

PIC16(L)F1825/1829 Data Sheet

14/20-Pin Flash Microcontrollers with XLP Technology

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in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates.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 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, PIC³² 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.

© 2010-2012, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

Printed on recycled paper.

QUALITY MANAGEMENT SYSTEM CERTIFIED BY DNV

= ISO/TS 16949=

ISBN: 9781620764039

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.



14/20-Pin Flash Microcontrollers with XLP Technology

High-Performance RISC CPU:

- Only 49 Instructions to Learn:
 - All single-cycle instructions except branches
- · Operating Speed:
 - DC 32 MHz oscillator/clock input
 - DC 125 ns instruction cycle
- Up to 16 Kbytes Linear Program Memory Addressing
- Up to 1024 bytes Linear Data Memory Addressing
- Interrupt Capability with Automatic Context Saving
- 16-Level Deep Hardware Stack with Optional Overflow/Underflow Reset
- · Direct, Indirect and Relative Addressing modes:
 - Two full 16-bit File Select Registers (FSRs)
 - FSRs can read program and data memory

Flexible Oscillator Structure:

- Precision 32 MHz Internal Oscillator Block:
 - Factory calibrated to ± 1%, typical
 - Software selectable frequencies range of 31 kHz to 32 MHz
- · 31 kHz Low-Power Internal Oscillator
- · Four Crystal modes up to 32 MHz
- Three External Clock modes up to 32 MHz
- 4x Phase Lock Loop (PLL)
- · Fail-Safe Clock Monitor:
 - Allows for safe shutdown if peripheral clock stops
- · Two-Speed Oscillator Start-Up
- · Reference Clock Module:
 - Programmable clock output frequency and duty cycle

Special Microcontroller Features:

- 1.8V-5.5V Operation PIC16F1825/1829
- 1.8V-3.6V Operation PIC16LF1825/1829
- · Self-Programmable under Software Control
- Power-on Reset (POR), Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
- Programmable Brown-out Reset (BOR)
- Extended Watchdog Timer (WDT)
- In-Circuit Serial Programming[™] (ICSP[™]) via two pins
- · In-Circuit Debug (ICD) via Two Pins
- Enhanced Low-Voltage Programming (LVP)
- · Programmable Code Protection
- · Power-Saving Sleep mode

Extreme Low-Power Management PIC16LF1825/1829 with XLP:

- Sleep mode: 20 nA @ 1.8V, typical
- · Watchdog Timer: 300 nA @ 1.8V, typical
- Timer1 Oscillator: 650 nA @ 32 kHz, 1.8V, typical
- Operating Current: 48 μA/MHz @ 1.8V, typical

Analog Features:

- Analog-to-Digital Converter (ADC) Module:
 - 10-bit resolution, up to 12 channels
 - Auto acquisition capability
 - Conversion available during Sleep
- · Analog Comparator Module:
 - Two rail-to-rail analog comparators
 - Power mode control
 - Software controllable hysteresis
- · Voltage Reference Module:
 - Fixed Voltage Reference (FVR) with 1.024V, 2.048V and 4.096V output levels
 - 5-bit rail-to-rail resistive DAC with positive and negative reference selection

Peripheral Highlights:

- Up to 17 I/O Pins and 1 Input Only Pin:
 - High current sink/source 25 mA/25 mA
 - Programmable weak pull-ups
 - Programmable interrupt-on-change pins
- · Timer0: 8-Bit Timer/Counter with 8-Bit Prescaler
- Enhanced Timer1:
 - 16-bit timer/counter with prescaler
- External Gate Input mode
- Dedicated, low-power 32 kHz oscillator driver
- Three Timer2-types: 8-Bit Timer/Counter with 8-Bit Period Register. Prescaler and Postscaler
- Two Capture, Compare, PWM (CCP) Modules
- · Two Enhanced CCP (ECCP) Modules:
 - Software selectable time bases
 - Auto-shutdown and auto-restart
 - PWM steering
- Up to Two Master Synchronous Serial Port (MSSP) with SPI and I²C[™] with:
 - 7-bit address masking
 - SMBus/PMBus[™] compatibility
- Enhanced Universal Synchronous Asynchronous Receiver Transmitter (EUSART) Module
- mTouch™ Sensing Oscillator Module:
 - Up to 12 input channels

Peripheral Highlights (Continued):

- · Data Signal Modulator Module:
 - Selectable modulator and carrier sources
- · SR Latch:
 - Multiple Set/Reset input options
 - Emulates 555 Timer applications

PIC12(L)F1822/1840/PIC16(L)F182x/1847 Family Types

· · · · · · · · · · · · · · · · · · ·	1012(2)1 1022/1040/11010(2)1 102X/1047 1 diffing 1 ypcs														
Device	Data Sheet Index	Program Memory Flash (words)	Data EEPROM (bytes)	Data SRAM (bytes)	I/O'S ⁽²⁾	10-bit ADC (ch)	CapSense (ch)	Comparators	Timers (8/16-bit)	EUSART	MSSP (I ² C TM /SPI)	ECCP (Full-Bridge) ECCP (Half-Bridge) CCP	SR Latch	Debug ⁽¹⁾	XLP
PIC12(L)F1822	(1)	2K	256	128	6	4	4	1	2/1	1	1	0/1/0	Υ	I/H	Υ
PIC12(L)F1840	(2)	4K	256	256	6	4	4	1	2/1	1	1	0/1/0	Υ	I/H	Υ
PIC16(L)F1823	(1)	2K	256	128	12	8	8	2	2/1	1	1	1/0/0	Υ	I/H	Υ
PIC16(L)F1824	(3)	4K	256	256	12	8	8	2	4/1	1	1	1/1/2	Υ	I/H	Υ
PIC16(L)F1825	(4)	8K	256	1024	12	8	8	2	4/1	1	1	1/1/2	Υ	I/H	Υ
PIC16(L)F1826	(5)	2K	256	256	16	12	12	2	2/1	1	1	1/0/0	Υ	I/H	Υ
PIC16(L)F1827	(5)	4K	256	384	16	12	12	2	4/1	1	2	1/1/2	Υ	I/H	Υ
PIC16(L)F1828	(3)	4K	256	256	18	12	12	2	4/1	1	1	1/1/2	Υ	I/H	Υ
PIC16(L)F1829	(4)	8K	256	1024	18	12	12	2	4/1	1	2	1/1/2	Υ	I/H	Υ
PIC16(L)F1847	(6)	8K	256	1024	16	12	12	2	4/1	1	2	1/1/2	Υ	I/H	Υ

Note 1: I - Debugging, Integrated on Chip; H - Debugging, available using Debug Header.

2: One pin is input-only.

Data Sheet Index: (Unshaded devices are described in this document.)

1: DS41413 PIC12(L)F1822/PIC16(L)F1823 Data Sheet, 8/14-Pin Flash Microcontrollers.

2: DS41441 PIC12(L)F1840 Data Sheet, 8-Pin Flash Microcontrollers.

3: DS41419 PIC16(L)F1824/1828 Data Sheet, 28/40/44-Pin Flash Microcontrollers.

4: DS41440 PIC16(L)F1825/1829 Data Sheet, 14/20-Pin Flash Microcontrollers.

5: DS41391 PIC16(L)F1826/1827 Data Sheet, 18/20/28-Pin Flash Microcontrollers.

6: DS41453 PIC16(L)F1847 Data Sheet, 18/20/28-Pin Flash Microcontrollers.

Note: For other small form-factor package availability and marking information, please visit www.microchip.com/packaging or contact your local sales office.

FIGURE 1: 14-PIN DIAGRAM FOR PIC16(L)F1825

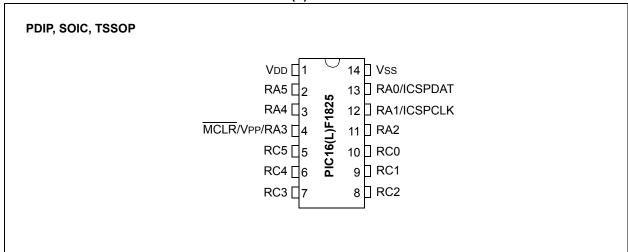


FIGURE 2: 16-PIN DIAGRAM FOR PIC16(L)F1825

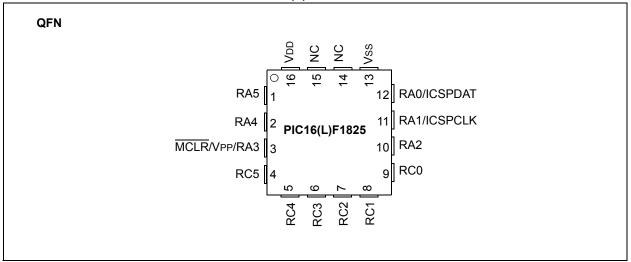
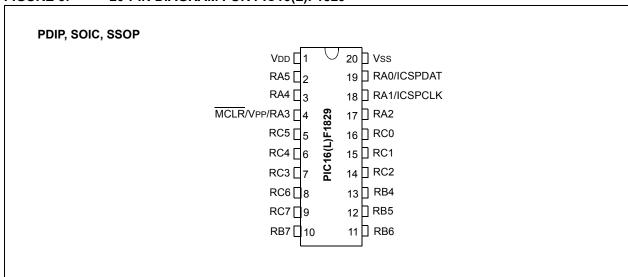


TABLE 1: 14-PIN AND 16-PIN ALLOCATION TABLE (PIC16(L)F1825)

										(–)	,				
I/O	14-Pin PDIP/SOIC/TSSOP	16-Pin QFN	A/D	Reference	Cap Sense	Comparator	SR Latch	Timers	ECCP	EUSART	MSSP	Interrupt	Modulator	Pull-up	Basic
RA0	13	12	AN0	VREF- DACOUT	CPS0	C1IN+	_	_	1	TX ⁽¹⁾ CK ⁽¹⁾	1	IOC	_	Υ	ICSPDAT ICDDAT
RA1	12	11	AN1	VREF+	CPS1	C12IN0-	SRI		ı	RX ⁽¹⁾ DT ⁽¹⁾	l	IOC	_	Υ	ICSPCLK ICDCLK
RA2	11	10	AN2		CPS2	C1OUT	SRQ	T0CKI	CCP3 FLT0	_	-	INT/ IOC		Υ	_
RA3	4	3	_	_		_	_	T1G ⁽¹⁾	-	_	SS1 ⁽¹⁾	IOC	_	Υ	MCLR VPP
RA4	3	2	AN3	_	CPS3	_	_	T1G ⁽¹⁾ T1OSO	P2B ⁽¹⁾	_	SDO1 ⁽¹⁾	IOC	_	Υ	OSC2 CLKOUT CLKR
RA5	2	1	-		1	l	_	T1CKI T1OSI	CCP2 P2A ⁽¹⁾	l	l	IOC		Υ	OSC1 CLKIN
RC0	10	9	AN4		CPS4	C2IN+	_	1	P1D ⁽¹⁾	l	SCL SCK	1		Υ	I
RC1	9	8	AN5		CPS5	C12IN1-	_	1	CCP4 P1C ⁽¹⁾	l	SDA SDI	1	_	Υ	l
RC2	8	7	AN6	1	CPS6	C12IN2-	_	1	P1D ⁽¹⁾ P2B ⁽¹⁾	l	SDO1 ⁽¹⁾	1	MDCIN1	Y	ı
RC3	7	6	AN7		CPS7	C12IN3-	_		CCP2 ⁽¹⁾ P1C ⁽¹⁾ P2A ⁽¹⁾	l	SS1 ⁽¹⁾	1	MDMIN	Υ	1
RC4	6	5	_	_	_	C2OUT	SRNQ	_	P1B	TX ⁽¹⁾ CK ⁽¹⁾		_	MDOUT	Υ	_
RC5	5	4	_	_	ı	_	_	_	CCP1 P1A	RX ⁽¹⁾ DT ⁽¹⁾			MDCIN2	Υ	
VDD	1	16	_	_	_	_	_	_	_	_	_	_	_	_	VDD
Vss	14	13	_	_	_	_	_	_	_	_	_	_	_	_	Vss

Note 1: Pin function is selectable via the APFCON0 or APFCON1 register.

FIGURE 3: 20-PIN DIAGRAM FOR PIC16(L)F1829





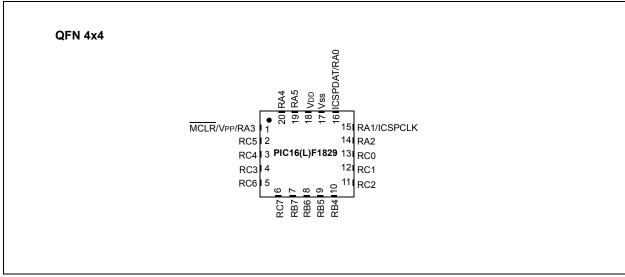


TABLE 2: 20-PIN ALLOCATION TABLE (PIC16(L)F1829)

0/1	20-Pin PDIP/SOIC/SSOP	20-Pin QFN	A/D	Reference	Cap Sense	Comparator	SR Latch	Timers	CCP	EUSART	SSP	Interrupt	Modulator	Pull-up	Basic
RA0	19	16	AN0	VREF- DACOUT	CPS0	C1IN+		-	_	_	_	IOC	_	Υ	ICSPDAT/ ICDDAT
RA1	18	15	AN1	VREF+	CPS1	C12IN0-	SRI	_	_	_	_	IOC	_	Υ	ICSPCLK/ ICDCLK
RA2	17	14	AN2	_	CPS2	C1OUT	SRQ	T0CKI	CCP3 FLT0	_	_	INT/ IOC	_	Υ	_
RA3	4	1		_	_	_		T1G ⁽¹⁾	_	_	_	IOC	_	Y ⁽⁴⁾	MCLR VPP
RA4	3	20	AN3	_	CPS3	_	1	T1G ⁽¹⁾ T1OSO	P2B ⁽¹⁾	_	SS2 ⁽¹⁾	IOC	_	Y	OSC2 CLKOUT CLKR
RA5	2	19	_	_	_	_	_	T1CKI T1OSI	CCP2 ⁽¹⁾ P2A ⁽¹⁾	_	SDO2 ⁽¹⁾	IOC	_	Y	OSC1 CLKIN
RB4	13	10	AN10	_	CPS10	_		_	_	_	SDA1 SDI1	IOC	_	Υ	_
RB5	12	9	AN11	_	CPS11	_	_	_	_	RX ⁽¹⁾ DT ⁽¹⁾	SDA2 SDI2	IOC	_	Y	_
RB6	11	8		_		_			_	_	SCL1 SCK1	IOC	_	Y	_
RB7	10	7	_	_	_	_	_		_	TX ⁽¹⁾ CK ⁽¹⁾	SCL2 SCK2	IOC	_	Y	_
RC0	16	13	AN4	_	CPS4	C2IN+	_	_	P1D ⁽¹⁾	_	SS2 ⁽¹⁾	_	_	Υ	_
RC1	15	12	AN5	_	CPS5	C12IN1-	_	_	P1C ⁽¹⁾	_	SDO2 ⁽¹⁾	_	_	Υ	_
RC2	14	11	AN6	_	CPS6	C12IN2-		_	P1D ⁽¹⁾ P2B ⁽¹⁾	_	_	_	MDCIN1	Υ	_
RC3	7	4	AN7	_	CPS7	C12IN3-		_	P1C ⁽¹⁾ CCP2 ⁽¹⁾ P2A ⁽¹⁾	_	_	_	MDMIN	Y	_
RC4	6	3	-	_	_	C2OUT	SRNQ	_	P1B	TX ⁽¹⁾ CK ⁽¹⁾	_	_	MDOUT	Υ	_
RC5	5	2	_	_	_	_	_	_	CCP1 P1A	RX ⁽¹⁾ DT ⁽¹⁾	_	_	MDCIN2	Y	_
RC6	8	5	AN8	_	CPS8	_	_		CCP4	_	SS1	_	_	Υ	_
RC7	9	6	AN9	_	CPS9	_	_	_	_	_	SDO1	_	_	Υ	_
VDD	1	18	_	_	_	_	_	_	_	_	_	_	_		VDD
Vss	20	17	_	_	_		_	_	_	_		_	_	_	Vss

Note 1: Pin function is selectable via the APFCON0 or APFCON1 register.

Table of Contents

1.0 Device Overview	
2.0 Enhanced Mid-range CPU	19
3.0 Memory Organization	21
4.0 Device Configuration	
5.0 Oscillator Module (With Fail-Safe Clock Monitor)	55
6.0 Reference Clock Module	73
7.0 Resets	77
8.0 Interrupts	
9.0Power-Down Mode (Sleep)	99
10.0Watchdog Timer	101
11.0Data EEPROM and Flash Program Memory Control	105
12.0 I/O Ports	119
13.0 Interrupt-On-Change	139
14.0 Fixed Voltage Reference (FVR)	
15.0 Temperature Indicator Module	
16.0 Analog-to-Digital Converter (ADC) Module	149
17.0 Digital-to-Analog Converter (DAC) Module	163
18.0 SR Latch	169
19.0 Comparator Module	175
20.0 Timer0 Module	183
21.0 Timer1 Module with Gate Control	187
22.0 Timer2/4/6 Modules	199
23.0 Data Signal Modulator	203
24.0 Capture/Compare/PWM Modules	213
25.0 Master Synchronous Serial Port (MSSP1 and MSSP2) Module	
26.0 Enhanced Universal Synchronous Asynchronous Receiver Transmitter (EUSART)	293
27.0 Capacitive Sensing (CPS) Module	321
28.0 In-Circuit Serial Programming™ (ICSP™)	329
29.0 Instruction Set Summary	333
30.0 Electrical Specifications	347
31.0 DC and AC Characteristics Graphs and Charts	
32.0 Development Support	411
33.0 Packaging Information	415
Appendix A: Data Sheet Revision History	437
Appendix B: Migrating From Other PIC® Devices	437
The Microchip Web Site	447
Customer Change Notification Service	
Customer Support	447
Reader Response	448
Product Identification System	449
Worldwide Sales and Service	450

TO OUR VALUED CUSTOMERS

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

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

Most Current Data Sheet

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

http://www.microchip.com

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

Errata

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

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

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

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

Customer Notification System

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

1.0 DEVICE OVERVIEW

The PIC16(L)F1825/1829 are described within this data sheet. They are available in 14/20 pin packages. Figure 1-1 shows a block diagram of the PIC16(L)F1825/1829 devices. Tables 1-2 and 1-3 show the pinout descriptions.

Reference Table 1-1 for peripherals available per device.

