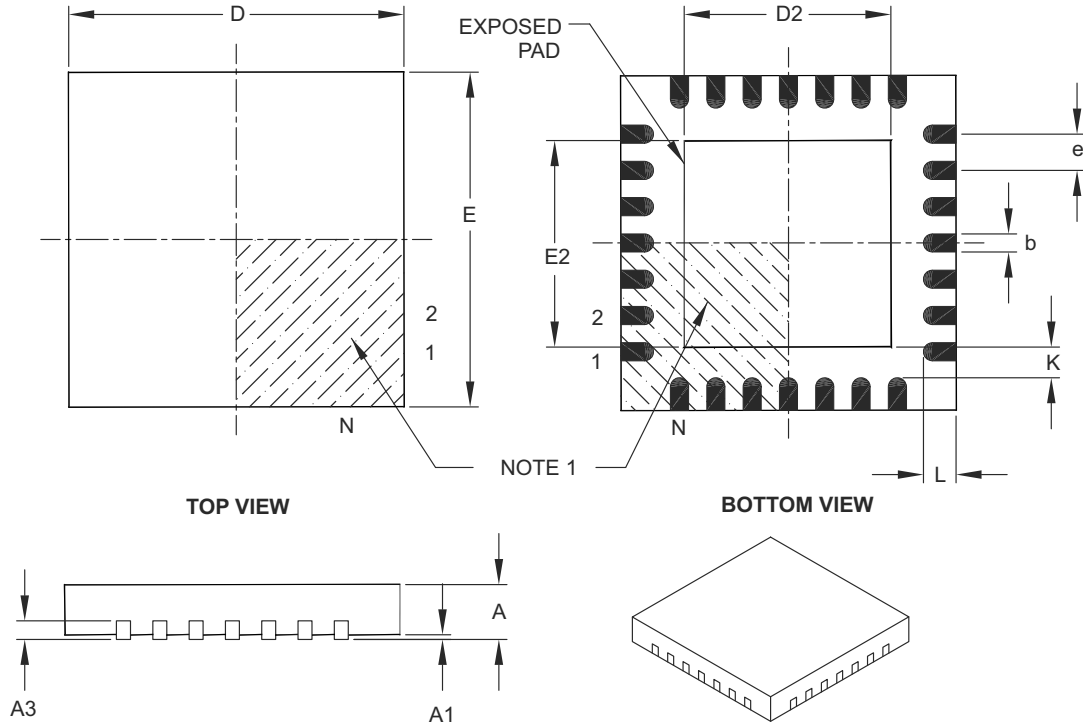
1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2052A

## 28-Lead Plastic Quad Flat, No Lead Package (ML) – 6x6 mm Body [QFN] with 0.55 mm Contact Length

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



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Number of Pins	N	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	4.20
Overall Length	D	6.00 BSC		
Exposed Pad Length	D2	3.65	3.70	4.20
Contact Width	b	0.23	0.30	0.35
Contact Length	L	0.50	0.55	0.70
Contact-to-Exposed Pad	K	0.20	–	–

**Notes:**

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- Package is saw singulated.
- Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

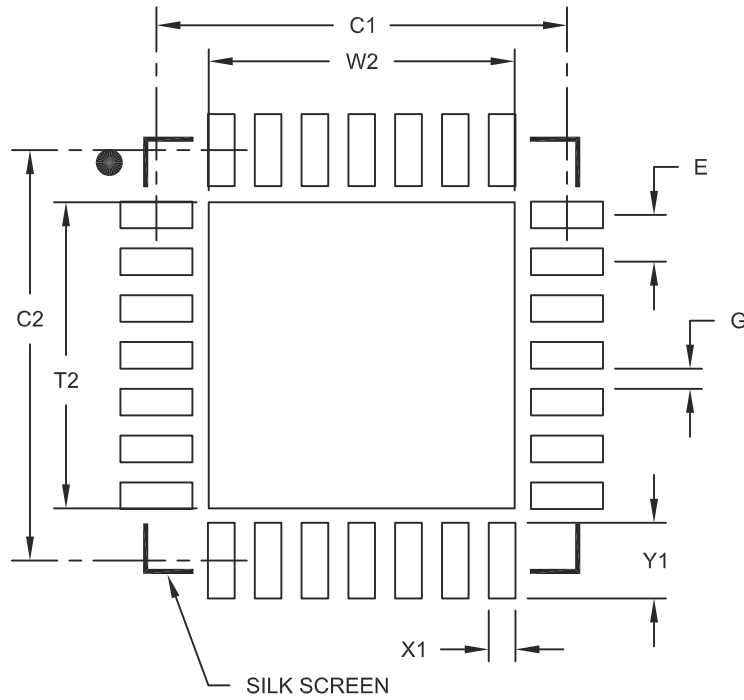
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-105B