TABLE 1-1: DEVICE PERIPHERAL SUMMARY

	ı						
Peripheral	PIC16(L)F1825	PIC16(L)F1829					
ADC	•	•					
Capacitive Sensing (CP	S) Module	•	•				
Data EEPROM		•	•				
Digital-to-Analog Conve	rter (DAC)	•	•				
Digital Signal Modulator	(DSM)	•	•				
EUSART		•	•				
Fixed Voltage Reference	•	•					
SR Latch	•	•					
Capture/Compare/PWM Modules							
	ECCP1	•	•				
	ECCP2	•	•				
	CCP3	•	•				
	CCP4	•	•				
Comparators							
	C1	•	•				
	C2	•	•				
Master Synchronous Se							
	MSSP1	•	•				
	MSSP2		•				
Timers							
	Timer0	•	•				
	Timer1	•	•				
	Timer2	•	•				
	Timer4	•	•				
	Timer6	•	•				

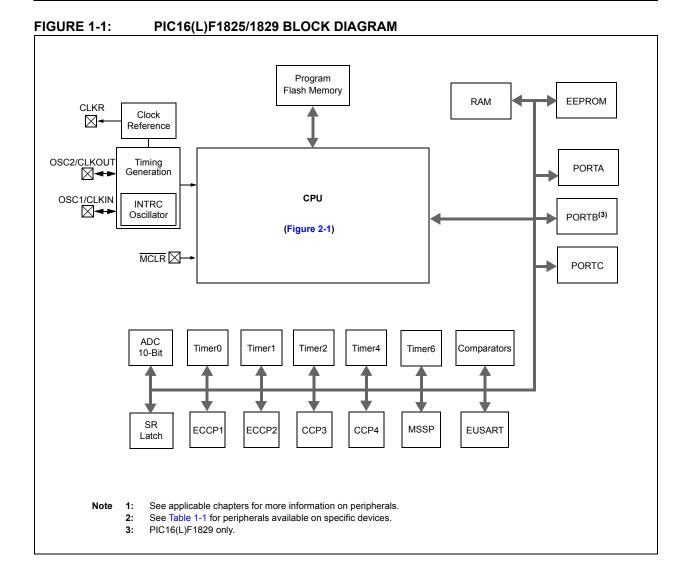


TABLE 1-2: PIC16(L)F1825 PINOUT DESCRIPTION

Name	Function	Input Type	Output Type	Description
RA0/AN0/CPS0/C1IN+/VREF-/	RA0	TTL	CMOS	General purpose I/O.
DACOUT/TX ⁽¹⁾ /CK ⁽¹⁾ /	AN0	AN	_	A/D Channel 0 input.
ICSPDAT/ICDDAT	CPS0	AN	_	Capacitive sensing input 0.
	C1IN+	AN	_	Comparator C1 positive input.
	VREF-	AN	_	A/D and DAC Negative Voltage Reference input.
	DACOUT	_	AN	Digital-to-Analog Converter output.
	TX	_	CMOS	USART asynchronous transmit.
	CK	ST	CMOS	USART synchronous clock.
	ICSPDAT	ST	CMOS	ICSP™ Data I/O.
	ICDDAT	ST	CMOS	In-Circuit Data I/O.
RA1/AN1/CPS1/C12IN0-/VREF+/	RA1	TTL	CMOS	General purpose I/O.
SRI/RX ⁽¹⁾ /DT ⁽¹⁾ /ICSPCLK/	AN1	AN	_	A/D Channel 1 input.
ICDCLK	CPS1	AN	_	Capacitive sensing input 1.
	C12IN0-	AN	_	Comparator C1 or C2 negative input.
	VREF+	AN	_	A/D and DAC Positive Voltage Reference input.
	SRI	ST	_	SR Latch input.
	RX	ST	_	USART asynchronous input.
	DT	ST	CMOS	USART synchronous data.
	ICSPCLK	ST	_	Serial Programming Clock.
	ICDCLK	ST	_	In-Circuit Debug Clock.
RA2/AN2/CPS2/T0CKI/INT/	RA2	ST	CMOS	General purpose I/O.
C1OUT/SRQ/CCP3/FLT0	AN2	AN	_	A/D Channel 2 input.
	CPS2	AN	_	Capacitive sensing input 2.
	T0CKI	ST	_	Timer0 clock input.
	INT	ST	_	External interrupt.
	C1OUT	_	CMOS	Comparator C1 output.
	SRQ	_	CMOS	SR Latch non-inverting output.
	CCP3	ST	CMOS	Capture/Compare/PWM3.
	FLT0	ST	_	ECCP Auto-Shutdown Fault input.
RA3/SS1 ⁽¹⁾ /T1G ⁽¹⁾ /VPP/MCLR	RA3	TTL	_	General purpose input.
	SS1	ST	_	Slave Select input.
	T1G	ST	_	Timer1 Gate input.
	VPP	HV	_	Programming voltage.
	MCLR	ST	_	Master Clear with internal pull-up.

Legend: AN = Analog input or output CMOS = CMOS compatible input or output OD = Open Drain

TTL = TTL compatible input ST = Schmitt Trigger input with CMOS levels I^2C^{TM} = Schmitt Trigger input with I^2C HV = High Voltage XTAL = Crystal levels

Note 1: Pin functions can be moved using the APFCON0 or APFCON1 register.

TABLE 1-2: PIC16(L)F1825 PINOUT DESCRIPTION (CONTINUED)

Name	Function	Input Type	Output Type	Description
RA4/AN3/CPS3/OSC2/	RA4	TTL	CMOS	General purpose I/O.
CLKOUT/T1OSO/CLKR/ SDO1 ⁽¹⁾ /P2B ⁽¹⁾ /T1G ^(1,2)	AN3	AN	_	A/D Channel 3 input.
SDOT "P2B" TIG	CPS3	AN	_	Capacitive sensing input 3.
	OSC2	_	CMOS	Comparator C2 output.
	CLKOUT	_	CMOS	Fosc/4 output.
	T10S0	XTAL	XTAL	Timer1 oscillator connection.
	CLKR	_	CMOS	Clock Reference output.
	SDO1	_	CMOS	SPI data output.
	P2B	_	CMOS	PWM output.
	T1G	ST	_	Timer1 Gate input.
RA5/CLKIN/OSC1/T1OSI/	RA5	TTL	CMOS	General purpose I/O.
T1CKI/P2A ⁽¹⁾ /CCP2 ⁽¹⁾	CLKIN	CMOS	_	External clock input (EC mode).
	OSC1	XTAL	_	Crystal/Resonator (LP, XT, HS modes).
	T10SI	XTAL	XTAL	Timer1 oscillator connection.
	T1CKI	ST	_	Timer1 clock input.
	P2A	_	CMOS	PWM output.
	CCP2	ST	CMOS	Capture/Compare/PWM2.
RC0/AN4/CPS4/C2IN+/SCL/	RC0	TTL	CMOS	General purpose I/O.
SCK/P1D ⁽¹⁾	AN4	AN	_	A/D Channel 4 input.
	CPS4	AN	_	Capacitive sensing input 4.
	C2IN+	AN	_	Comparator C2 positive input.
	SCL	I ² C	OD	I ² C™ clock.
	SCK	ST	CMOS	SPI clock.
	P1D	_	CMOS	PWM output.
RC1/AN5/CPS5/C12IN1-/SDA/	RC1	TTL	CMOS	General purpose I/O.
SDI/P1C ⁽¹⁾ /CCP4	AN5	AN	_	A/D Channel 5 input.
	CPS5	AN	_	Capacitive sensing input 5.
	C12IN1-	AN	_	Comparator C1 or C2 negative input.
	SDA	I ² C	OD	I ² C data input/output.
	SDI	CMOS	_	SPI data input.
	P1C	_	CMOS	PWM output.
	CCP4	AN	_	Capture/Compare/PWM4.
RC2/AN6/CPS6/C12IN2-/	RC2	TTL	CMOS	General purpose I/O.
P1D ^(1,2) /P2B ^(1,2) /SDO1 ^(1,2) /	AN6	AN	_	A/D Channel 6 input.
MDCIN1	CPS6	AN	_	Capacitive sensing input 6.
	C12IN2-	AN	_	Comparator C1 or C2 negative input.
	P1D	_	CMOS	PWM output.
	P2B	_	CMOS	PWM output.
	SDO1	_	CMOS	SPI data output.
	MDCIN1	ST	_	Modulator Carrier Input 1.

Legend: AN = Analog input or output CMOS = CMOS compatible input or output OD = Open Drain

TTL = TTL compatible input ST = Schmitt Trigger input with CMOS levels I^2C^{TM} = Schmitt Trigger input with I^2C HV = High Voltage XTAL = Crystal levels

Note 1: Pin functions can be moved using the APFCON0 or APFCON1 register.

2. Default function leasting

TABLE 1-2: PIC16(L)F1825 PINOUT DESCRIPTION (CONTINUED)

Name	Function	Input Type	Output Type	Description
RC3/AN7/CPS7/C12IN3-/	RC3	TTL	CMOS	General purpose I/O.
P2A ^(1,2) /CCP2 ^(1,2) /P1C ^(1,2) / SS1 ^(1,2) /MDMIN	AN7	AN	_	A/D Channel 7 input.
35 IV / //MIDIMIN	CPS7	AN	_	Capacitive sensing input 7.
	C12IN3-	AN	_	Comparator C1 or C2 negative input.
	P2A		CMOS	PWM output.
	CCP2	AN	_	Capture/Compare/PWM2.
	P1C		CMOS	PWM output.
	SS1	ST	_	Slave Select input.
	MDMIN	ST	_	Modulator source input.
RC4/C2OUT/SRNQ/P1B/TX ^(1,2) /	RC4	TTL	CMOS	General purpose I/O.
CK ^(1,2) /MDOUT	C2OUT		CMOS	Comparator C2 output.
	SRNQ		CMOS	SR Latch inverting output.
	P1B		CMOS	PWM output.
	TX		CMOS	USART asynchronous transmit.
	CK	ST	CMOS	USART synchronous clock.
	MDOUT		CMOS	Modulator output.
RC5/P1A/CCP1/DT ^(1,2) /RX ^(1,2) /	RC5	TTL	CMOS	General purpose I/O.
MDCIN2	P1A		CMOS	PWM output.
	CCP1	ST	CMOS	Capture/Compare/PWM1.
	RX	ST	_	USART asynchronous input.
	DT	ST	CMOS	USART synchronous data.
	MDCIN2	ST	_	Modulator Carrier Input 2.
VDD	VDD	Power	_	Positive supply.
Vss	Vss	Power	_	Ground reference.

Legend: AN = Analog input or output CMOS = CMOS compatible input or output OD = Open Drain

TTL = TTL compatible input ST = Schmitt Trigger input with CMOS levels I^2C^{TM} = Schmitt Trigger input with I^2C HV = High Voltage ST = Crystal levels

Note 1: Pin functions can be moved using the APFCON0 or APFCON1 register.

TABLE 1-3: PIC16(L)F1829 PINOUT DESCRIPTION

Name	Function	Input Type	Output Type	Description
RA0/AN0/CPS0/C1IN+/VREF-/	RA0	TTL	CMOS	General purpose I/O.
DACOUT/ICSPDAT/ICDDAT	AN0	AN	_	A/D Channel 0 input.
	CPS0	AN	_	Capacitive sensing input 0.
	C1IN+	AN	_	Comparator C1 positive input.
	VREF-	AN	_	A/D and DAC Negative Voltage Reference input.
	DACOUT	_	AN	Digital-to-Analog Converter output.
	ICSPDAT	ST	CMOS	ICSP™ Data I/O.
	ICDDAT	ST	CMOS	In-Circuit Data I/O.
RA1/AN1/CPS1/C12IN0-/VREF+/	RA1	TTL	CMOS	General purpose I/O.
SRI/ICSPCLK/ICDCLK	AN1	AN	_	A/D Channel 1 input.
	CPS1	AN	_	Capacitive sensing input 1.
	C12IN0-	AN	_	Comparator C1 or C2 negative input.
	VREF+	AN	_	A/D and DAC Positive Voltage Reference input.
	SRI	ST	_	SR Latch input.
	ICSPCLK	ST	_	Serial Programming Clock.
	ICDCLK	ST	_	In-Circuit Debug Clock.
RA2/AN2/CPS2/T0CKI/INT/	RA2	ST	CMOS	General purpose I/O.
C1OUT/SRQ/CCP3/FLT0	AN2	AN	_	A/D Channel 2 input.
	CPS2	AN	_	Capacitive sensing input 2.
	T0CKI	ST	_	Timer0 clock input.
	INT	ST	_	External interrupt.
	C10UT	_	CMOS	Comparator C1 output.
	SRQ	_	CMOS	SR Latch non-inverting output.
	CCP3	ST	CMOS	Capture/Compare/PWM3.
	FLT0	ST	_	ECCP Auto-Shutdown Fault input.
RA3/T1G ⁽¹⁾ /VPP/MCLR	RA3	TTL	_	General purpose input.
	T1G	ST	_	Timer1 Gate input.
	VPP	HV	_	Programming voltage.
	MCLR	ST	_	Master Clear with internal pull-up.
RA4/AN3/CPS3/OSC2/	RA4	TTL	CMOS	General purpose I/O.
CLKOUT/T1OSO/CLKR/SS2 ⁽¹⁾ / P2B ⁽¹⁾ /T1G ^(1,2)	AN3	AN	_	A/D Channel 3 input.
FZB\ // I IG\ / /	CPS3	AN	_	Capacitive sensing input 3.
	OSC2	_	CMOS	Comparator C2 output.
	CLKOUT		CMOS	Fosc/4 output.
	T10S0	XTAL	XTAL	Timer1 oscillator connection.
	CLKR		CMOS	Clock Reference output.
	SS2	ST		Slave Select input 2.
	P2B		CMOS	PWM output.
	T1G	ST	_	Timer1 Gate input.

Legend: AN = Analog input or output CMOS = CMOS compatible input or output OD = Open Drain

TTL = TTL compatible input ST = Schmitt Trigger input with CMOS levels I^2C^{TM} = Schmitt Trigger input with I^2C HV = High Voltage XTAL = Crystal levels

Note 1: Pin functions can be moved using the APFCON0 or APFCON1 register.

TABLE 1-3: PIC16(L)F1829 PINOUT DESCRIPTION (CONTINUED)

Name	Function	Input Type	Output Type	Description
RA5/CLKIN/OSC1/T1OSI/	RA5	TTL	CMOS	General purpose I/O.
SD02 ⁽¹⁾ /T1CKI/P2A ⁽¹⁾ /CCP2 ⁽¹⁾	CLKIN	CMOS		External clock input (EC mode).
	OSC1	XTAL	_	Crystal/Resonator (LP, XT, HS modes).
	T10SI	XTAL	XTAL	Timer1 oscillator connection.
	SD02	_	CMOS	SPI data output 2.
	T1CKI	ST	_	Timer1 clock input.
	P2A	_	CMOS	PWM output.
	CCP2	ST	CMOS	Capture/Compare/PWM2.
RB4/AN10/CPS10/SDA1/SDI1	RB4	TTL	CMOS	General purpose I/O.
	AN10	AN	_	A/D Channel 10 input.
	CPS10	AN	_	Capacitive sensing input 10.
	SDA1	I ² C	OD	I ² C™ data input/output.
	SDI1	CMOS	_	SPI data input.
RB5/AN11/CPS11/RX ^(1,2) /	RB5	TTL	CMOS	General purpose I/O.
DT ^(1,2) /SDA2/SDI2	AN11	AN	_	A/D Channel 11 input.
	CPS11	AN	_	Capacitive sensing input 11.
	RX	ST	_	USART asynchronous input.
	DT	ST	CMOS	USART synchronous data.
	SDA2	I ² C	OD	I ² C data input/output 2.
	SDI2	CMOS	_	SPI data input 2.
RB6/SCL1/SCK1	RB6	TTL	CMOS	General purpose I/O.
	SCL1	I ² C	OD	I ² C™ clock 1.
	SCK1	ST	CMOS	SPI clock 1.
RB7/TX ^(1,2) /CK ^(1,2) /SCL2/SCK2	RB7	TTL	CMOS	General purpose I/O.
	TX		CMOS	USART asynchronous transmit.
	CK	ST	CMOS	USART synchronous clock.
	SCL2	I ² C	OD	I ² C™ clock 2.
	SCK2	ST	CMOS	SPI clock 2.
RC0/AN4/CPS4/C2IN+/P1D(1)/	RC0	TTL	CMOS	General purpose I/O.
SS2 ^(1,2)	AN4	AN		A/D Channel 4 input.
	CPS4	AN	1	Capacitive sensing input 4.
	C2IN+	AN		Comparator C2 positive input.
	P1D	_	CMOS	PWM output.
	SS2	ST		Slave Select input 2.
RC1/AN5/CPS5/C12IN1-/P1C(1)/	RC1	TTL	CMOS	General purpose I/O.
SD02 ^(1,2)	AN5	AN	_	A/D Channel 5 input.
	CPS5	AN	_	Capacitive sensing input 5.
	C12IN1-	AN	_	Comparator C1 or C2 negative input.
	P1C		CMOS	PWM output.
	SDO2	_	CMOS	SPI data output 2.

Legend: AN = Analog input or output CMOS = CMOS compatible input or output OD = Open Drain

TTL = TTL compatible input ST = Schmitt Trigger input with CMOS levels I^2C^{TM} = Schmitt Trigger input with I^2C HV = High Voltage XTAL = Crystal levels

Note 1: Pin functions can be moved using the APFCON0 or APFCON1 register.

TABLE 1-3: PIC16(L)F1829 PINOUT DESCRIPTION (CONTINUED)

Name	Function	Input Type	Output Type	Description
RC2/AN6/CPS6/C12IN2-/	RC2	TTL	CMOS	General purpose I/O.
P1D ^(1,2) /P2B ^(1,2) /MDCIN1	AN6	AN	_	A/D Channel 6 input.
	CPS6	AN	_	Capacitive sensing input 6.
	C12IN2-	AN	_	Comparator C1 or C2 negative input.
	P1D	_	CMOS	PWM output.
	P2B	_	CMOS	PWM output.
	MDCIN1	ST	_	Modulator Carrier Input 1.
RC3/AN7/CPS7/C12IN3-/	RC3	TTL	CMOS	General purpose I/O.
P2A ^(1,2) /CCP2 ^(1,2) /P1C ^(1,2) /	AN7	AN	_	A/D Channel 7 input.
MDMIN	CPS7	AN	_	Capacitive sensing input 7.
	C12IN3-	AN	_	Comparator C1 or C2 negative input.
	P2A	_	CMOS	PWM output.
	CCP2	AN	_	Capture/Compare/PWM2.
	P1C	_	CMOS	PWM output.
	MDMIN	ST	_	Modulator source input.
RC4/C2OUT/SRNQ/P1B/TX ⁽¹⁾ /	RC4	TTL	CMOS	General purpose I/O.
CK ⁽¹⁾ /MDOUT	C2OUT	_	CMOS	Comparator C2 output.
	SRNQ	_	CMOS	SR Latch inverting output.
	P1B	_	CMOS	PWM output.
	TX	_	CMOS	USART asynchronous transmit.
	CK	ST	CMOS	USART synchronous clock.
	MDOUT	_	CMOS	Modulator output.
RC5/P1A/CCP1/DT ⁽¹⁾ /RX ⁽¹⁾ /	RC5	TTL	CMOS	General purpose I/O.
MDCIN2	P1A	_	CMOS	PWM output.
	CCP1	ST	CMOS	Capture/Compare/PWM1.
	RX	ST	_	USART asynchronous input.
	DT	ST	CMOS	USART synchronous data.
	MDCIN2	ST	_	Modulator Carrier Input 2.
RC6/AN8/CPS8/CCP4/SS1	RC6	TTL	CMOS	General purpose I/O.
	AN8	AN	_	A/D Channel 8 input.
	CPS8	AN	_	Capacitive sensing input 8.
	CCP4	AN	_	Capture/Compare/PWM4.
	SS1	ST	_	Slave Select input.
RC7/AN9/CPS9/SDO1	RC7	TTL	CMOS	General purpose I/O.
	AN9	AN	_	A/D Channel 9 input.
	CPS9	AN	_	Capacitive sensing input 9.
	SDO1	_	CMOS	SPI data output.
VDD	VDD	Power	_	Positive supply.
Vss	Vss	Power	_	Ground reference.

Legend: AN = Analog input or output CMOS = CMOS compatible input or output OD = Open Drain

TTL = TTL compatible input ST = Schmitt Trigger input with CMOS levels I^2C^{TM} = Schmitt Trigger input with I^2C HV = High Voltage XTAL = Crystal levels

Note 1: Pin functions can be moved using the APFCON0 or APFCON1 register.

2.0 ENHANCED MID-RANGE CPU

This family of devices contain an enhanced mid-range 8-bit CPU core. The CPU has 49 instructions. Interrupt capability includes automatic context saving. The hardware stack is 16 levels deep and has Overflow and Underflow Reset capability. Direct, Indirect, and Relative Addressing modes are available. Two File Select Registers (FSRs) provide the ability to read program and data memory.

- · Automatic Interrupt Context Saving
- · 16-level Stack with Overflow and Underflow
- · File Select Registers
- · Instruction Set

2.1 Automatic Interrupt Context Saving

During interrupts, certain registers are automatically saved in shadow registers and restored when returning from the interrupt. This saves stack space and user code. See Section 8.5 "Automatic Context Saving", for more information.

2.2 16-level Stack with Overflow and Underflow

These devices have an external stack memory 15 bits wide and 16 words deep. A Stack Overflow or Underflow will set the appropriate bit (STKOVF or STKUNF) in the PCON register, and if enabled will cause a software Reset. See section **Section 3.4** "Stack" for more details.

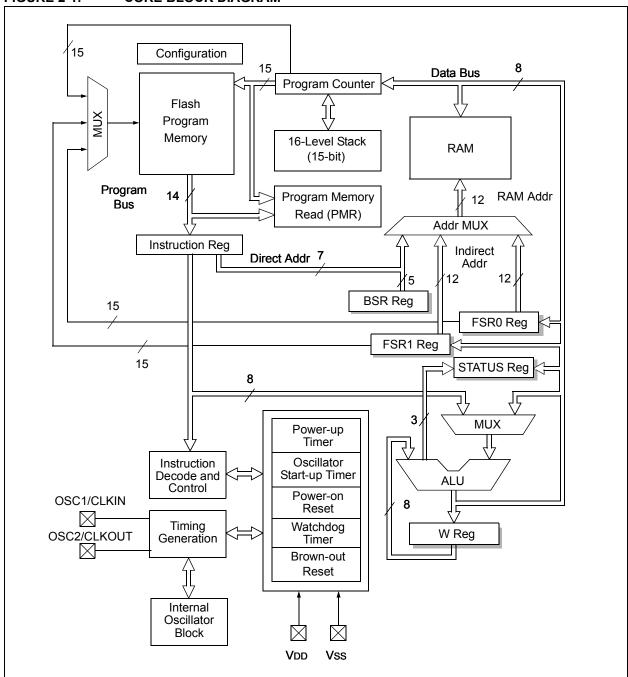
2.3 File Select Registers

There are two 16-bit File Select Registers (FSR). FSRs can access all file registers and program memory, which allows one Data Pointer for all memory. When an FSR points to program memory, there is one additional instruction cycle in instructions using INDF to allow the data to be fetched. General purpose memory can now also be addressed linearly, providing the ability to access contiguous data larger than 80 bytes. There are also new instructions to support the FSRs. See Section 3.5 "Indirect Addressing" for more details.

2.4 Instruction Set

There are 49 instructions for the enhanced mid-range CPU to support the features of the CPU. See Section 29.0 "Instruction Set Summary" for more details.

FIGURE 2-1: CORE BLOCK DIAGRAM



3.0 MEMORY ORGANIZATION

These devices contain the following types of memory:

- · Program Memory
 - Configuration Words
 - Device ID
 - User ID
 - Flash Program Memory
- · Data Memory
 - Core Registers
 - Special Function Registers
 - General Purpose RAM
 - Common RAM
- Data EEPROM memory⁽¹⁾

Note 1: The Data EEPROM Memory and the method to access Flash memory through the EECON registers is described in Section 11.0 "Data EEPROM and Flash Program Memory Control".

The following features are associated with access and control of program memory and data memory:

- · PCL and PCLATH
- Stack
- · Indirect Addressing

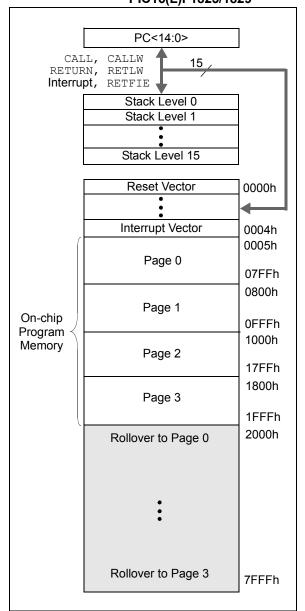
3.1 Program Memory Organization

The enhanced mid-range core has a 15-bit program counter capable of addressing 32K x 14 program memory space. Table 3-1 shows the memory sizes implemented for the PIC16(L)F1825/1829 family. Accessing a location above these boundaries will cause a wrap-around within the implemented memory space. The Reset vector is at 0000h and the interrupt vector is at 0004h (See Figure 3-1).

TABLE 3-1: DEVICE SIZES AND ADDRESSES

Device	Program Memory Space (Words)	Last Program Memory Address			
PIC16(L)F1825 PIC16(L)F1829	8,192	7FFFh			

FIGURE 3-1: PROGRAM MEMORY MAP AND STACK FOR PIC16(L)F1825/1829



3.1.1 READING PROGRAM MEMORY AS DATA

There are two methods of accessing constants in program memory. The first method is to use tables of RETLW instructions. The second method is to set an FSR to point to the program memory.

3.1.1.1 RETLW Instruction

The RETLW instruction can be used to provide access to tables of constants. The recommended way to create such a table is shown in Example 3-1.

EXAMPLE 3-1: RETLW INSTRUCTION

```
constants
   BRW
                      ;Add Index in W to
                      ;program counter to
                      ;select data
                      ;Index0 data
   RETLW DATA0
   RETLW DATA1
                      ;Index1 data
   RETLW DATA2
   RETLW DATA3
my_function
   ;... LOTS OF CODE...
   MOVLW DATA INDEX
   call constants
   ; ... THE CONSTANT IS IN W
```

The BRW instruction makes this type of table very simple to implement. If your code must remain portable with previous generations of microcontrollers, then the BRW instruction is not available so the older table read method must be used.

3.1.1.2 Indirect Read with FSR

The program memory can be accessed as data by setting bit 7 of the FSRxH register and reading the matching INDFx register. The MOVIW instruction will place the lower eight bits of the addressed word in the W register. Writes to the program memory cannot be performed via the INDF registers. Instructions that access the program memory via the FSR require one extra instruction cycle to complete. Example 3-2 demonstrates accessing the program memory via an FSR.

The High directive will set bit<7> if a label points to a location in program memory.

EXAMPLE 3-2: ACCESSING PROGRAM MEMORY VIA FSR

```
constants
                      ;Index0 data
   RETLW DATAO
   RETLW DATA1
                      ;Index1 data
   RETLW DATA2
   RETLW DATA3
my_function
   ;... LOTS OF CODE...
   MOVLW LOW constants
   MOVWF FSR1L
   MOVLW HIGH constants
   MOVWF
          FSR1H
   MOVIW
          0[FSR1]
;THE PROGRAM MEMORY IS IN W
```

3.2 Data Memory Organization

The data memory is partitioned in 32 memory banks with 128 bytes in a bank. Each bank consists of (Figure 3-2):

- · 12 core registers
- · 20 Special Function Registers (SFR)
- Up to 80 bytes of General Purpose RAM (GPR)
- · 16 bytes of common RAM

The active bank is selected by writing the bank number into the Bank Select Register (BSR). Unimplemented memory will read as '0'. All data memory can be accessed either directly (via instructions that use the file registers) or indirectly via the two File Select Registers (FSR). See Section 3.5 "Indirect Addressing" for more information.

Data Memory uses a 12-bit address. The upper 7-bits of the address define the Bank address and the lower 5-bits select the registers/RAM in that bank.

3.2.1 CORE REGISTERS

The core registers contain the registers that directly affect the basic operation of the PIC16(L)F1825/1829. These registers are listed below:

- INDF0
- INDF1
- PCL
- STATUS
- FSR0 Low
- · FSR0 High
- FSR1 Low
- FSR1 High
- BSR
- WREG
- PCLATH
- INTCON

Note: The core registers are the first 12 addresses of every data memory bank.

3.2.1.1 STATUS Register

The STATUS register, shown in Register 3-1, contains:

- · the arithmetic status of the ALU
- · the Reset status

The STATUS register can be the destination for any instruction, like any other register. If the STATUS register is the destination for an instruction that affects the Z, DC or C bits, then the write to these three bits is disabled. These bits are set or cleared according to the device logic. Furthermore, the $\overline{\text{TO}}$ and $\overline{\text{PD}}$ bits are not writable. Therefore, the result of an instruction with the STATUS register as destination may be different than intended.

For example, CLRF STATUS will clear the upper three bits and set the Z bit. This leaves the STATUS register as '000u u1uu' (where u = unchanged).

It is recommended, therefore, that only BCF, BSF, SWAPF and MOVWF instructions are used to alter the STATUS register, because these instructions do not affect any Status bits. For other instructions not affecting any Status bits (Refer to Section 29.0 "Instruction Set Summary").

Note 1: The <u>C</u> and <u>DC</u> bits operate as Borrow and <u>Digit Borrow</u> out bits, respectively, in subtraction

REGISTER 3-1: STATUS: STATUS REGISTER

U-0	U-0	U-0	R-1/q	R-1/q	R/W-0/u	R/W-0/u	R/W-0/u
_	_	_	TO	PD	Z	DC ⁽¹⁾	C ⁽¹⁾
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	q = Value depends on condition

bit 7-5	Unimplemented: Read as	(\cap)
DII 7-5	Unimplemented: Read as	()

bit 4 **TO:** Time-out bit

1 = After power-up, CLRWDT instruction or SLEEP instruction

0 = A WDT time-out occurred

bit 3 **PD:** Power-down bit

1 = After power-up or by the ${\tt CLRWDT}$ instruction

0 = By execution of the SLEEP instruction

bit 2 **Z**: Zero bit

1 = The result of an arithmetic or logic operation is zero

0 = The result of an arithmetic or logic operation is not zero

bit 1 DC: Digit Carry/Digit Borrow bit (ADDWF, ADDLW, SUBLW, SUBWF instructions)(1)

1 = A carry-out from the 4th low-order bit of the result occurred

0 = No carry-out from the 4th low-order bit of the result

bit 0 C: Carry/Borrow bit⁽¹⁾ (ADDWF, ADDLW, SUBLW, SUBWF instructions)⁽¹⁾

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

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

Note 1: For Borrow, the polarity is reversed. A subtraction is executed by adding the two's complement of the second operand. For rotate (RRF, RLF) instructions, this bit is loaded with either the high-order or low-order bit of the source register.

3.2.2 SPECIAL FUNCTION REGISTER

The Special Function Registers (FSR) are registers used by the application to control the desired operation of peripheral functions in the device. The Special Function Registers occupy the 20 bytes after the core registers of every data memory bank (addresses x0Ch/x8Ch through x1Fh/x9Fh). The registers associated with the operation of the peripherals are described in the appropriate peripheral chapter of this data sheet.

3.2.3 GENERAL PURPOSE RAM

There are up to 80 bytes of GPR in each data memory bank. The Special Function Registers occupy the 20 bytes after the core registers of every data memory bank (addresses x0Ch/x8Ch through x1Fh/x9Fh).

3.2.3.1 Linear Access to GPR

The general purpose RAM can be accessed in a non-banked method via the FSRs. This can simplify access to large memory structures. See **Section 3.5.2** "Linear Data Memory" for more information.

3.2.4 COMMON RAM

There are 16 bytes of common RAM accessible from all banks.

FIGURE 3-2: BANKED MEMORY PARTITIONING

7-bit Bank Offset	Memory Region
00h 0Bh	Core Registers (12 bytes)
0Ch	Special Function Registers (20 bytes maximum)
1Fh 20h	
	General Purpose RAM (80 bytes maximum)
6Fh 70h	Common RAM
7Fh	(16 bytes)

3.2.5 DEVICE MEMORY MAPS

The memory maps for the device family are as shown in Table 3-2.

TABLE 3-2: MEMORY MAP TABLES

Device	Banks	Table No.		
PIC16(L)F1825	0-7	Table 3-3		
PIC16(L)F1829	8-15	Table 3-4		
	16-23	Table 3-5		
	24-31	Table 3-6		
	31	Table 3-7		

DS41440C-page 26

TABL	.E 3-3: PI	C16(L)F1825/182	9 MEI	MORY MAP.	BAN	KS 0-7								
	BANK 0	`	BANK 1		BANK 2		BANK 3		BANK 4		BANK 5		BANK 6		BANK 7
000h	INDF0	080h	INDF0	100h	INDF0	180h	INDF0	200h	INDF0	280h	INDF0	300h	INDF0	380h	INDF0
001h	INDF1	081h	INDF1	101h	INDF1] 181h	INDF1	201h	INDF1	281h	INDF1	301h	INDF1	381h	INDF1
002h	PCL	082h	PCL	102h	PCL	182h	PCL	202h	PCL	282h	PCL	302h	PCL	382h	PCL
003h	STATUS	083h	STATUS	103h	STATUS	183h	STATUS	203h	STATUS	283h	STATUS	303h	STATUS	383h	STATUS
004h	FSR0L	084h	FSR0L	104h	FSR0L	184h	FSR0L	204h	FSR0L	284h	FSR0L	304h	FSR0L	384h	FSR0L
005h	FSR0H	085h	FSR0H	105h	FSR0H	185h	FSR0H	205h	FSR0H	285h	FSR0H	305h	FSR0H	385h	FSR0H
006h	FSR1L	086h	FSR1L	106h	FSR1L	186h	FSR1L	206h	FSR1L	286h	FSR1L	306h	FSR1L	386h	FSR1L
007h	FSR1H	087h	FSR1H	107h	FSR1H	187h	FSR1H	207h	FSR1H	287h	FSR1H	307h	FSR1H	387h	FSR1H
008h	BSR	088h	BSR	108h	BSR	188h	BSR	208h	BSR	288h	BSR	308h	BSR	388h	BSR
009h	WREG PCLATH	089h 08Ah	WREG PCLATH	109h	WREG PCLATH	189h 18Ah	WREG PCLATH	209h 20Ah	WREG	289h 28Ah	WREG PCLATH	309h 30Ah	WREG PCLATH	389h	WREG
00Ah 00Bh	INTCON	08Bh	INTCON	10Ah 10Bh	INTCON	18An 18Bh	INTCON	20An 20Bh	PCLATH INTCON	28An 28Bh	INTCON	30An 30Bh	INTCON	38Ah 38Bh	PCLATH INTCON
00Bn	PORTA	08Ch	TRISA	10Gh	LATA	18Ch	ANSELA	20Bii	WPUA	28Ch	—	30Ch	INTCON	38Ch	INLVLA
00Dh	PORTB ⁽¹⁾	08Dh	TRISB ⁽¹⁾	10Dh	LATA LATB ⁽¹⁾	18Dh	ANSELB ⁽¹⁾	20Dh	WPUB ⁽¹⁾	28Dh		30Dh	_	38Dh	INLVLA
00Eh	PORTC	08Eh	TRISC	10Eh	LATC	18Eh	ANSELEC	20Eh	WPUC	28Eh		30Eh		38Eh	INLVLB
00E11	–	08Fh	- IRISC	10EII	— LATC	18Fh	ANSELC —	20EII		28Fh		30Fh		38Fh	
010h		090h		110h		190h		210h	<u>_</u>	290h		310h		390h	_
011h	PIR1	091h	PIE1	111h	CM1CON0	191h	EEADRL	211h	SSP1BUF	291h	CCPR1L	311h	CCPR3L	391h	IOCAP
011h	PIR2	09111 092h	PIE2	112h	CM1CON1	192h	EEADRH	212h	SSP1ADD	292h	CCPR1H	312h	CCPR3H	392h	IOCAN
			PIE2 PIE3					-	SSP1MSK	•			CCP3CON		IOCAN
013h	PIR3	093h		113h	CM2CON0	193h	EEDATL	213h		293h	CCP1CON	313h	CCP3CON	393h	
014h	PIR4 ⁽¹⁾	094h	PIE4 ⁽¹⁾	114h	CM2CON1	194h	EEDATH	214h	SSP1STAT	294h	PWM1CON	314h		394h	IOCBP ⁽¹⁾
015h	TMR0	095h	OPTION_REG	115h	CMOUT	195h	EECON1	215h	SSP1CON1	295h	CCP1AS	315h		395h	IOCBN ⁽¹⁾
016h	TMR1L	096h	PCON	116h	BORCON	196h	EECON2	216h	SSP1CON2	296h	PSTR1CON	316h		396h	IOCBF ⁽¹⁾
017h	TMR1H	097h	WDTCON	117h	FVRCON	197h		217h	SSP1CON3	297h	_	317h	_	397h	_
018h	T1CON	098h	OSCTUNE	118h	DACCON0	198h		218h		298h	CCPR2L	318h	CCPR4L	398h	_
019h	T1GCON	099h	OSCCON	119h	DACCON1	199h	RCREG	219h	SSP2BUF ⁽¹⁾	299h	CCPR2H	319h	CCPR4H	399h	_
01Ah	TMR2	09Ah	OSCSTAT	11Ah	SRCON0	19Ah	TXREG	21Ah	SSP2ADD ⁽¹⁾	29Ah	CCP2CON	31Ah	CCP4CON	39Ah	CLKRCON
01Bh	PR2	09Bh	ADRESL	11Bh	SRCON1	19Bh	SPBRGL	21Bh	SSP2MSK ⁽¹⁾	29Bh	PWM2CON	31Bh		39Bh	_
01Ch	T2CON	09Ch	ADRESH	11Ch		19Ch	SPBRGH	21Ch	SSP2STAT ⁽¹⁾	29Ch	CCP2AS	31Ch		39Ch	MDCON
01Dh	_	09Dh	ADCON0	11Dh	APFCON0	19Dh	RCSTA	21Dh	SSP2CON1 ⁽¹⁾	29Dh	PSTR2CON	31Dh		39Dh	MDSRC
01Eh	CPSCON0	09Eh	ADCON1	11Eh	APFCON1	19Eh	TXSTA	21Eh	SSP2CON2 ⁽¹⁾	29Eh	CCPTMRS	31Eh	_	39Eh	MDCARL
01Fh	CPSCON1	09Fh	_	11Fh	_	19Fh	BAUDCON	21Fh	SSP2CON3 ⁽¹⁾	29Fh	_	31Fh	_	39Fh	MDCARH
020h		0A0h		120h		1A0h		220h		2A0h		320h		3A0h	
	General		General		General		General		General		General		General		General
	Purpose		Purpose		Purpose		Purpose		Purpose		Purpose		Purpose		Purpose
	Register		Register		Register		Register		Register		Register		Register		Register
	96 Bytes		80 Bytes		80 Bytes		80 Bytes		80 Bytes		80 Bytes		80 Bytes		80 Bytes
06Fh		0EFh		16Fh		1EFh		26Fh		2EFh		36Fh		3EFh	
070h		0F0h		170h		1F0h		270h		2F0h		370h		3F0h	
	0		Accesses		Accesses		Accesses		Accesses		Accesses		Accesses		Accesses
	Common RAM		70h – 7Fh		70h – 7Fh		70h – 7Fh		70h – 7Fh		70h – 7Fh		70h – 7Fh		70h – 7Fh
07Fh		0FFh		17Fh		1FFh		27Fh		2FFh		37Fh		3FFh	

Legend: = Unimplemented data memory locations, read as '0'

Note 1: Available only on PIC16(L)F1829.