# PIC24FJ16MC101/102

## 28-Lead Plastic Quad Flat, No Lead Package (ML) – 6x6 mm Body [QFN] with 0.55 mm Contact Length

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



RECOMMENDED LAND PATTERN

Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Contact Pitch	E		0.65 BSC	
Optional Center Pad Width	W2			4.25
Optional Center Pad Length	T2			4.25
Contact Pad Spacing	C1		5.70	
Contact Pad Spacing	C2		5.70	
Contact Pad Width (X28)	X1			0.37
Contact Pad Length (X28)	Y1			1.00
Distance Between Pads	G	0.20		

**Notes:**

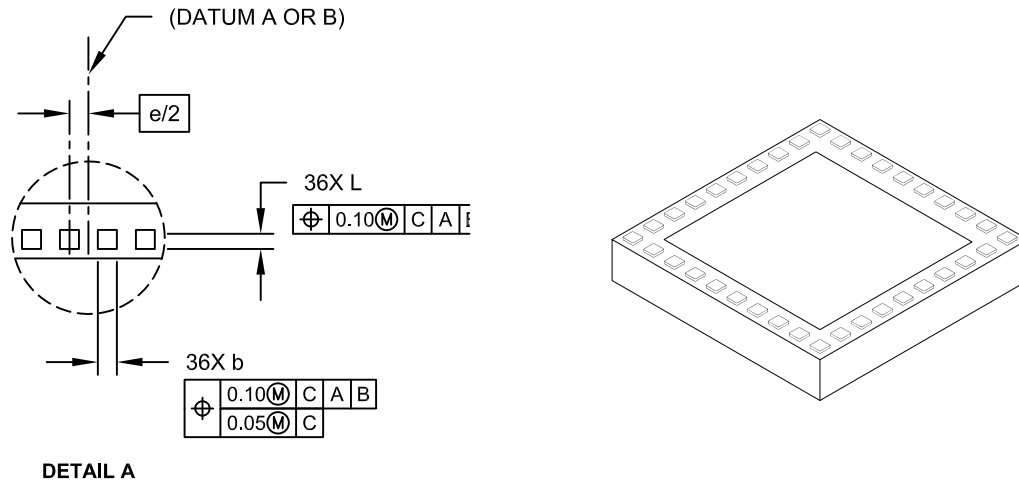
1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2105A

## 36-Lead Thermal Leadless Array Package (TL) – 5x5x0.9 mm Body with Exposed Pad [TLA]

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



Dimension	Units	MILLIMETERS		
		Limits	MIN	NOM
Number of Pins	N		36	
Number of Pins per Side	ND		10	
Number of Pins per Side	NE		8	
Pitch	e		0.50 BSC	
Overall Height	A	0.80	0.90	1.00
Standoff	A1	0.025	-	0.075
Overall Width	E		5.00 BSC	
Exposed Pad Width	E2	3.60	3.75	3.90
Overall Length	D		5.00 BSC	
Exposed Pad Length	D2	3.60	3.75	3.90
Contact Width	b	0.20	0.25	0.30
Contact Length	L	0.20	0.25	0.30
Contact-to-Exposed Pad	K	0.20	-	-

**Notes:**

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- Package is saw singulated.
- 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 C04-187B Sheet 2 of 2

# PIC24FJ16MC101/102

---

---

NOTES:

## APPENDIX A: REVISION HISTORY

### Revision A (February 2011)

This is the initial released version of this document.

### Revision B (June 2011)

This revision includes the following global updates:

- All JTAG references have been removed

All other major changes are referenced by their respective section in [Table A-1](#).

In addition, minor text and formatting changes were incorporated throughout the document.

**TABLE A-1: MAJOR SECTION UPDATES**

Section Name	Update Description
<a href="#">High-Performance, Ultra Low Cost 16-bit Microcontrollers</a>	The TMS, TDI, TDO, and TCK pin names were removed from these pin diagrams: <ul style="list-style-type: none"> <li>• 28-pin SPDIP/SOIC/SSOP</li> <li>• 28-pin QFN</li> <li>• 36-pin TLA</li> </ul>
<a href="#">Section 1.0 “Device Overview”</a>	Updated the Buffer Type to Digital for the CTED1 and CTED2 pins (see <a href="#">Table 1-1</a> ).
<a href="#">Section 4.0 “Memory Organization”</a>	Updated the SR and CORCON SFRs in the CPU Core Register Map (see <a href="#">Table 4-1</a> ).  Updated the SFR Address for IC2CON, IC3BUF, and IC3CON in the Input Capture Register Map (see <a href="#">Table 4-6</a> ).  Added the VREGS bit to the RCON register in the System Control Register Map (see <a href="#">Table 4-24</a> ).
<a href="#">Section 6.0 “Resets”</a>	Added the VREGS bit to the RCON register (see <a href="#">Register 6-1</a> ).
<a href="#">Section 8.0 “Oscillator Configuration”</a>	Updated the definition for COSC<2:0> = 001 and NOSC<2:0> = 001 in the OSCCON register (see <a href="#">Register 8-1</a> ).
<a href="#">Section 15.0 “Motor Control PWM Module”</a>	Updated the title for <a href="#">Example 15-1</a> to include a reference to the Assembly language.  Added <a href="#">Example 15-2</a> , which provides a C code version of the write-protected register unlock and fault clearing sequence.  Changed the bit PWMLOCK to PWMKEY in the PWM Key Unlock Register (see <a href="#">Register 15-15</a> ).
<a href="#">Section 19.0 “10-bit Analog-to-Digital Converter (ADC)”</a>	Updated the CH0 section and added Note 2 in both ADC block diagrams (see <a href="#">Figure 19-1</a> and <a href="#">Figure 19-2</a> ).  Updated the multiplexer values in the ADC Conversion Clock Period Block Diagram (see <a href="#">Figure 19-3</a> ).  Added the 01110 bit definitions and updated the 01101 bit definitions for the CH0SB<4:0> and CH0SA<4:0> bits in the AD1CHS0 register (see <a href="#">Register 19-5</a> ).
<a href="#">Section 22.0 “Charge Time Measurement Unit (CTMU)”</a>	Removed Section 22.1 “Measuring Capacitance”, Section 22.2 “Measuring Time”, and Section 22.3 “Pulse Generation and Delay”  Updated the key features.  Added the CTMU Block Diagram (see <a href="#">Figure 22-1</a> ).  Updated the ITRIM<5:0> bit definitions and added Note 1 to the CTMU Current Control register (see <a href="#">Register 22-3</a> ).

# PIC24FJ16MC101/102

**TABLE A-1: MAJOR SECTION UPDATES (CONTINUED)**

Section Name	Update Description
<p><b>Section 23.0 “Special Features”</b></p>	<p>Updated bits 5 and 4 of FPOR, modified Note 2, and removed Note 3 from the Configuration Shadow Register Map (see <a href="#">Table 23-1</a>).</p> <p>Updated bit 14 of CONFIG1 and removed Note 4 from the Configuration Flash Words (see <a href="#">Table 23-2</a>).</p> <p>Updated the PLLKEN Configuration bit description (see <a href="#">Table 23-3</a>).</p> <p>Added Note 3 to Connections for the On-Chip Voltage Regulator (see <a href="#">Figure 23-1</a>).</p>
<p><b>Section 26.0 “Electrical Characteristics”</b></p>	<p>Updated the Standard Operating Conditions to: 3.0V to 3.6V in all tables.</p> <p>Removed the Voltage on VCAP with respect to VSS entry in <a href="#">Absolute Maximum Ratings</a><sup>(1)</sup>.</p> <p>Updated the VDD Range (in Volts) in Operating MIPS vs. Voltage (see <a href="#">Table 26-1</a>).</p> <p>Removed parameter DC18 and updated the minimum value for parameter DC 10 in the DC Temperature and Voltage Specifications (see <a href="#">Table 26-4</a>).</p> <p>Updated the Characteristic definition and the Typical value for parameter BO10 in Electrical Characteristics: BOR (see <a href="#">Table 26-5</a>).</p> <p>Updated Note 2 in the DC Characteristics: Operating Current (IDD) (see <a href="#">Table 26-6</a>).</p> <p>Updated Note 2 in the DC Characteristics: Idle Current (IIDLE) (see <a href="#">Table 26-7</a>).</p> <p>Updated Note 2 and parameters DC60C and DC61a-DC61d in the DC Characteristics: Power-Down Current (IPD) (see <a href="#">Table 26-8</a>).</p> <p>Updated Note 2 in the DC Characteristics: Doze Current (IDOZE) (see <a href="#">Table 26-9</a>).</p> <p>Added Note 1 to the Internal Voltage Regulator Specifications (see <a href="#">Table 26-13</a>).</p> <p>Updated the Minimum and Maximum values for parameter F20a and the Typical value for parameter F20b in AC Characteristics: Internal Fast RC (FRC) Accuracy (see <a href="#">Table 26-18</a>).</p> <p>Updated the Minimum, Typical, and Maximum values for parameters F21a and F21b in Internal Low-Power RC (LPRC) Accuracy (see <a href="#">Table 26-19</a>).</p> <p>Updated the Minimum, Typical, and Maximum values for parameter D305 in the Comparator Module Specifications (see <a href="#">Table 26-43</a>).</p> <p>Added parameters CTMUFV1 and CTMUFV2 and updated Note 1 and the Conditions for all parameters in the CTMU Current Source Specifications (see <a href="#">Table 26-46</a>).</p> <p>Added Forward Voltage Versus Temperature (see <a href="#">Figure 26-25</a>).</p>