Ğ
Ś
4
\rightarrow
4
4
\circ
ŏ
ဂို
ဂ္ဂ
)C-pa
)C-pag
þa
þa

1ABLE 3-4. FIG 10(L)F 1023/1023 WIEWORT WAP, DANKS 0-13	TABLE 3-4:	PIC16(L)F1825/1829 MEMORY MAP, BANKS 8-1	5
---	------------	--	---

Apple MOPF Apple MOPF Sob MOPF		BANK 8		BANK 9		BANK 10		BANK 11		BANK 12		BANK 13		BANK 14		BANK 15
A02h FCL A82h PCL 502h FCL 502h FCL 602h FCL 682h FCL 702h FCL 782h FCL 782	400h	INDF0	480h	INDF0	500h	INDF0	580h	INDF0	600h	INDF0	680h	INDF0	700h	INDF0	780h	INDF0
409h FSROL 489h FSROL 509h FSROL	401h	INDF1	481h	INDF1	501h	INDF1	581h	INDF1	601h		681h	INDF1	701h		781h	INDF1
Adob	402h		482h		502h		582h		602h		682h		702h		782h	
496h	403h	STATUS	483h	STATUS	503h	STATUS	583h	STATUS		STATUS	683h	STATUS	703h	STATUS	783h	STATUS
406h FSR1L 488h FSR1L 506h FSR1L 507h	404h		484h		504h				604h				704h		784h	
497h					505h		585h		605h				705h			
A98h BSR 488h BSR 598h WREG 599h WREG 599h WREG 599h WREG 699h WREG 799h WREG 79	406h		486h		506h		586h		606h		686h		706h		786h	
499	407h								607h							
40Ab					508h		588h		608h				708h			
ABB INTCON ABB INTCON ABB INTCON SOBh INTCON SOBh INTCON SOBh CON C																
48Ch	40Ah		48Ah	PCLATH			58Ah		60Ah						78Ah	
40Dh — 48Dh — 50Dh — 6DDh — 68Dh — 70Dh — 78Dh — 40Eh — 50Eh — 58Bh — 60Dh — 68Eh — 70Dh — 78Bh — 40Fh — 49Ph — 50Ph — 69Ph — 68Fh — 70Eh — 78Bh — 410h — 499h — 610h — 590h — 611h — 699h — 710h — 799h — 412h — 492h — 612h — 592h — 612h — 692h — 712h — 792h — 413h — 493h — 513h — 593h — 713h — 793h — 416h PR4 496h — 516h — 596h <td< td=""><td></td><td>INTCON</td><td></td><td>INTCON</td><td></td><td>INTCON</td><td></td><td>INTCON</td><td></td><td>INTCON</td><td></td><td>INTCON</td><td></td><td>INTCON</td><td></td><td>INTCON</td></td<>		INTCON		INTCON		INTCON		INTCON		INTCON		INTCON		INTCON		INTCON
48Eh	40Ch	_	48Ch	_	50Ch	_	58Ch	_	60Ch	_	68Ch	_	70Ch	_	78Ch	_
49Fh		_		_		_		_		_		_		_		_
490h	40Eh		48Eh		_		58Eh				68Eh	_	70Eh	_	- 1	_
41th	40Fh	_		_	50Fh	_	58Fh	_		_		_	70Fh	_	78Fh	_
412h	410h				510h				610h		690h	_		_	790h	_
413h	411h	_		_	511h	_		_	611h	_	691h	_		_	-	_
414h	412h	_		_		_		_	612h	_		_		_	792h	_
A15h	413h	_		_	513h	_	593h	_		_	693h	_		_	793h	_
A16h	414h	_	494h	_	514h	_	594h	_	614h	_	694h	_	714h	_	794h	_
417h T4CON 497h — 517h — 597h — 617h — 697h — 717h — 797h — 418h — 498h — 518h — 598h — 618h — 698h — 718h — 799h — 410h — 498h — 514h — 598h — 619h — 699h — 719h — 799h — 41Bh — 498h — 518h — 598h — 618h — 699h — 719h — 799h — 41Ch TMR6 490h — 510h — 590h — 610h — 698h — 716h — 790h — 41Ch TMR6 490h — 510h — 590h — 610h — 690h — 710h —	415h			_	515h	_		_	615h	_	695h	_	715h	_	795h	_
418h	416h	PR4	496h	_	516h	_	596h	_	616h	_	696h	_	716h	_	796h	_
419h — 499h — 519h — 599h — 619h — 699h — 719h — 799h — 41Bh — 49Bh — 514h — 59Ah — 61Ah — 69Ah — 714h — 799h — 41Bh — 49Bh — 51Bh — 59Bh — 61Bh — 69Bh — 71Bh — 79Ah — 41Dh PR6 49Dh — 51Dh — 59Dh — 61Dh — 69Dh — 71Bh — 79Bh — 41Dh PR6 49Dh — 51Dh — 59Dh — 61Dh — 69Dh — 71Dh — 79Dh — 41Fh — 49Fh — 51Fh — 59Fh — 61Fh — 69Fh — 71Fh — <	417h	T4CON		_		_	597h	_	617h	_	697h	_	717h	_	797h	_
41Ah — 49Ah — 51Ah — 59Ah — 61Ah — 69Ah — 71Ah — 79Ah — 41Bh — 49Bh — 51Bh — 59Bh — 61Bh — 69Bh — 71Bh — 79Bh — 41Dh TRR6 49Ch — 51Ch — 59Ch — 61Ch — 69Ch — 71Ch — 79Ch — 41Dh PR6 49Dh — 51Eh — 59Ph — 61Ch — 69Ch — 71Ch — 79Ch — 41Fh — 49Fh — 51Eh — 59Fh — 61Eh — 69Fh — 71Eh — 79Fh — 42Ch General Purpose Register Register 80 Bytes Register 80 Bytes 80 Bytes 56Fh 56Fh 56Fh 57Oh Accesses 70h — 7Fh <td< td=""><td>418h</td><td>_</td><td></td><td>_</td><td>518h</td><td>_</td><td></td><td>_</td><td>618h</td><td>_</td><td></td><td>_</td><td></td><td>_</td><td></td><td>_</td></td<>	418h	_		_	518h	_		_	618h	_		_		_		_
41Bh — 49Bh — 51Bh — 59Bh — 61Bh — 69Bh — 71Bh — 79Bh — 41Dh PR6 49Dh — 51Dh — 59Dh — 61Dh — 69Dh — 71Dh — 79Dh — 41Eh T6CON 49Eh — 51Eh — 59Fh — 61Eh — 69Fh — 71Dh — 79Dh — 41Fh — 49Fh — 51Fh — 59Fh — 61Fh — 69Fh — 71Eh — 79Fh — 420h General Purpose Register 80 Bytes Register 80 Bytes Register 80 Bytes 80 Bytes 80 Bytes 65Ph 55Ph — 64Fh 64Fh 48 Bytes Unimplemented Read as '0' White Read as '0' White Read as '0' White Read as '0' 65Ph 46Fh 4Fh 4Foh 4Foh 4Foh 4Foh	419h	_	499h	_	519h	_	599h	_	619h	_	699h	_	719h	_	799h	_
41Ch TMR6 49Ch — 51Ch — 59Ch — 61Ch — 69Ch — 71Ch — 79Ch — 41Dh PR6 49Dh — 51Dh — 59Dh — 61Dh — 69Dh — 71Ch — 79Ch — 41Eh T6CON 49Eh — 51Eh — 59Eh — 61Eh — 69Ch — 71Dh — 79Dh — 41Fh — 49Fh — 51Fh — 59Fh — 61Fh — 69Eh — 71Eh — 79Eh — 420h General Purpose Register 80 Bytes Purpose Register 80 Bytes Register 80 Bytes 64Fh 64Fh 65Oh Unimplemented Read as '0' Read as '0' Windows Windows Read as '0' Windows Windows Read as '0' Windows Accesses 70h - 7Fh Accesses 70h - 7Fh Accesses 70h - 7Fh	41Ah	_		_		_		_		_		_		_		_
Accesses T0h - 7Fh T0h - 7Fh Accesses	41Bh			_		_		_	61Bh	_	69Bh	_	71Bh	_	79Bh	_
A1Eh	41Ch		49Ch	_	51Ch	_	59Ch	_	61Ch	_	69Ch	_	71Ch	_	79Ch	_
41Fh — 49Fh — 51Fh — 59Fh — 620h General Purpose Register 80 Bytes 46Fh 470h Accesses 70h – 7Fh Accesses 70h – 7Fh 49Fh — 51Fh — 59Fh — 61Fh — 620h General Purpose Register 80 Bytes 41Fh Accesses 70h – 7Fh	41Dh		49Dh	_	51Dh	_		_	61Dh	_		_	71Dh	_	79Dh	_
420h General Purpose Register 80 Bytes Accesses 70h - 7Fh Accesses 70h - 7Fh 4A0h General Purpose Register 80 Bytes Accesses 70h - 7Fh Accesses 70h - 7Fh 520h General Purpose Register 80 Bytes Accesses 70h - 7Fh		T6CON										_		_	-	_
General Purpose Register 80 Bytes Purpose Register 870h – 7Fh Accesses 70h – 7Fh Accesses		_	49Fh	_	51Fh	_	59Fh	_		_	69Fh	_	71Fh	_	79Fh	_
Purpose Register 80 Bytes Accesses 70h - 7Fh Accesses	420h		4A0h		520h		5A0h		620h	General Purpose	6A0h		720h		7A0h	
Purpose Register 80 Bytes		General		General		General		General								
Register 80 Bytes									64Fh	48 Bytes		Unimplemented		Unimplemented		Unimplemented
46Fh 470h Accesses 70h - 7Fh 70h - 7Fh 70h 70h 77h 70h 70h 77h 70h 70h 77h 70h 70									650h							
46Fh									00011							
470h 4F0h 570h 5F0h 6F0h 6F0h 770h 7F0h 7F0h 7F0h Accesses Accesses<	46Eb	·	4EEb	·	56Eb	•	5EEb	•	COE	Read as '0'	6EEh		76Eb		7EEh	
Accesses 70h - 7Fh	-				1						-					
70h - 7Fh	4/UN		4FUN		5/UN		5FUN		670h		6FUh		//Uh		/FUN	
47Fh 4FFh 57Fh 5FFh 67Fh 6FFh 77Fh 77Fh		70h – 7Fh		70h – 7Fh		70h – 7Fh		70h – 7Fh		70h – 7Fh						
	47Fh		4FFh		57Fh		5FFh		67Fh		6FFh		77Fh		7FFh	

= Unimplemented data memory locations, read as '0' Legend:

DS41440C-page 28

TABLE 3-5:	PIC16(L)F1825/1829 MEMORY MAP, BANKS 16-2	3

	BANK 16		BANK 17		BANK 18		BANK 19		BANK 20		BANK 21		BANK 22		BANK 23
800h	INDF0	880h	INDF0	900h	INDF0	980h	INDF0	A00h	INDF0	A80h	INDF0	B00h	INDF0	B80h	INDF0
801h	INDF1	881h	INDF1	901h	INDF1	981h	INDF1	A01h	INDF1	A81h	INDF1	B01h	INDF1	B81h	INDF1
802h	PCL	882h	PCL	902h	PCL	982h	PCL	A02h	PCL	A82h	PCL	B02h	PCL	B82h	PCL
803h	STATUS	883h	STATUS	903h	STATUS	983h	STATUS	A03h	STATUS	A83h	STATUS	B03h	STATUS	B83h	STATUS
804h	FSR0L	884h	FSR0L	904h	FSR0L	984h	FSR0L	A04h	FSR0L	A84h	FSR0L	B04h	FSR0L	B84h	FSR0L
805h	FSR0H	885h	FSR0H	905h	FSR0H	985h	FSR0H	A05h	FSR0H	A85h	FSR0H	B05h	FSR0H	B85h	FSR0H
806h	FSR1L	886h	FSR1L	906h	FSR1L	986h	FSR1L	A06h	FSR1L	A86h	FSR1L	B06h	FSR1L	B86h	FSR1L
807h	FSR1H	887h	FSR1H	907h	FSR1H	987h	FSR1H	A07h	FSR1H	A87h	FSR1H	B07h	FSR1H	B87h	FSR1H
808h	BSR	888h	BSR	908h	BSR	988h	BSR	A08h	BSR	A88h	BSR	B08h	BSR	B88h	BSR
809h	WREG	889h	WREG	909h	WREG	989h	WREG	A09h	WREG	A89h	WREG	B09h	WREG	B89h	WREG
80Ah	PCLATH	88Ah	PCLATH	90Ah	PCLATH	98Ah	PCLATH	A0Ah	PCLATH	A8Ah	PCLATH	B0Ah	PCLATH	B8Ah	PCLATH
80Bh	INTCON	88Bh	INTCON	90Bh	INTCON	98Bh	INTCON	A0Bh	INTCON	A8Bh	INTCON	B0Bh	INTCON	B8Bh	INTCON
80Ch	_	88Ch		90Ch		98Ch		A0Ch		A8Ch	_	B0Ch	_	B8Ch	_
80Dh	_	88Dh		90Dh	_	98Dh	_	A0Dh	_	A8Dh	_	B0Dh	_	B8Dh	_
80Eh	_	88Eh		90Eh		98Eh		A0Eh		A8Eh		B0Eh		B8Eh	_
80Fh	_	88Fh		90Fh	_	98Fh	_	A0Fh		A8Fh		B0Fh		B8Fh	_
810h	_	890h		910h	_	990h	_	A10h	_	A90h	_	B10h	_	B90h	_
811h	_	891h		911h	_	991h	_	A11h		A91h		B11h		B91h	_
812h	_	892h	_	912h	_	992h	_	A12h	_	A92h	_	B12h	_	B92h	_
813h	_	893h		913h		993h		A13h	_	A93h	_	B13h	_	B93h	_
814h	_	894h	_	914h	_	994h	_	A14h	I	A94h		B14h	_	B94h	_
815h	_	895h		915h		995h		A15h	_	A95h	_	B15h	_	B95h	_
816h	_	896h		916h	_	996h	_	A16h		A96h		B16h		B96h	_
817h	_	897h		917h	_	997h	_	A17h		A97h	_	B17h		B97h	_
818h	_	898h		918h	_	998h	_	A18h	_	A98h	_	B18h	_	B98h	_
819h	_	899h		919h		999h		A19h		A99h	_	B19h	_	B99h	_
81Ah	_	89Ah		91Ah		99Ah		A1Ah		A9Ah		B1Ah		B9Ah	_
81Bh	_	89Bh		91Bh	_	99Bh	_	A1Bh	_	A9Bh	_	B1Bh	_	B9Bh	_
81Ch	_	89Ch		91Ch	_	99Ch	_	A1Ch	_	A9Ch	_	B1Ch	_	B9Ch	_
81Dh	_	89Dh		91Dh	_	99Dh	_	A1Dh		A9Dh	_	B1Dh		B9Dh	_
81Eh	_	89Eh		91Eh	_	99Eh	_	A1Eh	I	A9Eh		B1Eh		B9Eh	_
81Fh	_	89Fh		91Fh		99Fh		A1Fh		A9Fh		B1Fh		B9Fh	_
820h		8A0h		920h		9A0h		A20h		AA0h		B20h		BA0h	
	Unimplemented Read as '0'														
	Redu as 0		Reau as 0		Reau as 0		Reau as 0		Redu ds 0		Redu dS 0		Redu ds 0		Redu as 0
86Fh		8EFh		96Fh		9EFh		A6Fh		AEFh		B6Fh		BEFh	
870h	A000000	8F0h	A0000000	970h	A0000000	9F0h	A0000000	A70h	A 0000000	AF0h	A 0000000	B70h	A 0000000	BF0h	A 0000000
	Accesses 70h – 7Fh														
87Fh	7011 7111	8FFh	7011 7111	97Fh	7011 7111	9FFh	7011 7111	A7Fh	7011 7111	AFFh	7011 7111	B7Fh	7011 7111	BFFh	7011 7111
8/FN		orrn		9/FN		arru		A/FN		AFFN		B/FN		BEEN	

Legend: = Unimplemented data memory locations, read as '0'.