## INDEX

### A

AC Characteristics .....	244
Internal Fast RC (FRC) Accuracy .....	246
Internal Low-Power RC (LPRC) Accuracy .....	246
Load Conditions .....	244
ADC .....	
Initialization .....	173
Key Features .....	173
ADC Module .....	
ADC1 Register Map .....	37
ADC11 Register Map .....	38
Alternate Interrupt Vector Table (AIVT) .....	63
Analog-to-Digital Converter (ADC) .....	173
Arithmetic Logic Unit (ALU) .....	26
Assembler .....	
MPASM Assembler .....	232

### B

Block Diagrams .....	
16-bit Timer1 Module .....	123
Comparator I/O Operating Modes .....	186
Comparator Voltage Reference .....	187
Connections for On-Chip Voltage Regulator .....	220
CTMU Configurations .....	
Time Measurement .....	210
Device Clock .....	93
Digital Filter Interconnect .....	188
Input Capture .....	131
Output Compare .....	133
PIC24FJ12MC201/202 .....	12
PIC24FJ16MC101/102 CPU Core .....	22
PWM Module .....	138
Reset System .....	55
Shared Port Structure .....	107
SPI .....	153
Timer2 (16-bit) .....	127
Timer2/3 (32-bit) .....	126
UART .....	167
User Programmable Blanking Function .....	187
Watchdog Timer (WDT) .....	221

### C

C Compilers .....	
MPLAB C18 .....	232
Charge Time Measurement Unit. See CTMU. ....	
Clock Switching .....	100
Enabling .....	100
Sequence .....	100
Code Examples .....	
Port Write/Read .....	108
PWRSAV Instruction Syntax .....	101
Code Protection .....	215, 222
Configuration Bits .....	215
Configuration Register Map .....	216
Configuring Analog Port Pins .....	108
CPU .....	
Control Register .....	24
CPU Clocking System .....	94
PLL Configuration .....	95
Selection .....	94
Sources .....	94
CTMU Module .....	
Register Map .....	39

Customer Change Notification Service .....	301
Customer Notification Service .....	301
Customer Support .....	301

### D

Data Address Space .....	29
Alignment .....	29
Memory Map for PIC24FJ16MC101/102 Devices with 1 KB RAM .....	30
Near Data Space .....	29
Software Stack .....	44
Width .....	29
DC Characteristics .....	236
BOR .....	237
I/O Pin Input Specifications .....	242
I/O Pin Output Specifications .....	242
Idle Current (I <sub>DOZE</sub> ) .....	241
Idle Current (I <sub>IDLE</sub> ) .....	239
Operating Current (I <sub>DD</sub> ) .....	238
Power-Down Current (I <sub>Pd</sub> ) .....	240
Program Memory .....	243
Temperature and Voltage Specifications .....	237
Development Support .....	231
Doze Mode .....	102

### E

Electrical Characteristics .....	235
AC .....	244
Equations .....	
Device Operating Frequency .....	94
Errata .....	10

### F

Flash Program Memory .....	51
Control Registers .....	52
Operations .....	52
Programming Algorithm .....	52
RTSP Operation .....	52
Table Instructions .....	51
Flexible Configuration .....	215

### I

I/O Ports .....	107
Parallel I/O (PIO) .....	107
Write/Read Timing .....	108
I <sup>2</sup> C .....	
Addresses .....	160
Operating Modes .....	159
Registers .....	159
I <sup>2</sup> C Module .....	
I2C1 Register Map .....	35
In-Circuit Debugger .....	222
In-Circuit Emulation .....	215
In-Circuit Serial Programming (ICSP) .....	215, 222
Input Capture .....	131
Registers .....	132
Input Change Notification .....	108
Instruction Addressing Modes .....	44
File Register Instructions .....	44
Fundamental Modes Supported .....	45
MCU Instructions .....	44
Move and Accumulator Instructions .....	45
Other Instructions .....	45
Instruction Set .....	

# PIC24FJ16MC101/102

Overview .....	226	Table Instructions .....	48
Summary .....	223	Data Access from, Address Generation .....	47
Instruction-Based Power-Saving Modes .....	101	Memory Map .....	27
Idle .....	102	Table Read Instructions	
Sleep .....	101	TBLRDH .....	48
Internal RC Oscillator		TBLRDL .....	48
Use with WDT .....	221	Visibility Operation .....	49
Internet Address .....	301	Program Memory	
Interrupt Control and Status Registers .....	66	Interrupt Vector .....	28
IECx .....	66	Organization .....	28
IFSx .....	66	Reset Vector .....	28
INTCON1 .....	66	PWM Time Base .....	141
INTCON2 .....	66		
IPCx .....	66	<b>R</b>	
Interrupt Setup Procedures .....	91	Reader Response .....	302
Initialization .....	91	Register Map	
Interrupt Disable .....	91	Real-Time Clock and Calendar .....	39
Interrupt Service Routine .....	91	Register Maps	
Trap Service Routine .....	91	Comparator .....	40
Interrupt Vector Table (IVT) .....	63	Registers	
Interrupts Coincident with Power Save Instructions .....	102	AD1CHS123 (ADC1 Input Channel 1, 2, 3 Select) .....	181
<b>M</b>		ADxCHS0 (ADCx Input Channel 0 Select) .....	182
Memory Organization .....	27	ADxCON1 (ADCx Control 1) .....	177
Microchip Internet Web Site .....	301	ADxCON2 (ADCx Control 2) .....	179
Motor Control PWM .....	137	ADxCON3 (ADCx Control 3) .....	180
Motor Control PWM Module		ADxCSSL (ADCx Input Scan Select Low) .....	183
6-Output Register Map .....	35	ADxPCFGL (ADCx Port Configuration Low) .....	184
MPLAB ASM30 Assembler, Linker, Librarian .....	232	CLKDIV (Clock Divisor) .....	98
MPLAB Integrated Development Environment Software ..	231	CMSTAT (Comparator Status) .....	189
MPLAB PM3 Device Programmer .....	234	CMxCON (Comparator Control) .....	190
MPLAB REAL ICE In-Circuit Emulator System .....	233	CMxFLTR (Comparator Filter Control) .....	196
MPLINK Object Linker/MPLIB Object Librarian .....	232	CMxMSKCON (Comparator Mask Gating Control) ..	194
Multi-Bit Data Shifter .....	26	CMxMSKSRC (Comparator Mask Source Control) ..	192
<b>N</b>		CORCON (Core Control) .....	25, 67
NVM Module		CTMUCON (CTMU Control) .....	211, 212
Register Map .....	43	CTMUCON1 (CTMU Control Register 1) .....	211
<b>O</b>		CTMUCON1 (CTMU Control Register 2) .....	212
Open-Drain Configuration .....	108	CTMUICON (CTMU Current Control) .....	213
Output Compare .....	133	CVRCON (Comparator Voltage Reference Control) ..	197
<b>P</b>		DEVID (Device ID) .....	219
Packaging .....	277	DEVREV (Device Revision) .....	219
Details .....	279	I2CxCON (I2Cx Control) .....	161
Marking .....	277, 278	I2CxMSK (I2Cx Slave Mode Address Mask) .....	165
PAD Configuration		I2CxSTAT (I2Cx Status) .....	163
Register Map .....	39	IEC0 (Interrupt Enable Control 0) .....	75
Peripheral Module Disable (PMD) .....	102	IEC1 (Interrupt Enable Control 1) .....	77
Pinout I/O Descriptions (table) .....	13	IEC2 (Interrupt Enable Control 2) .....	78
PMD Module		IEC3 (Interrupt Enable Control 3) .....	78
Register Map .....	43	IEC4 (Interrupt Enable Control 4) .....	79
PORTA		IFS0 (Interrupt Flag Status 0) .....	70
Register Map .....	42	IFS1 (Interrupt Flag Status 1) .....	72
PORTB		IFS2 (Interrupt Flag Status 2) .....	73
Register Map for PIC24FJ16MC101 .....	42	IFS3 (Interrupt Flag Status 3) .....	73
Register Map for PIC24FJ16MC102 .....	42	IFS4 (Interrupt Flag Status 4) .....	74
Power-on Reset (POR) .....	59	INTCON1 (Interrupt Control 1) .....	68
Power-Saving Features .....	101	INTCON2 (Interrupt Control 2) .....	69
Clock Frequency and Switching .....	101	INTTREG Interrupt Control and Status Register .....	90
Program Address Space .....	27	IPC0 (Interrupt Priority Control 0) .....	80
Construction .....	46	IPC1 (Interrupt Priority Control 1) .....	81
Data Access from Program Memory Using		IPC14 (Interrupt Priority Control 14) .....	86
Program Space Visibility .....	49	IPC15 (Interrupt Priority Control 15) .....	87
Data Access from Program Memory Using		IPC16 (Interrupt Priority Control 16) .....	88
		IPC19 (Interrupt Priority Control 19) .....	89
		IPC2 (Interrupt Priority Control 2) .....	82
		IPC3 (Interrupt Priority Control 3) .....	83
		IPC4 (Interrupt Priority Control 4) .....	84