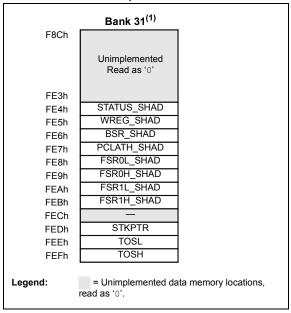
	CFI
	Lege
ı	

TABLE 3-6: PIC16(L)F1825/1829 MEMORY MAP, BANKS 24-31

	BANK 24		BANK 25		BANK 26		BANK 27		BANK 28		BANK 29		BANK 30		BANK 31
C00h	INDF0	C80h	INDF0	D00h	INDF0	D80h	INDF0	E00h	INDF0	E80h	INDF0	F00h	INDF0	F80h	INDF0
C01h	INDF1	C81h	INDF1	D01h	INDF1	D81h	INDF1	E01h	INDF1	E81h	INDF1	F01h	INDF1	F81h	INDF1
C02h	PCL	C82h	PCL	D02h	PCL	D82h	PCL	E02h	PCL	E82h	PCL	F02h	PCL	F82h	PCL
C03h	STATUS	C83h	STATUS	D03h	STATUS	D83h	STATUS	E03h	STATUS	E83h	STATUS	F03h	STATUS	F83h	STATUS
C04h	FSR0L	C84h	FSR0L	D04h	FSR0L	D84h	FSR0L	E04h	FSR0L	E84h	FSR0L	F04h	FSR0L	F84h	FSR0L
C05h	FSR0H	C85h	FSR0H	D05h	FSR0H	D85h	FSR0H	E05h	FSR0H	E85h	FSR0H	F05h	FSR0H	F85h	FSR0H
C06h	FSR1L	C86h	FSR1L	D06h	FSR1L	D86h	FSR1L	E06h	FSR1L	E86h	FSR1L	F06h	FSR1L	F86h	FSR1L
C07h	FSR1H	C87h	FSR1H	D07h	FSR1H	D87h	FSR1H	E07h	FSR1H	E87h	FSR1H	F07h	FSR1H	F87h	FSR1H
C08h	BSR	C88h	BSR	D08h	BSR	D88h	BSR	E08h	BSR	E88h	BSR	F08h	BSR	F88h	BSR
C09h	WREG	C89h	WREG	D09h	WREG	D89h	WREG	E09h	WREG	E89h	WREG	F09h	WREG	F89h	WREG
C0Ah	PCLATH	C8Ah	PCLATH	D0Ah	PCLATH	D8Ah	PCLATH	E0Ah	PCLATH	E8Ah	PCLATH	F0Ah	PCLATH	F8Ah	PCLATH
C0Bh	INTCON	C8Bh	INTCON	D0Bh	INTCON	D8Bh	INTCON	E0Bh	INTCON	E8Bh	INTCON	F0Bh	INTCON	F8Bh	INTCON
C0Ch	_	C8Ch	_	D0Ch	_	D8Ch		E0Ch		E8Ch	_	F0Ch		F8Ch	
C0Dh	_	C8Dh		D0Dh		D8Dh	_	E0Dh	_	E8Dh		F0Dh	_	F8Dh	
C0Eh	_	C8Eh	_	D0Eh	_	D8Eh		E0Eh		E8Eh	_	F0Eh	_	F8Eh	
C0Fh	_	C8Fh	_	D0Fh	_	D8Fh		E0Fh		E8Fh	_	F0Fh		F8Fh	
C10h	_	C90h	_	D10h	_	D90h		E10h		E90h	_	F10h		F90h	
C11h	_	C91h		D11h		D91h	_	E11h	_	E91h		F11h	_	F91h	
C12h	_	C92h		D12h		D92h	_	E12h	_	E92h		F12h	_	F92h	
C13h	_	C93h		D13h		D93h	_	E13h	_	E93h		F13h	_	F93h	
C14h	_	C94h		D14h		D94h	_	E14h	_	E94h		F14h	_	F94h	
C15h	_	C95h		D15h		D95h	_	E15h	_	E95h		F15h	_	F95h	
C16h	_	C96h		D16h		D96h	_	E16h	_	E96h		F16h	_	F96h	
C17h	_	C97h		D17h		D97h	_	E17h	_	E97h		F17h	_	F97h	See Table 3-7 for
C18h	_	C98h		D18h		D98h	_	E18h	_	E98h		F18h	_	F98h	register mapping
C19h	_	C99h		D19h		D99h	_	E19h	_	E99h		F19h	_	F99h	details
C1Ah	_	C9Ah	_	D1Ah	_	D9Ah	_	E1Ah	_	E9Ah	_	F1Ah	_	F9Ah	
C1Bh	_	C9Bh	_	D1Bh	_	D9Bh	_	E1Bh	_	E9Bh	_	F1Bh	_	F9Bh	
C1Ch	_	C9Ch		D1Ch		D9Ch	_	E1Ch	_	E9Ch		F1Ch	_	F9Ch	
C1Dh	_	C9Dh	_	D1Dh	_	D9Dh	_	E1Dh	_	E9Dh		F1Dh	_	F9Dh	
C1Eh	_	C9Eh	_	D1Eh	_	D9Eh	_	E1Eh	_	E9Eh	_	F1Eh	_	F9Eh	
C1Fh	_	C9Fh	_	D1Fh	_	D9Fh	_	E1Fh	_	E9Fh	_	F1Fh	_	F9Fh	
C20h		CA0h		D20h		DA0h		E20h		EA0h		F20h		FA0h	
	Unimplemented														
	Read as '0'														
C6Fh		CEFh		D6Fh		DEFh		E6Fh		EEFh		F6Fh		FEFh	
C70h		CF0h		D70h		DF0h		E70h		EF0h		F70h		FF0h	
	Accesses 70h – 7Fh														
	-	055	/UII — /FN	D7E:	/UII — /FII	DEE:	-		/011 – / F11		/UII — /FII	F7F;	/UII — /FII	,	/011 – / F11
CFFh		CFFh		D7Fh		DFFh		E7Fh		EFFh		F7Fh		FFFh	

Legend: = Unimplemented data memory locations, read as '0'.

TABLE 3-7: PIC16(L)F1825/1829 MEMORY MAP, BANK 31



3.2.6 SPECIAL FUNCTION REGISTERS SUMMARY

The Special Function Register summary for the device family are as follows:

Device	Bank(s)	Page No.		
	0	31		
	1	32		
	2	33		
	3	34		
	4	35		
PIC16(L)F1825	5	36		
PIC16(L)F1829	6	37		
	7	38		
	8	39		
	9-30	40		
	31	41		

TABLE 3-8: SPECIAL FUNCTION REGISTER SUMMARY

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets			
Bank 0														
000h ⁽¹⁾	INDF0	Addressing the (not a physical		es contents of	FSR0H/FSR0	L to address	data memory	1		xxxx xxxx	xxxx xxxx			
001h ⁽¹⁾	INDF1	Addressing the (not a physical		es contents of	FSR1H/FSR1	L to address	data memory	1		xxxx xxxx	xxxx xxxx			
002h ⁽¹⁾	PCL	Program Cou	nter (PC) Lea	st Significant B	syte					0000 0000	0000 0000			
003h ⁽¹⁾	STATUS	_	_	_	TO	PD	Z	DC	С	1 1000	q quuu			
004h ⁽¹⁾	FSR0L	Indirect Data	Memory Addr	ess 0 Low Poir	nter					0000 0000	uuuu uuuu			
005h ⁽¹⁾	FSR0H	Indirect Data	Memory Addr		0000 0000	0000 0000								
006h ⁽¹⁾	FSR1L	Indirect Data	Memory Addr		0000 0000	uuuu uuuu								
007h ⁽¹⁾	FSR1H	Indirect Data	direct Data Memory Address 1 High Pointer											
008h ⁽¹⁾	BSR	_	_	_			BSR<4:0>			0 0000	0 0000			
009h ⁽¹⁾	WREG	Working Regi	ister							0000 0000	uuuu uuuu			
00Ah ⁽¹⁾	PCLATH	_	Write Buffer f	or the upper 7	bits of the Pro	ogram Counte	r			-000 0000	-000 0000			
00Bh ⁽¹⁾	INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 0000	0000 0000			
00Ch	PORTA	_	_	RA5	RA4	RA3	RA2	RA1	RA0	xx xxxx	xx xxxx			
00Dh	PORTB ⁽²⁾	RB7	RB6	RB5	RB4	_	_	_	_	xxxx	xxxx			
00Eh	PORTC	RC7 ⁽²⁾	RC6 ⁽²⁾	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	xxxx xxxx			
00Fh	_	Unimplement	ed							_	_			
010h	_	Unimplement	ed							_	_			
011h	PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000			
012h	PIR2	OSFIF	C2IF	C1IF	EEIF	BCL1IF	_	_	CCP2IF	0000 00	0000 00			
013h	PIR3	_	ı	CCP4IF	CCP3IF	TMR6IF	_	TMR4IF	_	00 0-0-	00 0-0-			
014h	PIR4 ⁽²⁾	_	ı	ı	_	_	_	BCL2IF	SSP2IF	00	00			
015h	TMR0	Timer0 Modu	le Register							xxxx xxxx	uuuu uuuu			
016h	TMR1L	Holding Regis	ster for the Le	ast Significant	Byte of the 16	6-bit TMR1 Re	gister			xxxx xxxx	uuuu uuuu			
017h	TMR1H	Holding Regis	ster for the Mo	st Significant E	Byte of the 16	-bit TMR1 Re	gister			xxxx xxxx	uuuu uuuu			
018h	T1CON	TMR1CS1	TMR1CS0	T1CKP	S<1:0>	T10SCEN	T1SYNC	_	TMR10N	0000 00-0	uuuu uu-u			
019h	T1GCON	TMR1GE	T1GPOL	T1GTM	T1GSPM	T <u>1GGO</u> / DONE	T1GVAL	T1GSS	S<1:0>	0000 0x00	uuuu uxuu			
01Ah	TMR2	Timer2 Modu	le Register		0000 0000	0000 0000								
01Bh	PR2	Timer2 Period	d Register							1111 1111	1111 1111			
01Ch	T2CON	_		T2OUTP	'S<3:0>		TMR2ON	T2CKP	S<1:0>	-000 0000	-000 0000			
01Dh	_	Unimplement	ed		_	_								
01Eh	CPSCON0	CPSON	CPSRM	_	_	CPSRN	G<1:0>	CPSOUT	T0XCS	00 0000	00 0000			
01Fh	CPSCON1	_	_	_	_		CPSCI	H<3:0>		0000	0000			

Note 1: These registers can be addressed from any bank.

- 2: PIC16(L)F1829 only.
- 3: PIC16(L)F1825 only.
- 4: Unimplemented, read as '1'.

TABLE 3-8: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Bit 7		<u> </u>		1	I INLOIST			1	- <i>-</i> ,		1	I
080hf ⁽¹⁾ INDF0	Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0		Value on all other Resets
Comparison Com	Bank 1											
(not a physical register) PCL	080h ⁽¹⁾	INDF0			es contents of	FSR0H/FSR0	L to address	data memory	/		xxxx xxxx	xxxx xxxx
D83hf STATUS	081h ⁽¹⁾	INDF1			es contents of	FSR1H/FSR1	L to address	data memory	/		xxxx xxxx	xxxx xxxx
084h ⁽¹⁾ FSROL	082h ⁽¹⁾	PCL	Program Cou	ınter (PC) Lea	st Significant E	Byte					0000 0000	0000 0000
085h ⁽¹⁾ FSR0H	083h ⁽¹⁾	STATUS	_	_	_	TO	PD	Z	DC	С	1 1000	q quuu
086h(1) FSR1L	084h ⁽¹⁾	FSR0L	Indirect Data	Memory Addr	ess 0 Low Poir	nter					0000 0000	uuuu uuuu
087h(1) FSR1H	085h ⁽¹⁾	FSR0H	Indirect Data	Memory Addr		0000 0000	0000 0000					
D88h ⁽¹⁾ D8R	086h ⁽¹⁾	FSR1L	Indirect Data	Memory Addr	ess 1 Low Poir	nter					0000 0000	uuuu uuuu
089h(1) WREG Working Register 0000 0000 uuuu uu u	087h ⁽¹⁾	FSR1H	Indirect Data	Memory Addr	ess 1 High Poi	nter					0000 0000	0000 0000
08Ah ⁽¹⁾ PCLATH — Write Buffer for the upper 7 bits of the Program Counter -000 0000 -0000 0000 0000 0000 0000 00	088h ⁽¹⁾	BSR	_	_	_			BSR<4:0>			0 0000	0 0000
08Bh(1) INTCON GIE PEIE TMR0IE INTE IOCIE TMR0IF INTF IOCIF 0000 0000 08Ch TRISA — — TRISA5 TRISA4 TRISA3 TRISA2 TRISA1 TRISA0 —-11 1111 —-11 1111 —-11 1111 —-11 111 —-11 111 —-11 111 —-11 111 —-11	089h ⁽¹⁾	WREG	Working Reg	ister							0000 0000	uuuu uuuu
OBCh	08Ah ⁽¹⁾	PCLATH	_	Write Buffer f	or the upper 7	bits of the Pro	gram Counte	er			-000 0000	-000 0000
08Dh TRISB ⁽²⁾ TRISB7 TRISB6 TRISB5 TRISB4 — — — 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 11	08Bh ⁽¹⁾	INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 0000	0000 0000
08Eh TRISC TRISC(7) TRISC6(2) TRISC5 TRISC4 TRISC3 TRISC2 TRISC1 TRISC0 1111 <th< td=""><td>08Ch</td><td>TRISA</td><td>_</td><td>_</td><td>TRISA5</td><td>TRISA4</td><td>TRISA3</td><td>TRISA2</td><td>TRISA1</td><td>TRISA0</td><td>11 1111</td><td>11 1111</td></th<>	08Ch	TRISA	_	_	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	11 1111	11 1111
D8Fh	08Dh	TRISB ⁽²⁾	TRISB7	TRISB6	TRISB5	TRISB4	_	_	_	_	1111	1111
O90h	08Eh	TRISC	TRISC7 ⁽²⁾	TRISC6 ⁽²⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111
O91h	08Fh	_	Unimplement	ted							_	_
O92h	090h	_	Unimplement	ted							_	_
093h PIE3 — — CCP4IE CCP3IE TMR6IE — TMR4IE —	091h	PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
094h PIE4 ⁽²⁾ — — — — BCL2IE SSP2IE —	092h	PIE2	OSFIE	C2IE	C1IE	EEIE	BCL1IE	_	_	CCP2IE	0000 00	0000 00
095h OPTION_REG WPUEN INTEDG TMR0CS TMR0SE PSA PS<2:0> 1111 </td <td>093h</td> <td>PIE3</td> <td>_</td> <td>_</td> <td>CCP4IE</td> <td>CCP3IE</td> <td>TMR6IE</td> <td>_</td> <td>TMR4IE</td> <td>_</td> <td>00 0-0-</td> <td>00 0-0-</td>	093h	PIE3	_	_	CCP4IE	CCP3IE	TMR6IE	_	TMR4IE	_	00 0-0-	00 0-0-
096h PCON STKOVF STKUNF — RMCLR RI POR BOR 00 11qq qq 0 097h WDTCON — — WDTPS<4:0> SWDTEN 01 0110 01 0 098h OSCTUNE — — TUN<5:0> 00 0000 00 0	094h	PIE4 ⁽²⁾	_	_	_	_	_	_	BCL2IE	SSP2IE	00	00
097h WDTCON — — WDTPS<4:0> SWDTEN 01 0110 01 01 0910 01 00	095h	OPTION_REG	WPUEN	INTEDG	TMR0CS	TMR0SE	PSA		PS<2:0>		1111 1111	1111 1111
098h OSCTUNE — — TUN<5:0> — O000000000000000000000000000000000000	096h	PCON	STKOVF	STKUNF	_	_	RMCLR	RI	POR	BOR	00 11qq	qq qquu
099h OSCCON SPLLEN IRCF<3:0> — SCS<1:0> 0011 1-00 0011 1 09Ah OSCSTAT T1OSCR PLLR OSTS HFIOFR HFIOFL MFIOFR LFIOFR HFIOFS 10q0 0q00 qqqq 0 09Bh ADRESL A/D Result Register Low XXXX XXXX uuuu 0 <t< td=""><td>097h</td><td>WDTCON</td><td>_</td><td>_</td><td></td><td>V</td><td>VDTPS<4:0></td><td></td><td></td><td>SWDTEN</td><td>01 0110</td><td>01 0110</td></t<>	097h	WDTCON	_	_		V	VDTPS<4:0>			SWDTEN	01 0110	01 0110
09Ah OSCSTAT T10SCR PLLR OSTS HFI0FR HFI0FL MFI0FR LFI0FR HFI0FS 10q0 0q00 qqqq qqqq 09Bh ADRESL A/D Result Register Low XXXX XXXXX XXXX XXXX XXXX XXXX </td <td>098h</td> <td>OSCTUNE</td> <td>_</td> <td>_</td> <td></td> <td></td> <td>TUN<</td> <td>5:0></td> <td></td> <td></td> <td>00 0000</td> <td>00 0000</td>	098h	OSCTUNE	_	_			TUN<	5:0>			00 0000	00 0000
09Bh ADRESL A/D Result Register Low xxxx	099h	OSCCON	SPLLEN		IRCF	<3:0>		_	SCS	<1:0>	0011 1-00	0011 1-00
09Ch ADRESH A/D Result Register High xxxx xxxx <th< td=""><td>09Ah</td><td>OSCSTAT</td><td>T10SCR</td><td colspan="8">PLLR OSTS HFIOFR HFIOFL MFIOFR LFIOFR HFIOFS</td><td>qqqq qq0q</td></th<>	09Ah	OSCSTAT	T10SCR	PLLR OSTS HFIOFR HFIOFL MFIOFR LFIOFR HFIOFS								qqqq qq0q
09Dh ADCON0 — CHS<4:0> GO/DONE ADON -000 0000 -000 0 09Eh ADCON1 ADFM ADCS<2:0> — ADNREF ADPREF<1:0> 0000 -000 0000 -	09Bh	ADRESL	A/D Result R	egister Low							xxxx xxxx	uuuu uuuu
09Eh ADCON1 ADFM ADCS<2:0> — ADNREF ADPREF<1:0> 0000 -000 0000 -	09Ch	ADRESH	A/D Result R	egister High							xxxx xxxx	uuuu uuuu
	09Dh	ADCON0	_		ADON	-000 0000	-000 0000					
DOCK Unimplemented	09Eh	ADCON1	ADFM		ADCS<2:0>		_	ADNREF	ADPRE	F<1:0>	0000 -000	0000 -000
	09Fh	_	Unimplement	ted				•	•		_	_

x = unknown, u = unchanged, q = value depends on condition, - = unimplemented, r = reserved. Shaded locations are unimplemented, read as '0'. Legend:

Note 1: These registers can be addressed from any bank.

2: PIC16(L)F1829 only.

3: PIC16(L)F1825 only.

TABLE 3-8: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

						,				Value on	Value on all
Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR, BOR	other Resets
Bank 2											
100h ⁽¹⁾	INDF0	Addressing to (not a physic	nis location use al register)	es contents of	FSR0H/FSR0	L to address	data memory	У		xxxx xxxx	xxxx xxxx
101h ⁽¹⁾	INDF1	Addressing to (not a physic	nis location use al register)	es contents of	FSR1H/FSR1	L to address	data memory	y		xxxx xxxx	xxxx xxxx
102h ⁽¹⁾	PCL	Program Cou	inter (PC) Leas	st Significant B	yte					0000 0000	0000 0000
103h ⁽¹⁾	STATUS	_	_	-	TO	PD	Z	DC	С	1 1000	q quuu
104h ⁽¹⁾	FSR0L	Indirect Data	Memory Addre	ess 0 Low Poir	nter					0000 0000	uuuu uuuu
105h ⁽¹⁾	FSR0H	Indirect Data	Memory Addre		0000 0000	0000 0000					
106h ⁽¹⁾	FSR1L	Indirect Data	Memory Addre		0000 0000	uuuu uuuu					
107h ⁽¹⁾	FSR1H	Indirect Data	Memory Addre		0000 0000	0000 0000					
108h ⁽¹⁾	BSR	_	BSR<4:0>								
109h ⁽¹⁾	WREG	Working Reg	ister							0000 0000	uuuu uuuu
10Ah ⁽¹⁾	PCLATH	_	Write Buffer fo	or the upper 7	bits of the Pro	gram Counte	r			-000 0000	-000 0000
10Bh ⁽¹⁾	INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 0000	0000 0000
10Ch	LATA	_	_	LATA5	LATA4	_	LATA2	LATA1	LATA0	xx -xxx	uu -uuu
10Dh	LATB ⁽²⁾	LATB7	LATB6	LATB5	LATB4	_	_	_	_	xxxx	xxxx
10Eh	LATC	LATC7 ⁽²⁾	LATC6 ⁽²⁾	LATC5	LATC4	LATC3	LATC2	LATC1	LATC0	xxxx xxxx	uuuu uuuu
10Fh		Unimplement	ted							_	_
110h		Unimplement	ted							_	_
111h	CM1CON0	C1ON	C1OUT	C10E	C1POL	_	C1SP	C1HYS	C1SYNC	0000 -100	0000 -100
112h	CM1CON1	C1INTP	C1INTN	C1PCH	l<1:0>	1	_	C1NCH1	C1NCH0	00000	00000
113h	CM2CON0	C2ON	C2OUT	C2OE	C2POL	ı	C2SP	C2HYS	C2SYNC	0000 -100	0000 -100
114h	CM2CON1	C2INTP	C2INTN	C2PCH	l<1:0>	ı	_	C2NCH	H<1:0>	000000	000000
115h	CMOUT	_	_	-	_	_	_	MC2OUT	MC10UT	00	00
116h	BORCON	SBOREN	_	-	_	_	_	_	BORRDY	1q	uu
117h	FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDAFV	R<1:0>	ADFVF	R<1:0>	0000 00p0	0000 00p0
118h	DACCON0	DACEN	DACLPS	DACOE	_	DACPS	S<1:0>	_	DACNSS	000- 00-0	000- 00-0
119h	DACCON1	_	_	-			DACR<4:0>			0 0000	0 0000
11Ah	SRCON0	SRLEN		SRCLK<2:0>	SRQEN SRNQEN SRPS SRPR					0000 0000	0000 0000
11Bh	SRCON1	SRSPE	SRSCKE	SRSC2E ⁽²⁾	SRSC1E	SRRPE	0000 0000	0000 0000			
11Ch	_	Unimplement	ted							_	
11Dh	APFCON0	RXDTSEL	SDO1SEL ⁽³⁾	SS1SEL ⁽³⁾	_	T1GSEL	TXCKSEL	_	_	000- 0000	000- 0000
11Eh	APFCON1	_	_	SDO2SEL ⁽²⁾	SS2SEL ⁽²⁾	P1DSEL	P1CSEL	P2BSEL	CCP2SEL	00 0000	00 0000
11Fh	_	Unimplement	ted							_	

Legend: x = unknown, u = unchanged, q = value depends on condition, -= unimplemented, r = reserved. Shaded locations are unimplemented, read as '0'.

Note 1: These registers can be addressed from any bank.

2: PIC16(L)F1829 only.

3: PIC16(L)F1825 only.

TABLE 3-8: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
Bank 3											
180h ⁽¹⁾	INDF0	Addressing to (not a physic		es contents of	FSR0H/FSR0	L to address	data memory	/		xxxx xxxx	xxxx xxxx
181h ⁽¹⁾	INDF1	Addressing to (not a physic		es contents of	FSR1H/FSR1	L to address	data memory	/		xxxx xxxx	xxxx xxxx
182h ⁽¹⁾	PCL	Program Cou	ınter (PC) Lea	st Significant E	syte					0000 0000	0000 0000
183h ⁽¹⁾	STATUS	_	_	_	TO	PD	Z	DC	С	1 1000	q quuu
184h ⁽¹⁾	FSR0L	Indirect Data	Memory Addr	ess 0 Low Poir	nter					0000 0000	uuuu uuuu
185h ⁽¹⁾	FSR0H	Indirect Data	Memory Addr	ess 0 High Poi	nter					0000 0000	0000 0000
186h ⁽¹⁾	FSR1L	Indirect Data	Memory Addr	ess 1 Low Poir	nter					0000 0000	uuuu uuuu
187h ⁽¹⁾	FSR1H	Indirect Data	Memory Addr	ess 1 High Poi	nter					0000 0000	0000 0000
188h ⁽¹⁾	BSR	_	_	_			BSR<4:0>			0 0000	0 0000
189h ⁽¹⁾	WREG	Working Reg	ister							0000 0000	uuuu uuuu
18Ah ⁽¹⁾	PCLATH	_	Write Buffer f	or the upper 7	bits of the Pro	ogram Counte	er			-000 0000	-000 0000
18Bh ⁽¹⁾	INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 0000	0000 0000
18Ch	ANSELA	_	_	_	ANSA4	_	ANSA2	ANSA1	ANSA0	1 -111	1 -111
18Dh	ANSELB ⁽²⁾	ANSB7	ANSB6	ANSB5	ANSB4	_	_	_	_	1111	1111
18Eh	ANSELC	ANSC7 ⁽²⁾	ANSC6 ⁽²⁾	_	_	ANSC3	ANSC2	ANSC1	ANSC0	11 1111	11 1111
18Fh	_	Unimplement	ted	•	•					_	_
190h	_	Unimplement	ted							_	_
191h	EEADRL	EEPROM / P	rogram Memo	ry Address Re	gister Low By	te				0000 0000	0000 0000
192h	EEADRH	_(4)	EEPROM / P	rogram Memo	ry Address Re	gister High B	yte			1000 0000	1000 0000
193h	EEDATL	EEPROM / P	rogram Memo	ry Read Data	Register Low	Byte				xxxx xxxx	uuuu uuuu
194h	EEDATH	_	_	EEPROM / Pi	ogram Memo	ry Read Data	Register Hig	gh Byte		xx xxxx	uu uuuu
195h	EECON1	EEPGD	CFGS	LWLO	FREE	WRERR	WREN	WR	RD	0000 x000	0000 q000
196h	EECON2	EEPROM co	ntrol register 2		•	•				0000 0000	0000 0000
197h	_	Unimplement	ted							_	_
198h	_	Unimplement	ted							_	_
199h	RCREG	USART Rece	eive Data Regi	ster						0000 0000	0000 0000
19Ah	TXREG	USART Tran	smit Data Reg	ister						0000 0000	0000 0000
19Bh	SPBRGL	Baud Rate G	enerator Data	Register Low						0000 0000	0000 0000
19Ch	SPBRGH	Baud Rate G		0000 0000	0000 0000						
19Dh	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
19Eh	TXSTA	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	0000 0010	0000 0010
19Fh	BAUDCON	ABDOVF	RCIDL	_	SCKP	BRG16	_	WUE	ABDEN	01-0 0-00	01-0 0-00

x = unknown, u = unchanged, q = value depends on condition, - = unimplemented, r = reserved. Shaded locations are unimplemented, read as '0'. Legend:

Note 1: These registers can be addressed from any bank. 2: PIC16(L)F1829 only.

3: PIC16(L)F1825 only.

TABLE 3-8: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
Bank 4											
200h ⁽¹⁾	INDF0	Addressing the (not a physical		es contents of	FSR0H/FSR0	L to address	data memory	/		xxxx xxxx	xxxx xxxx
201h ⁽¹⁾	INDF1	Addressing the (not a physical		es contents of	FSR1H/FSR1	L to address	data memory	/		xxxx xxxx	xxxx xxxx
202h ⁽¹⁾	PCL	Program Cou	ınter (PC) Lea	st Significant B	syte					0000 0000	0000 0000
203h ⁽¹⁾	STATUS	_	_	_	TO	PD	Z	DC	С	1 1000	q quuu
204h ⁽¹⁾	FSR0L	Indirect Data	Memory Addr	ess 0 Low Poir	nter	•				0000 0000	uuuu uuuu
205h ⁽¹⁾	FSR0H	Indirect Data	Memory Addr		0000 0000	0000 0000					
206h ⁽¹⁾	FSR1L	Indirect Data	Memory Addr		0000 0000	uuuu uuuu					
207h ⁽¹⁾	FSR1H	Indirect Data	Memory Addr	0000 0000	0000 0000						
208h ⁽¹⁾	BSR	_	_	_			BSR<4:0>			0 0000	0 0000
209h ⁽¹⁾	WREG	Working Reg	ister							0000 0000	uuuu uuuu
20Ah ⁽¹⁾	PCLATH	_	Write Buffer f	or the upper 7	bits of the Pro	ogram Counte	er			-000 0000	-000 0000
20Bh ⁽¹⁾	INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 0000	0000 0000
20Ch	WPUA	_	_	WPUA5	WPUA4	WPUA3	WPUA2	WPUA1	WPUA0	11 1111	11 1111
20Dh	WPUB	WPUB7	WPUB6	WPUB5	WPUB4	_	_	_	_	1111	1111
20Eh	WPUC	WPUC7 ⁽²⁾	WPUC6 ⁽²⁾	WPUC5	WPUC4	WPUC3	WPUC2	WPUC1	WPUC0	1111 1111	1111 1111
20Fh	_	Unimplement	ted							_	_
210h	_	Unimplement	ted							_	_
211h	SSP1BUF	Synchronous	Serial Port Re	eceive Buffer/T	ransmit Regis	ster				xxxx xxxx	uuuu uuuu
212h	SSP1ADD				ADD<7	':0>				0000 0000	0000 0000
213h	SSP1MSK				MSK<7	' :0>				1111 1111	1111 1111
214h	SSP1STAT	SMP	CKE	D/Ā	Р	S	R/W	UA	BF	0000 0000	0000 0000
215h	SSP1CON1	WCOL	SSPOV	SSPEN	CKP		SSPM	1<3:0>		0000 0000	0000 0000
216h	SSP1CON2	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	0000 0000	0000 0000
217h	SSP1CON3	ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	0000 0000	0000 0000
218h	_	Unimplement	ted							_	_
219h	SSP2BUF ⁽²⁾	Synchronous	Serial Port Re	eceive Buffer/T	ransmit Regis	ster				xxxx xxxx	uuuu uuuu
21Ah	SSP2ADD ⁽²⁾			0000 0000	0000 0000						
21Bh	SSP2MSK ⁽²⁾			1111 1111	1111 1111						
21Ch	SSP2STAT ⁽²⁾	SMP	CKE	D/A	Р	S	R/W	UA	BF	0000 0000	0000 0000
21Dh	SSP2CON1 ⁽²⁾	WCOL	SSPOV	SSPEN	CKP		SSPM	1<3:0>		0000 0000	0000 0000
21Eh	SSP2CON2 ⁽²⁾	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	0000 0000	0000 0000
21Fh	SSP2CON3 ⁽²⁾	ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	0000 0000	0000 0000

x = unknown, u = unchanged, q = value depends on condition, - = unimplemented, r = reserved. Shaded locations are unimplemented, read as '0'.

Note 1: These registers can be addressed from any bank.

2: PIC16(L)F1829 only.

3: PIC16(L)F1825 only.

TABLE 3-8: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

IABLE	3-0. 3	PECIAL F	ONO HOI	1 INEGIST		ואולו (טנ	JITTINGL	.U)							
Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets				
Bank 5															
280h ⁽¹⁾	INDF0	Addressing the (not a physical		es contents of	FSR0H/FSR0	L to address	data memory	1		xxxx xxxx	xxxx xxxx				
281h ⁽¹⁾	INDF1		ddressing this location uses contents of FSR1H/FSR1L to address data memory ot a physical register)												
282h ⁽¹⁾	PCL	Program Cou	rogram Counter (PC) Least Significant Byte												
283h ⁽¹⁾	STATUS	_	_	_	TO	PD	Z	DC	С	1 1000	q quuu				
284h ⁽¹⁾	FSR0L	Indirect Data	Memory Addr	ess 0 Low Poi	nter					0000 0000	uuuu uuuu				
285h ⁽¹⁾	FSR0H	Indirect Data	Memory Addr	ess 0 High Po	inter					0000 0000	0000 0000				
286h ⁽¹⁾	FSR1L	Indirect Data	Memory Addr	ess 1 Low Poi	nter					0000 0000	uuuu uuuu				
287h ⁽¹⁾	FSR1H	Indirect Data	Memory Addr		0000 0000	0000 0000									
288h ⁽¹⁾	BSR	_	_	_			BSR<4:0>			0 0000	0 0000				
289h ⁽¹⁾	WREG	Working Regi	ster	•						0000 0000	uuuu uuuu				
28Ah ⁽¹⁾	PCLATH	_	Write Buffer f	or the upper 7	bits of the Pro	gram Counte	er			-000 0000	-000 0000				
28Bh ⁽¹⁾	INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 0000	0000 0000				
28Ch	_	Unimplement	ed	•				•		_	_				
28Dh		Unimplement	ed							_	_				
28Eh		Unimplement	ed							_	_				
28Fh		Unimplement	ed							_	_				
290h		Unimplement	ed							_	_				
291h	CCPR1L	Capture/Com	pare/PWM Re	egister 1 (LSB)						xxxx xxxx	uuuu uuuu				
292h	CCPR1H	Capture/Com	pare/PWM Re	egister 1 (MSB)					xxxx xxxx	uuuu uuuu				
293h	CCP1CON	P1M	<1:0>	DC1B	<1:0>		CCP1N	M<3:0>		0000 0000	0000 0000				
294h	PWM1CON	P1RSEN		•	F	P1DC<6:0>				0000 0000	0000 0000				
295h	CCP1AS	CCP1ASE		CCP1AS<2:0>	•	PSS1A	C<1:0>	PSS1B	D<1:0>	0000 0000	0000 0000				
296h	PSTR1CON	_		_	STR1SYNC	STR1D	STR1C	STR1B	STR1A	0 0001	0 0001				
297h	_	Unimplement	ed	•						_	_				
298h	CCPR2L	Capture/Com	pare/PWM Re	egister 2 (LSB)						xxxx xxxx	uuuu uuuu				
299h	CCPR2H	Capture/Com	xxxx xxxx	uuuu uuuu											
29Ah	CCP2CON	P2M1	P2M0	DC2B1	DC2B0	CCP2M3	CCP2M2	CCP2M1	CCP2M0	0000 0000	0000 0000				
29Bh	PWM2CON	P2RSEN	P2DC6	P2DC5	P2DC4	P2DC3	P2DC2	P2DC1	P2DC0	0000 0000	0000 0000				
29Ch	CCP2AS	CCP2ASE	CCP2AS2	CCP2AS1	CCP2AS0	PSS2AC1	PSS2AC0	PSS2BD1	PSS2BD0	0000 0000	0000 0000				
29Dh	PSTR2CON	_	_	_	STR2SYNC	STR2D	STR2C	STR2B	STR2A	0 0001	0 0001				
29Eh	CCPTMRS	C4TSEL1	C4TSEL0	C3TSEL1	C3TSEL0	C2TSEL1	C2TSEL0	C1TSEL1	C1TSEL0	0000 0000	0000 0000				
29Fh	_	Unimplement	ed		•		•			_	_				

x = unknown, u = unchanged, q = value depends on condition, - = unimplemented, r = reserved. Shaded locations are unimplemented, read as '0'. Legend:

Note 1: These registers can be addressed from any bank.

2: PIC16(L)F1829 only.
3: PIC16(L)F1825 only.

TABLE 3-8: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
Bank 6											
300h ⁽¹⁾	INDF0	Addressing the (not a physical		es contents of	FSR0H/FSR0	L to address	data memory	1		xxxx xxxx	xxxx xxxx
301h ⁽¹⁾	INDF1		Addressing this location uses contents of FSR1H/FSR1L to address data memory not a physical register)							xxxx xxxx	xxxx xxxx
302h ⁽¹⁾	PCL	Program Cou	nter (PC) Lea	st Significant E	Syte					0000 0000	0000 0000
303h ⁽¹⁾	STATUS	_	_	_	TO	PD	Z	DC	С	1 1000	q quuu
304h ⁽¹⁾	FSR0L	Indirect Data	Memory Addr	ess 0 Low Poir	nter					0000 0000	uuuu uuuu
305h ⁽¹⁾	FSR0H	Indirect Data	Memory Addr	ess 0 High Poi	nter					0000 0000	0000 0000
306h ⁽¹⁾	FSR1L	Indirect Data	Memory Addr	ess 1 Low Poir	nter					0000 0000	uuuu uuuu
307h ⁽¹⁾	FSR1H	Indirect Data	Memory Addr	ess 1 High Poi	nter					0000 0000	0000 0000
308h ⁽¹⁾	BSR	_	_	_			BSR<4:0>			0 0000	0 0000
309h ⁽¹⁾	WREG	Working Reg	ister							0000 0000	uuuu uuuu
30Ah ⁽¹⁾	PCLATH	_	Write Buffer f	or the upper 7	bits of the Pro	ogram Counte	r			-000 0000	-000 0000
30Bh ⁽¹⁾	INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 0000	0000 0000
30Ch	_	Unimplement	Jnimplemented							_	_
30Dh	_	Unimplement	Unimplemented							_	_
30Eh	_	Unimplement	ed							_	_
30Fh	_	Unimplement	ed							_	_
310h	_	Unimplement	ed							_	_
311h	CCPR3L	Capture/Com	pare/PWM Re	gister 3 (LSB)						xxxx xxxx	uuuu uuuu
312h	CCPR3H	Capture/Com	pare/PWM Re	egister 3 (MSB))					xxxx xxxx	uuuu uuuu
313h	CCP3CON	_	_	DC3B1	DC3B0	ССР3М3	CCP3M2	CCP3M1	CCP3M0	00 0000	00 0000
314h	_	Unimplement	ed							_	_
315h	_	Unimplement	ed							_	_
316h	_	Unimplement	ed							_	_
317h	_	Unimplement	ed							_	_
318h	CCPR4L	Capture/Com	pare/PWM Re	egister 4 (LSB)						xxxx xxxx	uuuu uuuu
319h	CCPR4H	Capture/Com	pare/PWM Re	egister 4 (MSB))					xxxx xxxx	uuuu uuuu
31Ah	CCP4CON	_	_	DC4B1	DC4B0	CCP4M3	CCP4M2	CCP4M1	CCP4M0	00 0000	00 0000
31Bh	_	Unimplement	ed							_	_
31Ch	_	Unimplemented							_	_	
31Dh	_	Unimplemented							_	_	
31Eh	_	Unimplemented							_	_	
31Fh	_	Unimplement	ed							_	_

Note 1: These registers can be addressed from any bank.

2: PIC16(L)F1829 only.

3: PIC16(L)F1825 only.

TABLE 3-8: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
Bank 7											
380h ⁽¹⁾	INDF0	Addressing the (not a physical		es contents of	FSR0H/FSR0	L to address	data memory	1		xxxx xxxx	xxxx xxxx
381h ⁽¹⁾	INDF1	Addressing the (not a physical		es contents of	FSR1H/FSR1	L to address	data memory	1		xxxx xxxx	xxxx xxxx
382h ⁽¹⁾	PCL	Program Cou	nter (PC) Lea	st Significant B	yte					0000 0000	0000 0000
383h ⁽¹⁾	STATUS	_	_	_	TO	PD	Z	DC	С	1 1000	q quuu
384h ⁽¹⁾	FSR0L	Indirect Data	Memory Addr	ess 0 Low Poir	nter		•	•		0000 0000	uuuu uuuu
385h ⁽¹⁾	FSR0H	Indirect Data	Memory Addr	ess 0 High Poi	nter					0000 0000	0000 0000
386h ⁽¹⁾	FSR1L	Indirect Data	Memory Addr	ess 1 Low Poir	nter					0000 0000	uuuu uuuu
387h ⁽¹⁾	FSR1H	Indirect Data	Memory Addr	ess 1 High Poi	nter					0000 0000	0000 0000
388h ⁽¹⁾	BSR	_	_	_			BSR<4:0>			0 0000	0 0000
389h ⁽¹⁾	WREG	Working Regi	ster							0000 0000	uuuu uuuu
38Ah ⁽¹⁾	PCLATH	_	Write Buffer f	or the upper 7	bits of the Pro	gram Counte	r			-000 0000	-000 0000
38Bh ⁽¹⁾	INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 0000	0000 0000
38Ch	INLVLA	_		INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	00 0100	00 0100
38Dh	INLVLB ⁽²⁾	INLVLB7	INLVLB6	INLVLB5	INLVLB4	_	_	_	_	0000	0000
38Eh	INLVLC ⁽³⁾	_	_	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	00 0000	00 0000
	INLVLC ⁽²⁾	INLVLC7	INLVLC6	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	1111 1111	1111 1111
38Fh	_	Unimplement	ed							_	_
390h	_	Unimplement	ed							_	_
391h	IOCAP	_	_	IOCAP5	IOCAP4	IOCAP3	IOCAP2	IOCAP1	IOCAP0	00 0000	00 0000
392h	IOCAN	_		IOCAN5	IOCAN4	IOCAN3	IOCAN2	IOCAN1	IOCAN0	00 0000	00 0000
393h	IOCAF	_	_	IOCAF5	IOCAF4	IOCAF3	IOCAF2	IOCAF1	IOCAF0	00 0000	00 0000
394h	IOCBP ⁽²⁾	IOCBP7	IOCBP6	IOCBP5	IOCBP4	_	_	_	_	0000	0000
395h	IOCBN ⁽²⁾	IOCBN7	IOCBN6	IOCBN5	IOCBN4	_	_	_	_	0000	0000
396h	IOCBF ⁽²⁾	IOCBF7	IOCBF6	IOCBF5	IOCBF4	_	_	_	_	0000	0000
397h	_	Unimplement	ed							_	_
398h	_	Unimplement	ed							_	_
399h	_	Unimplement	ed							_	_
39Ah	CLKRCON	CLKREN	CLKROE	CLKRSLR	CLKRD	C<1:0>	(CLKRDIV<2:0	>	0011 0000	0011 0000
39Bh	_	Unimplement	Unimplemented						_	_	
39Ch	MDCON	MDEN	MDOE	MDSLR	MDOPOL	MDOUT	_	_	MDBIT	00100	00100
39Dh	MDSRC	MDMSODIS	_	_	_		MDMS	S<3:0>		x xxxx	u uuuu
39Eh	MDCARL	MDCLODIS	MDCLPOL	MDCLSYNC	_	— MDCL<3:0>				xxx- xxxx	uuu- uuuu
39Fh	MDCARH	MDCHODIS	MDCHPOL	MDCHSYNC	_		MDCH	1<3:0>		xxx- xxxx	uuu- uuuu

Legend: x = unknown, u = unchanged, q = value depends on condition, -= unimplemented, r = reserved. Shaded locations are unimplemented, read as '0'.