# PIC24FJ16MC101/102

IPC5 (Interrupt Priority Control 5) .....	85	Software Reset Instruction (SWR).....	60
IPC7 (Interrupt Priority Control 7) .....	85	Software Simulator (MPLAB SIM) .....	233
IPC9 (Interrupt Priority Control 9) .....	86	Software Stack Pointer, Frame Pointer	
NVMCON (Flash Memory Control) .....	53	CALLL Stack Frame .....	44
NVMKEY (Nonvolatile Memory Key) .....	54	Special Features of the CPU .....	215
OCxCON (Output Compare x Control) .....	135	Special MCU Features.....	21
OSCCON (Oscillator Control) .....	96	SPI Module	
OSCTUN (FRC Oscillator Tuning) .....	99	SPI1 Register Map .....	36
PMD1 (Peripheral Module Disable		Symbols Used in Opcode Descriptions .....	224
Control Register 1).....	103	System Control	
PMD2 (Peripheral Module Disable		Register Map .....	43
Control Register 2).....	104	<b>T</b>	
PMD3 (Peripheral Module Disable		Temperature and Voltage Specifications	
Control Register 3).....	105	AC.....	244
PMD4 (Peripheral Module Disable		Timer1 .....	123
Control Register 4).....	105	Timer2/3 .....	125
PWMxCON1 (PWM Control 1).....	144	Timing Characteristics	
PWMxCON2 (PWM Control 2).....	145	CLKO and I/O .....	247
PWMxKEY (PWM Key Unlock Register) .....	152	Timing Diagrams	
PxDC1 (PWM Duty Cycle 1) .....	151	10-bit ADC Conversion (CHPS<1:0> = 01, SIMSAM = 0,	
PxDC2 (PWM Duty Cycle 2) .....	151	ASAM = 0, SSRC<2:0> = 000).....	272
PxDC3 (PWM Duty Cycle 3) .....	151	10-bit ADC Conversion (CHPS<1:0> = 01, SIMSAM = 0,	
PxDTCN1 (Dead-Time Control 1) .....	146	ASAM = 1, SSRC<2:0> = 111,	
PxDTCN2 (Dead-Time Control 2) .....	147	SAMC<4:0> = 00001).....	273
PxFLTAON (Fault A Control).....	148, 149	ADC Conversion Timing Characteristics	
PxOVDCON (Override Control) .....	150	(CHPS<1:0> = 01, SIMSAM = 0, ASAM = 1,	
PxSECMP (Special Event Compare).....	143	SSRC<2:0> = 111, SAMC<4:0> = 00001) .....	272
PxTCON (PWM Time Base Control).....	141	Brown-out Situations .....	59
PxTMR (PWM Timer Count Value).....	142	External Clock .....	245
PxTPER (PWM Time Base Period) .....	142	I2Cx Bus Data (Master Mode) .....	266
RCON (Reset Control) .....	56	I2Cx Bus Data (Slave Mode) .....	268
RPINR0 (Peripheral Pin Select Input Register 0) .....	112	I2Cx Bus Start/Stop Bits (Master Mode).....	266
RPINR1 (Peripheral Pin Select Input Register 1) .....	112	I2Cx Bus Start/Stop Bits (Slave Mode).....	268
RPINR11 (Peripheral Pin Select Input Register 11) .....	115	Input Capture (CAPx) .....	251
RPINR18 (Peripheral Pin Select Input Register 18) .....	116	Motor Control PWM .....	253
RPINR21 (Peripheral Pin Select Input Register 21) .....	117	Motor Control PWM Fault .....	253
RPINR3 (Peripheral Pin Select Input Register 3) .....	113	OC/PWM .....	252
RPINR7 (Peripheral Pin Select Input Register 7) .....	114	Output Compare (OCx) .....	252
RPINR8 (Peripheral Pin Select Input Register 8) .....	115	Reset, Watchdog Timer, Oscillator Start-up Timer	
RPOR0 (Peripheral Pin Select Output Register 0) .....	118	and Power-up Timer.....	248
RPOR1 (Peripheral Pin Select Output Register 1) .....	118	Timer1, 2 and 3 External Clock .....	249
RPOR2 (Peripheral Pin Select Output Register 2) .....	119	Timing Requirements	