Note 1: These registers can be addressed from any bank.

2: PIC16(L)F1829 only.

3: PIC16(L)F1825 only.

TABLE 3-8: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
Bank 8											
400h ⁽¹⁾	INDF0	Addressing the (not a physical		es contents of	FSR0H/FSR0	L to address	data memory	r		xxxx xxxx	xxxx xxxx
401h ⁽¹⁾	INDF1		dressing this location uses contents of FSR1H/FSR1L to address data memory t a physical register)							xxxx xxxx	xxxx xxxx
402h ⁽¹⁾	PCL	Program Cou	inter (PC) Lea	st Significant E	Byte					0000 0000	0000 0000
403h ⁽¹⁾	STATUS	_	_	_	TO	PD	Z	DC	С	1 1000	q quuu
404h ⁽¹⁾	FSR0L	Indirect Data	Memory Addr	ess 0 Low Poir	nter					0000 0000	uuuu uuuu
405h ⁽¹⁾	FSR0H	Indirect Data	Memory Addr	ess 0 High Poi	nter					0000 0000	0000 0000
406h ⁽¹⁾	FSR1L	Indirect Data	Memory Addr	ess 1 Low Poir	nter					0000 0000	uuuu uuuu
407h ⁽¹⁾	FSR1H	Indirect Data	Memory Addr	ess 1 High Poi	nter					0000 0000	0000 0000
408h ⁽¹⁾	BSR	_	_	_			BSR<4:0>			0 0000	0 0000
409h ⁽¹⁾	WREG	Working Reg	ister							0000 0000	uuuu uuuu
40Ah ⁽¹⁾	PCLATH	_	Write Buffer f	or the upper 7	bits of the Pro	ogram Counte	er			-000 0000	-000 0000
40Bh ⁽¹⁾	INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 0000	0000 0000
40Ch	_	Unimplement	ted							_	_
40Dh	_	Unimplement	ted							_	_
40Eh	_	Unimplement	ted							_	_
40Fh	_	Unimplement	ted							_	_
410h	_	Unimplement	ted							_	_
411h	_	Unimplement	ted							_	_
412h	_	Unimplement	ted							_	_
413h	_	Unimplement	ted							_	_
414h	_	Unimplement	ted							_	_
415h	TMR4	Timer4 Modu	le Register							0000 0000	0000 0000
416h	PR4	Timer4 Perio	d Register							1111 1111	1111 1111
417h	T4CON	_		T4OUTP	'S<3:0>		TMR4ON	T4CKP	S<1:0>	-000 0000	-000 0000
418h	_	Unimplement	ted							_	_
419h	_	Unimplement	ted							_	_
41Ah	_	Unimplement	ted							_	_
41Bh	_	Unimplement	ted							_	_
41Ch	TMR6	Timer6 Modu	le Register							0000 0000	0000 0000
41Dh	PR6	Timer6 Perio	Timer6 Period Register							1111 1111	1111 1111
41Eh	T6CON	_		T6OUTP	°S<3:0>		TMR6ON	T6CKP	S<1:0>	-000 0000	-000 0000
41Fh	_	Unimplement	ted							_	_

Note 1: These registers can be addressed from any bank.

2: PIC16(L)F1829 only.

3: PIC16(L)F1825 only.

TABLE 3-8: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
Banks	9-30										
x00h/ x80h ⁽¹⁾	INDF0	Addressing to (not a physic		es contents of	FSR0H/FSR0	L to address	data memory	r		xxxx xxxx	xxxx xxxx
x00h/ x81h ⁽¹⁾	INDF1	Addressing to (not a physic		es contents of	FSR1H/FSR1	L to address	data memory	r		xxxx xxxx	xxxx xxxx
x02h/ x82h ⁽¹⁾	PCL	Program Cou	ınter (PC) Lea	st Significant E	Byte					0000 0000	0000 0000
x03h/ x83h ⁽¹⁾	STATUS	_	_	_	TO	PD	Z	DC	С	1 1000	q quuu
x04h/ x84h ⁽¹⁾	FSR0L	Indirect Data	Memory Addr	ess 0 Low Poir	nter					0000 0000	uuuu uuuu
x05h/ x85h ⁽¹⁾	FSR0H	Indirect Data	Indirect Data Memory Address 0 High Pointer						0000 0000	0000 0000	
x06h/ x86h ⁽¹⁾	FSR1L	Indirect Data	Memory Addr	ess 1 Low Poir	nter					0000 0000	uuuu uuuu
x07h/ x87h ⁽¹⁾	FSR1H	Indirect Data	Memory Addr	ess 1 High Poi	nter					0000 0000	0000 0000
x08h/ x88h ⁽¹⁾	BSR	_	_	_			BSR<4:0>			0 0000	0 0000
x09h/ x89h ⁽¹⁾	WREG	Working Reg	ister							0000 0000	uuuu uuuu
x0Ah/ x8Ah ⁽¹⁾	PCLATH	_	Write Buffer f	or the upper 7	bits of the Pro	ogram Counte	er			-000 0000	-000 0000
x0Bh/ x8Bh ⁽¹⁾	INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 0000	0000 0000
x0Ch/ x8Ch — x1Fh/	_	Unimplement	ted							_	_
x9Fh											

 $\begin{tabular}{ll} \textbf{Legend:} & x = unknown, u = unchanged, q = value depends on condition, -= unimplemented, r = reserved. \\ & Shaded locations are unimplemented, read as '0'. \\ \end{tabular}$

Note 1: These registers can be addressed from any bank.

2: PIC16(L)F1829 only.

3: PIC16(L)F1825 only.

TABLE 3-8: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
Bank 3	1										
F80h ⁽¹⁾	INDF0	Addressing to (not a physic		es contents of	FSR0H/FSR0)L to address	data memory	y		xxxx xxxx	xxxx xxxx
F81h ⁽¹⁾	INDF1	Addressing to (not a physic		es contents of	FSR1H/FSR1	L to address	data memory	/		xxxx xxxx	xxxx xxxx
F82h ⁽¹⁾	PCL	Program Cou	ınter (PC) Lea	st Significant E	Byte					0000 0000	0000 0000
F83h ⁽¹⁾	STATUS	_	_	_	TO	PD	Z	DC	С	1 1000	q quuu
F84h ⁽¹⁾	FSR0L	Indirect Data	Memory Addr	ess 0 Low Poir	nter			•	•	0000 0000	uuuu uuuu
F85h ⁽¹⁾	FSR0H	Indirect Data	Memory Addr	ess 0 High Poi	inter					0000 0000	0000 0000
F86h ⁽¹⁾	FSR1L	Indirect Data	Memory Addr	ess 1 Low Poir	nter					0000 0000	uuuu uuuu
F87h ⁽¹⁾	FSR1H	Indirect Data	Memory Addr	ess 1 High Poi	inter					0000 0000	0000 0000
F88h ⁽¹⁾	BSR	_	_	_			BSR<4:0>			0 0000	0 0000
F89h ⁽¹⁾	WREG	Working Reg	ister		1					0000 0000	uuuu uuuu
F8Ah ⁽¹⁾	PCLATH	_	Write Buffer f	or the upper 7	bits of the Pro	ogram Counte	er			-000 0000	-000 0000
F8Bh ⁽¹⁾	INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 0000	0000 0000
F8Ch	_	Unimplement	ted				1			_	_
FE3h											
FE4h	STATUS_ SHAD	_	_	_	_	_	Z_SHAD	DC_SHAD	C_SHAD	xxx	uuu
FE5h	WREG_ SHAD	Working Reg	ister Shadow							0000 0000	uuuu uuuu
FE6h	BSR_ SHAD	_	_	_	Bank Select	Register Sha	dow			x xxxx	u uuuu
FE7h	PCLATH_ SHAD	_	Program Cou	inter Latch Hig	h Register Sh	adow				-xxx xxxx	uuuu uuuu
FE8h	FSR0L_ SHAD	Indirect Data	Memory Addr	ess 0 Low Poir	nter Shadow					xxxx xxxx	uuuu uuuu
FE9h	FSR0H_ SHAD	Indirect Data	Memory Addr	ess 0 High Poi	inter Shadow					xxxx xxxx	uuuu uuuu
FEAh	FSR1L_ SHAD	Indirect Data	Memory Addr	ess 1 Low Poir	nter Shadow					xxxx xxxx	uuuu uuuu
FEBh	FSR1H_ SHAD	Indirect Data	Memory Addr	ess 1 High Poi	inter Shadow					xxxx xxxx	uuuu uuuu
FECh	_	Unimplement	ted							_	_
FEDh	STKPTR	_	_	_	Current Stac	k pointer				1 1111	1 1111
			f-Stack Low byte							1	L
FEEh	TOSL	Top-of-Stack	Low byte		•					xxxx xxxx	uuuu uuuu

Legend: x = unknown, u = unchanged, q = value depends on condition, -= unimplemented, r = reserved. Shaded locations are unimplemented, read as '0'.

Note 1: These registers can be addressed from any bank.

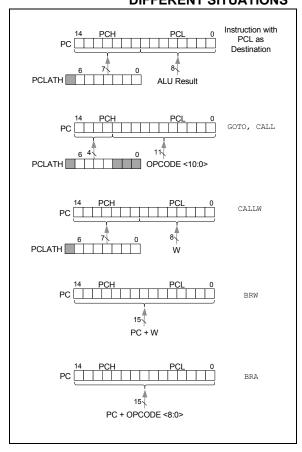
2: PIC16(L)F1829 only.

3: PIC16(L)F1825 only.

3.3 PCL and PCLATH

The Program Counter (PC) is 15 bits wide. The low byte comes from the PCL register, which is a readable and writable register. The high byte (PC<14:8>) is not directly readable or writable and comes from PCLATH. On any Reset, the PC is cleared. Figure 3-3 shows the five situations for the loading of the PC.

FIGURE 3-3: LOADING OF PC IN DIFFERENT SITUATIONS



3.3.1 MODIFYING PCL

Executing any instruction with the PCL register as the destination simultaneously causes the Program Counter PC<14:8> bits (PCH) to be replaced by the contents of the PCLATH register. This allows the entire contents of the program counter to be changed by writing the desired upper seven bits to the PCLATH register. When the lower eight bits are written to the PCL register, all 15 bits of the program counter will change to the values contained in the PCLATH register and those being written to the PCL register.

3.3.2 COMPUTED GOTO

A computed GOTO is accomplished by adding an offset to the program counter (ADDWF PCL). When performing a table read using a computed GOTO method, care should be exercised if the table location crosses a PCL memory boundary (each 256-byte block). Refer to the Application Note AN556, "Implementing a Table Read" (DS00556).

3.3.3 COMPUTED FUNCTION CALLS

A computed function CALL allows programs to maintain tables of functions and provide another way to execute state machines or look-up tables. When performing a table read using a computed function CALL, care should be exercised if the table location crosses a PCL memory boundary (each 256-byte block).

If using the CALL instruction, the PCH<2:0> and PCL registers are loaded with the operand of the CALL instruction. PCH<6:3> is loaded with PCLATH<6:3>.

The CALLW instruction enables computed calls by combining PCLATH and W to form the destination address. A computed CALLW is accomplished by loading the W register with the desired address and executing CALLW. The PCL register is loaded with the value of W and PCH is loaded with PCLATH.

3.3.4 BRANCHING

The branching instructions add an offset to the PC. This allows relocatable code and code that crosses page boundaries. There are two forms of branching, BRW and BRA. The PC will have incremented to fetch the next instruction in both cases. When using either branching instruction, a PCL memory boundary may be crossed.

If using BRW, load the W register with the desired unsigned address and execute BRW. The entire PC will be loaded with the address PC + 1 + W.

If using BRA, the entire PC will be loaded with PC + 1 +, the signed value of the operand of the BRA instruction.

3.4 Stack

All devices have a 16-level x 15-bit wide hardware stack (refer to Figures 3-4 through 3-7). The stack space is not part of either program or data space. The PC is PUSHed onto the stack when CALL or CALLW instructions are executed or an interrupt causes a branch. The stack is POPed in the event of a RETURN, RETLW or a RETFIE instruction execution. PCLATH is not affected by a PUSH or POP operation.

The stack operates as a circular buffer if the STVREN bit is programmed to '0' (Configuration Word 2). This means that after the stack has been PUSHed sixteen times, the seventeenth PUSH overwrites the value that was stored from the first PUSH. The eighteenth PUSH overwrites the second PUSH (and so on). The STKOVF and STKUNF flag bits will be set on an Overflow/Underflow, regardless of whether the Reset is enabled.

Note 1: There are no instructions/mnemonics called PUSH or POP. These are actions that occur from the execution of the CALL, CALLW, RETURN, RETLW and RETFIE instructions or the vectoring to an interrupt address.

3.4.1 ACCESSING THE STACK

The stack is available through the TOSH, TOSL and STKPTR registers. STKPTR is the current value of the Stack Pointer. TOSH:TOSL register pair points to the TOP of the stack. Both registers are read/writable. TOS is split into TOSH and TOSL due to the 15-bit size of the PC. To access the stack, adjust the value of STKPTR, which will position TOSH:TOSL, then read/write to TOSH:TOSL. STKPTR is five bits to allow detection of overflow and underflow.

Note: Care should be taken when modifying the STKPTR while interrupts are enabled.

During normal program operation, CALL, CALLW and Interrupts will increment STKPTR while RETLW, RETURN, and RETFIE will decrement STKPTR. At any time STKPTR can be inspected to see how much stack is left. The STKPTR always points at the currently used place on the stack. Therefore, a CALL or CALLW will increment the STKPTR and then write the PC, and a return will unload the PC and then decrement the STKPTR.

Reference Figure 3-4 through Figure 3-7 for examples of accessing the stack.

FIGURE 3-4: ACCESSING THE STACK EXAMPLE 1

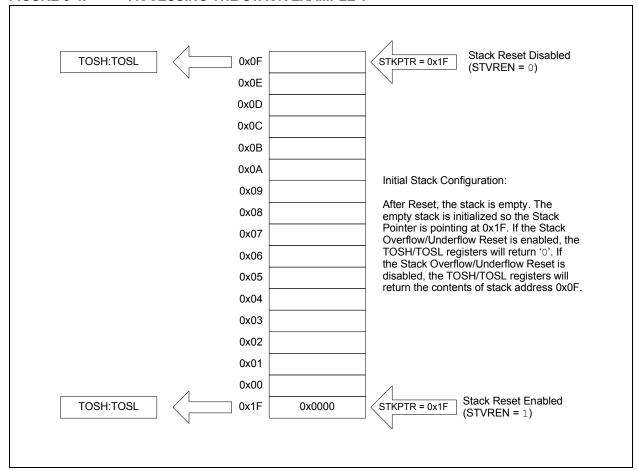
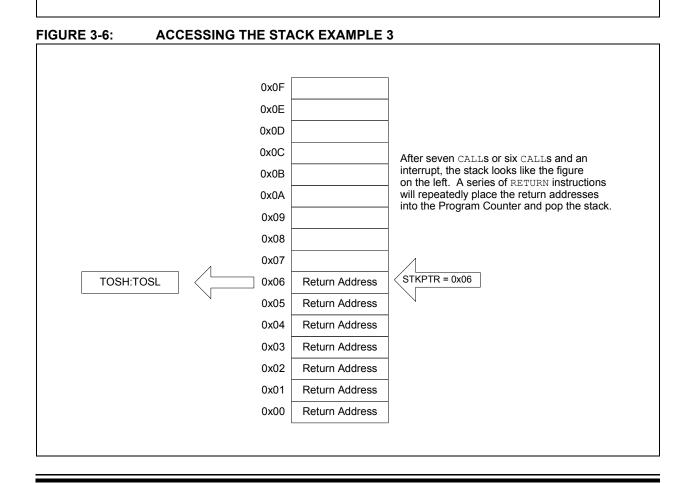
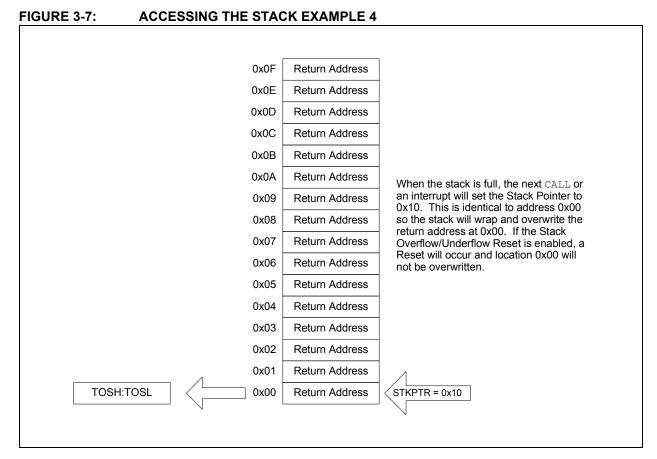


FIGURE 3-5: **ACCESSING THE STACK EXAMPLE 2** 0x0F 0x0E 0x0D 0x0C 0x0B 0x0A 0x09 This figure shows the stack configuration after the first CALL or a single interrupt. 80x0 If a RETURN instruction is executed, the return address will be placed in the 0x07 Program Counter and the Stack Pointer decremented to the empty state (0x1F). 0x06 0x05 0x04 0x03 0x02 0x01 TOSH:TOSL 0x00 Return Address STKPTR = 0x00





3.4.2 OVERFLOW/UNDERFLOW RESET

If the STVREN bit in Configuration Word 2 is programmed to '1', the device will be reset if the stack is PUSHed beyond the sixteenth level or POPed beyond the first level, setting the appropriate bits (STKOVF or STKUNF, respectively) in the PCON register.

3.5 Indirect Addressing

The INDFn registers are not physical registers. Any instruction that accesses an INDFn register actually accesses the register at the address specified by the File Select Registers (FSR). If the FSRn address specifies one of the two INDFn registers, the read will return '0' and the write will not occur (though Status bits may be affected). The FSRn register value is created by the pair FSRnH and FSRnL.