RPOR3 (Peripheral Pin Select Output Register 3) .....	119	CLKO and I/O .....	247
RPOR4 (Peripheral Pin Select Output Register 4) .....	120	External Clock .....	245
RPOR5 (Peripheral Pin Select Output Register 5) .....	120	Input Capture.....	251
RPOR6 (Peripheral Pin Select Output Register 6) .....	121	Timing Specifications	
RPOR7 (Peripheral Pin Select Output Register 7) .....	121	10-bit ADC Requirements.....	273
SPIxCON1 (SPIx Control 1).....	155	I2Cx Bus Data Requirements (Master Mode).....	267
SPIxCON2 (SPIx Control 2).....	157	I2Cx Bus Data Requirements (Slave Mode).....	269
SPIxSTAT (SPIx Status and Control) .....	154	Motor Control PWM Requirements.....	253
SR (CPU Status).....	24, 67	Output Compare Requirements.....	252
T1CON (Timer1 Control).....	124	PLL Clock .....	246
T2CON Control .....	128	Reset, Watchdog Timer, Oscillator Start-up Timer,	
T3CON Control .....	129	Power-up Timer and Brown-out Reset	
TCxCON (Input Capture x Control).....	132	Requirements .....	248
UxMODE (UARTx Mode).....	168	Simple OC/PWM Mode Requirements .....	252
UxSTA (UARTx Status and Control).....	170	Timer1 External Clock Requirements .....	249
Reset		Timer2 External Clock Requirements .....	250
Illegal Opcode .....	55, 61	Timer3 External Clock Requirements .....	250
Trap Conflict.....	60	<b>U</b>	
Uninitialized W Register.....	55, 61	UART Module	
Reset Sequence .....	63	UART1 Register Map .....	36
Resets.....	55	Universal Asynchronous Receiver Transmitter (UART) .....	167
<b>S</b>		Using the RCON Status Bits.....	61
Serial Peripheral Interface (SPI) .....	153		

# PIC24FJ16MC101/102

---

## V

Voltage Regulator (On-Chip)..... 220

## W

Watchdog Time-out Reset (WDTR) ..... 60

Watchdog Timer (WDT) ..... 215, 221

    Programming Considerations ..... 221

WWW Address..... 301

WWW, On-Line Support..... 10

## THE MICROCHIP WEB SITE

Microchip provides online support via our WWW site at [www.microchip.com](http://www.microchip.com). This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQs), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

## CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at [www.microchip.com](http://www.microchip.com). Under "Support", click on "Customer Change Notification" and follow the registration instructions.

## CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support
- Development Systems Information Line

Customers should contact their distributor, representative or field application engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

**Technical support is available through the web site at: <http://microchip.com/support>**

# PIC24FJ16MC101/102

---

---

## READER RESPONSE

It is our intention to provide you with the best documentation possible to ensure successful use of your Microchip product. If you wish to provide your comments on organization, clarity, subject matter, and ways in which our documentation can better serve you, please FAX your comments to the Technical Publications Manager at (480) 792-4150.

Please list the following information, and use this outline to provide us with your comments about this document.

TO: Technical Publications Manager Total Pages Sent \_\_\_\_\_

RE: Reader Response

From: Name \_\_\_\_\_

Company \_\_\_\_\_

Address \_\_\_\_\_

City / State / ZIP / Country \_\_\_\_\_

Telephone: (\_\_\_\_\_) \_\_\_\_\_ - \_\_\_\_\_ FAX: (\_\_\_\_\_) \_\_\_\_\_ - \_\_\_\_\_

Application (optional):

Would you like a reply?  Y  N

Device: PIC24FJ16MC101/102

Literature Number: DS39997B

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.

PIC 24 FJ 16 MC1 02 T E / SP - XXX	
<b>Microchip Trademark</b>	_____
<b>Architecture</b>	_____
<b>Flash Memory Family</b>	_____
<b>Program Memory Size (KB)</b>	_____
<b>Product Group</b>	_____
<b>Pin Count</b>	_____
<b>Tape and Reel Flag (if applicable)</b>	_____
<b>Temperature Range</b>	_____
<b>Package</b>	_____
<b>Pattern</b>	_____

Architecture:	24	=	16-bit Microcontroller
Flash Memory Family:	FJ	=	Flash program memory, 3.3V
Product Group:	MC1	=	Motor Control family
Pin Count:	01	=	18-pin and 20-pin
	02	=	28-pin and 32-pin
Temperature Range:	I	=	-40° C to+85° C (Industrial)
	E	=	-40° C to+125° C (Extended)
Package:	P	=	Plastic Dual In-Line - 300 mil body (PDIP)
	SS	=	Plastic Shrink Small Outline -5.3 mm body (SSOP)
	SP	=	Skinny Plastic Dual In-Line - 300 mil body (SPDIP)
	SO	=	Plastic Small Outline - Wide, 7.50 mil body (SOIC)
	ML	=	Plastic Quad, No Lead Package - (28-pin) 6x6 mm body (QFN)
	TL	=	Thermal Leadless Array Package - (36-pin) 5x5 mm body (TLA)

### Examples:

- a) PIC24FJ16MC102-E/SP:  
Motor Control PIC24, 16 KB program memory, 28-pin, Extended temperature, SPDIP package.



# MICROCHIP

## Worldwide Sales and Service

### AMERICAS

**Corporate Office**  
2355 West Chandler Blvd.  
Chandler, AZ 85224-6199  
Tel: 480-792-7200  
Fax: 480-792-7277  
Technical Support:  
<http://www.microchip.com/support>  
Web Address:  
[www.microchip.com](http://www.microchip.com)

**Atlanta**  
Duluth, GA  
Tel: 678-957-9614  
Fax: 678-957-1455

**Boston**  
Westborough, MA  
Tel: 774-760-0087  
Fax: 774-760-0088

**Chicago**  
Itasca, IL  
Tel: 630-285-0071  
Fax: 630-285-0075

**Cleveland**  
Independence, OH  
Tel: 216-447-0464  
Fax: 216-447-0643

**Dallas**  
Addison, TX  
Tel: 972-818-7423  
Fax: 972-818-2924

**Detroit**  
Farmington Hills, MI  
Tel: 248-538-2250  
Fax: 248-538-2260

**Indianapolis**  
Noblesville, IN  
Tel: 317-773-8323  
Fax: 317-773-5453

**Los Angeles**  
Mission Viejo, CA  
Tel: 949-462-9523  
Fax: 949-462-9608

**Santa Clara**  
Santa Clara, CA  
Tel: 408-961-6444  
Fax: 408-961-6445

**Toronto**  
Mississauga, Ontario,  
Canada  
Tel: 905-673-0699  
Fax: 905-673-6509

### ASIA/PACIFIC

**Asia Pacific Office**  
Suites 3707-14, 37th Floor  
Tower 6, The Gateway  
Harbour City, Kowloon  
Hong Kong  
Tel: 852-2401-1200  
Fax: 852-2401-3431

**Australia - Sydney**  
Tel: 61-2-9868-6733  
Fax: 61-2-9868-6755

**China - Beijing**  
Tel: 86-10-8569-7000  
Fax: 86-10-8528-2104

**China - Chengdu**  
Tel: 86-28-8665-5511  
Fax: 86-28-8665-7889

**China - Chongqing**  
Tel: 86-23-8980-9588  
Fax: 86-23-8980-9500

**China - Hangzhou**  
Tel: 86-571-2819-3180  
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