The FSR registers form a 16-bit address that allows an addressing space with 65536 locations. These locations are divided into three memory regions:

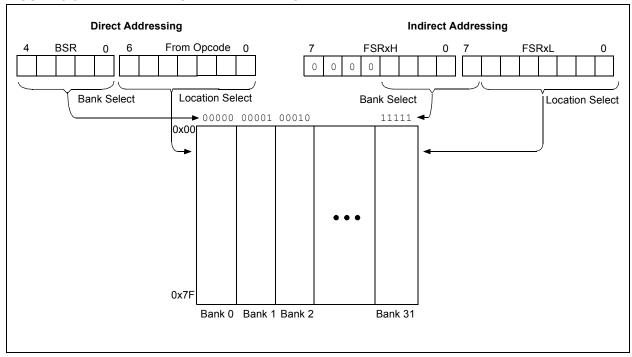
- · Traditional Data Memory
- · Linear Data Memory
- · Program Flash Memory

FIGURE 3-8: INDIRECT ADDRESSING 0x0000 0x0000 **Traditional Data Memory** 0x0FFF 0x0FFF 0x1000 Reserved 0x1FFF 0x2000 Linear **Data Memory** 0x29AF 0x29B0 Reserved **FSR** 0x7FFF Address 0x8000 Range 0x0000 **Program** Flash Memory 0xFFFF 0x7FFF Note: Not all memory regions are completely implemented. Consult device memory tables for memory limits.

3.5.1 TRADITIONAL DATA MEMORY

The traditional data memory is a region from FSR address 0x000 to FSR address 0xFFF. The addresses correspond to the absolute addresses of all SFR, GPR and common registers.

FIGURE 3-9: TRADITIONAL DATA MEMORY MAP



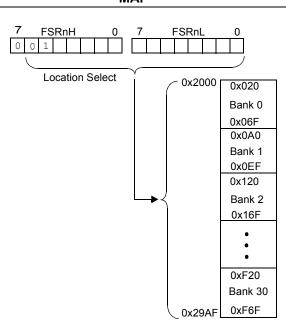
3.5.2 LINEAR DATA MEMORY

The linear data memory is the region from FSR address 0x2000 to FSR address 0x29AF. This region is a virtual region that points back to the 80-byte blocks of GPR memory in all the banks.

Unimplemented memory reads as 0x00. Use of the linear data memory region allows buffers to be larger than 80 bytes because incrementing the FSR beyond one bank will go directly to the GPR memory of the next bank.

The 16 bytes of common memory are not included in the linear data memory region.

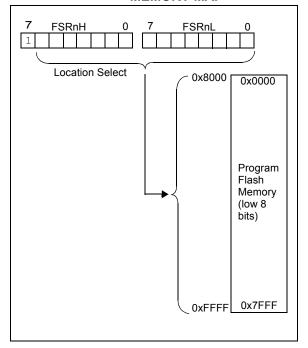
FIGURE 3-10: LINEAR DATA MEMORY MAP



3.5.3 PROGRAM FLASH MEMORY

To make constant data access easier, the entire program Flash memory is mapped to the upper half of the FSR address space. When the MSB of FSRnH is set, the lower 15 bits are the address in program memory which will be accessed through INDF. Only the lower eight bits of each memory location is accessible via INDF. Writing to the program Flash memory cannot be accomplished via the FSR/INDF interface. All instructions that access program Flash memory via the FSR/INDF interface will require one additional instruction cycle to complete.

FIGURE 3-11: PROGRAM FLASH MEMORY MAP



4.0 DEVICE CONFIGURATION

Device Configuration consists of Configuration Word 1 and Configuration Word 2, Code Protection and Device ID

4.1 Configuration Words

There are several Configuration Word bits that allow different oscillator and memory protection options. These are implemented as Configuration Word 1 at 8007h and Configuration Word 2 at 8008h.

Note: The DEBUG bit in Configuration Word 2 is managed automatically by device development tools including debuggers and programmers. For normal device operation, this bit should be maintained as a '1'.

REGISTER 4-1: CONFIGURATION WORD 1

R/P-1/1	R/P-1/1	R/P-1/1	R/P-1/1	R/P-1/1	R/P-1/1
FCMEN	IESO	CLKOUTEN	BORE	N<1:0>	CPD
bit 13					bit 8

R/P-1/1	R/P-1/1						
CP	MCLRE	PWRTE	WDTE	E<1:0>		FOSC<2:0	
bit 7							bit 0

Legend:

R = Readable bit P = Programmable bit U = Unimplemented bit, read as '1'
'0' = Bit is cleared '1' = Bit is set -n = Value when blank or after Bulk Erase

bit 13 FCMEN: Fail-Safe Clock Monitor Enable bit

1 = Fail-Safe Clock Monitor is enabled

0 = Fail-Safe Clock Monitor is disabled

bit 12 IESO: Internal External Switchover bit

1 = Internal/External Switchover mode is enabled

0 = Internal/External Switchover mode is disabled

bit 11 CLKOUTEN: Clock Out Enable bit

If FOSC configuration bits are set to LP, XT, HS modes:

This bit is ignored, CLKOUT function is disabled. Oscillator function on the CLKOUT pin.

All other FOSC modes:

1 = CLKOUT function is disabled. I/O function on the CLKOUT pin.

0 = CLKOUT function is enabled on the CLKOUT pin

bit 10-9 **BOREN<1:0>:** Brown-out Reset Enable bits⁽¹⁾

11 = BOR enabled

 ${\tt 10}$ = BOR enabled during operation and disabled in Sleep

01 = BOR controlled by SBOREN bit of the BORCON register

00 = BOR disabled

bit 8 CPD: Data Code Protection bit⁽²⁾

1 = Data memory code protection is disabled

0 = Data memory code protection is enabled

bit 7 CP: Code Protection bit⁽³⁾

1 = Program memory code protection is disabled

0 = Program memory code protection is enabled

bit 6 MCLRE: RA3/MCLR/VPP Pin Function Select bit

If LVP bit = 1:

This bit is ignored.

If LVP bit = 0:

1 = \overline{MCLR}/VPP pin function is \overline{MCLR} ; Weak pull-up enabled.

0 = MCLR/VPP pin function is digital input; MCLR internally disabled; Weak pull-up under control of WPUA register.

bit 5 **PWRTE**: Power-up Timer Enable bit⁽¹⁾

1 = PWRT disabled

0 = PWRT enabled

bit 4-3 WDTE<1:0>: Watchdog Timer Enable bit

11 = WDT enabled

10 = WDT enabled while running and disabled in Sleep

01 = WDT controlled by the SWDTEN bit in the WDTCON register

00 = WDT disabled

Note 1: Enabling Brown-out Reset does not automatically enable Power-up Timer.

2: The entire data EEPROM will be erased when the code protection is turned off during an erase.

3: The entire program memory will be erased when the code protection is turned off.

REGISTER 4-1: CONFIGURATION WORD 1 (CONTINUED)

bit 2-0 FOSC<2:0>: Oscillator Selection bits

- 111 = ECH: External Clock, High-Power mode (4-32 MHz): device clock supplied to CLKIN pin
- 110 = ECM: External Clock, Medium-Power mode (0.5-4 MHz): device clock supplied to CLKIN pin
- 101 = ECL: External Clock, Low-Power mode (0-0.5 MHz): device clock supplied to CLKIN pin
- 100 = INTOSC oscillator: I/O function on CLKIN pin
- 011 = EXTRC oscillator: External RC circuit connected to CLKIN pin
- 010 = HS oscillator: High-speed crystal/resonator connected between OSC1 and OSC2 pins
- 001 = XT oscillator: Crystal/resonator connected between OSC1 and OSC2 pins
- 000 = LP oscillator: Low-power crystal connected between OSC1 and OSC2 pins
- Note 1: Enabling Brown-out Reset does not automatically enable Power-up Timer.
 - 2: The entire data EEPROM will be erased when the code protection is turned off during an erase.
 - 3: The entire program memory will be erased when the code protection is turned off.

REGISTER 4-2: CONFIGURATION WORD 2

R/P-1/1	R/P-1/1	U-1	R/P-1/1	R/P-1/1	R/P-1/1
LVP ⁽¹⁾	DEBUG ⁽²⁾	_	BORV	STVREN	PLLEN
bit 13					bit 8

U-1	U-1	U-1	R-1	U-1	U-1	R/P-1/1	R/P-1/1
_	_	_	Reserved	_	_	WRT	<1:0>
bit 7							bit 0

Legend:

R = Readable bit P = Programmable bit U = Unimplemented bit, read as '1'
'0' = Bit is cleared '1' = Bit is set -n = Value when blank or after Bulk Erase

Legend:

bit 13 LVP: Low-Voltage Programming Enable bit⁽¹⁾

1 = Low-voltage programming enabled

 $0 = \text{High-voltage on } \overline{\text{MCLR}}$ must be used for programming

bit 12 **DEBUG:** In-Circuit Debugger Mode bit⁽²⁾

1 = In-Circuit Debugger disabled, ICSPCLK and ICSPDAT are general purpose I/O pins 0 = In-Circuit Debugger enabled, ICSPCLK and ICSPDAT are dedicated to the debugger

bit 11 **Unimplemented:** Read as '1'

bit 10 **BORV:** Brown-out Reset Voltage Selection bit⁽³⁾

1 = Brown-out Reset voltage (Vbor), low trip point selected 0 = Brown-out Reset voltage (Vbor), high trip point selected

bit 9 STVREN: Stack Overflow/Underflow Reset Enable bit

1 = Stack Overflow or Underflow will cause a Reset0 = Stack Overflow or Underflow will not cause a Reset

bit 8 PLLEN: PLL Enable bit 1 = 4xPLL enabled

0 = 4xPLL disabled

bit 7-5 **Unimplemented:** Read as '1'

bit 4 Reserved: This location should be programmed to a '1'.

bit 3-2 Unimplemented: Read as '1'

bit 1-0 WRT<1:0>: Flash Memory Self-Write Protection bits

11 = Write protection off

10 = 000h to 1FFh write-protected, 200h to FFFh may be modified by EECON control

01 = 000h to 7FFh write-protected, 800h to FFFh may be modified by EECON control

00 = 000h to FFFh write-protected, no addresses may be modified by EECON control

Note 1: The LVP bit cannot be programmed to '0' when Programming mode is entered via LVP.

2: The DEBUG bit in Configuration Word is managed automatically by device development tools including debuggers and programmers. For normal device operation, this bit should be maintained as a '1'.

3: See Vbor parameter for specific trip point voltages.

4.2 Code Protection

Code protection allows the device to be protected from unauthorized access. Program memory protection and data EEPROM protection are controlled independently. Internal access to the program memory and data EEPROM are unaffected by any code protection setting.

4.2.1 PROGRAM MEMORY PROTECTION

The entire program memory space is protected from external reads and writes by the \overline{CP} bit in Configuration Word 1. When \overline{CP} = 0, external reads and writes of program memory are inhibited and a read will return all '0's. The CPU can continue to read program memory, regardless of the protection bit settings. Writing the program memory is dependent upon the write protection setting. See Section 4.3 "Write Protection" for more information.

4.2.2 DATA EEPROM PROTECTION

The entire data EEPROM is protected from external reads and writes by the CPD bit. When CPD = 0, external reads and writes of data EEPROM are inhibited. The CPU can continue to read and write data EEPROM regardless of the protection bit settings.

4.3 Write Protection

Write protection allows the device to be protected from unintended self-writes. Applications, such as bootloader software, can be protected while allowing other regions of the program memory to be modified.

The WRT<1:0> bits in Configuration Word 2 define the size of the program memory block that is protected.

4.4 User ID

Four memory locations (8000h-8003h) are designated as ID locations where the user can store checksum or other code identification numbers. These locations are readable and writable during normal execution. See Section 11.5 "User ID, Device ID and Configuration Word Access" for more information on accessing these memory locations. For more information on checksum calculation, see the "PIC16F/LF182X/PIC12F/LF1822 Memory Programming Specification" (DS41390).

4.5 Device ID and Revision ID

The memory location 8006h is where the Device ID and Revision ID are stored. The upper nine bits hold the Device ID. The lower five bits hold the Revision ID. See Section 11.5 "User ID, Device ID and Configuration Word Access" for more information on accessing these memory locations.

Development tools, such as device programmers and debuggers, may be used to read the Device ID and Revision ID.

REGISTER 4-3: DEVICEID: DEVICE ID REGISTER⁽¹⁾

R	R	R	R	R	R
DEV8	DEV7	DEV6	DEV5	DEV4	DEV3
bit 13					bit 8

R	R	R	R	R	R	R	R
DEV2	DEV1	DEV0	REV4	REV3	REV2	REV1	REV0
bit 7							bit 0

Legend:			
R = Readable bit	'0' = Bit is cleared	'1' = Bit is set	

bit 13-5 **DEV<8:0>:** Device ID bits 1001111011 = PIC16F1825 1001111111 = PIC16F1829 101000011 = PIC16LF1825 101000111 = PIC16LF1829

bit 4-0 **REV<4:0>:** Revision ID bits

These bits are used to identify the revision.

Note 1: This location cannot be written.

5.0 OSCILLATOR MODULE (WITH FAIL-SAFE CLOCK MONITOR)

5.1 Overview

The oscillator module has a wide variety of clock sources and selection features that allow it to be used in a wide range of applications while maximizing performance and minimizing power consumption. Figure 5-1 illustrates a block diagram of the oscillator module.

Clock sources can be supplied from external oscillators, quartz crystal resonators, ceramic resonators and Resistor-Capacitor (RC) circuits. In addition, the system clock source can be supplied from one of two internal oscillators and PLL circuits, with a choice of speeds selectable via software. Additional clock features include:

- Selectable system clock source between external or internal sources via software.
- Two-Speed Start-up mode, which minimizes latency between external oscillator start-up and code execution.
- Fail-Safe Clock Monitor (FSCM) designed to detect a failure of the external clock source (LP, XT, HS, EC or RC modes) and switch automatically to the internal oscillator.
- Oscillator Start-up Timer (OST) ensures stability of crystal oscillator sources

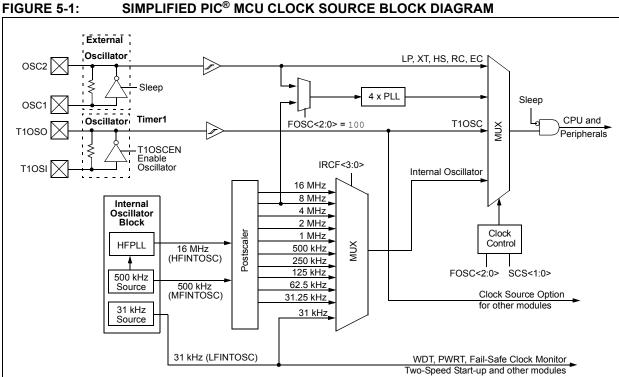
The oscillator module can be configured in one of eight clock modes.

- ECL External Clock Low-Power mode (0 MHz to 0.5 MHz)
- ECM External Clock Medium-Power mode (0.5 MHz to 4 MHz)
- 3. ECH External Clock High-Power mode (4 MHz to 32 MHz)
- 4. LP 32 kHz Low-Power Crystal mode.
- 5. XT Medium Gain Crystal or Ceramic Resonator Oscillator mode (up to 4 MHz)
- HS High Gain Crystal or Ceramic Resonator mode (4 MHz to 20 MHz)
- 7. RC External Resistor-Capacitor (RC).
- 8. INTOSC Internal oscillator (31 kHz to 32 MHz).

Clock Source modes are selected by the FOSC<2:0> bits in the Configuration Word 1. The FOSC bits determine the type of oscillator that will be used when the device is first powered.

The EC clock mode relies on an external logic level signal as the device clock source. The LP, XT and HS clock modes require an external crystal or resonator to be connected to the device. Each mode is optimized for a different frequency range. The RC clock mode requires an external resistor and capacitor to set the oscillator frequency.

The INTOSC internal oscillator block produces low, medium, and high frequency clock sources, designated LFINTOSC, MFINTOSC, and HFINTOSC. (see Internal Oscillator Block, Figure 5-1). A wide selection of device clock frequencies may be derived from these three clock sources.



5.2 Clock Source Types

Clock sources can be classified as external or internal.

External clock sources rely on external circuitry for the clock source to function. Examples are: oscillator modules (EC mode), quartz crystal resonators or ceramic resonators (LP, XT and HS modes) and Resistor-Capacitor (RC) mode circuits.

Internal clock sources are contained internally within the oscillator module. The internal oscillator block has two internal oscillators and a dedicated Phase-Lock Loop (HFPLL) that are used to generate three internal system clock sources: the 16 MHz High-Frequency Internal Oscillator (HFINTOSC), 500 kHZ (MFINTOSC) and the 31 kHz Low-Frequency Internal Oscillator (LFINTOSC).

The system clock can be selected between external or internal clock sources via the System Clock Select (SCS) bits in the OSCCON register. See **Section 5.3** "Clock Switching" for additional information.

5.2.1 EXTERNAL CLOCK SOURCES

An external clock source can be used as the device system clock by performing one of the following actions:

- Program the FOSC<2:0> bits in the Configuration Word 1 to select an external clock source that will be used as the default system clock upon a device Reset.
- Write the SCS<1:0> bits in the OSCCON register to switch the system clock source to:
 - Timer1 Oscillator during run-time, or
 - An external clock source determined by the value of the FOSC bits.

See Section 5.3 "Clock Switching"for more information.

5.2.1.1 EC Mode

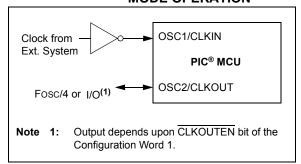
The External Clock (EC) mode allows an externally generated logic level signal to be the system clock source. When operating in this mode, an external clock source is connected to the OSC1 input. OSC2/CLKOUT is available for general purpose I/O or CLKOUT. Figure 5-2 shows the pin connections for EC mode.

EC mode has three power modes to select from through Configuration Word 1:

- High power, 4-32 MHz (FOSC = 111)
- Medium power, 0.5-4 MHz (FOSC = 110)
- Low power, 0-0.5 MHz (FOSC = 101)

The Oscillator Start-up Timer (OST) is disabled when EC mode is selected. Therefore, there is no delay in operation after a Power-on Reset (POR) or wake-up from Sleep. Because the PIC® MCU design is fully static, stopping the external clock input will have the effect of halting the device while leaving all data intact. Upon restarting the external clock, the device will resume operation as if no time had elapsed.

FIGURE 5-2: EXTERNAL CLOCK (EC)
MODE OPERATION



5.2.1.2 LP, XT, HS Modes

The LP, XT and HS modes support the use of quartz crystal resonators or ceramic resonators connected to OSC1 and OSC2 (Figure 5-3). The three modes select a low, medium or high gain setting of the internal inverter-amplifier to support various resonator types and speed.

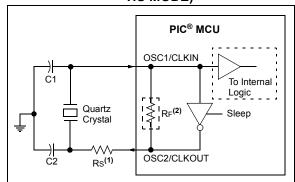
LP Oscillator mode selects the lowest gain setting of the internal inverter-amplifier. LP mode current consumption is the least of the three modes. This mode is designed to drive only 32.768 kHz tuning-fork type crystals (watch crystals).

XT Oscillator mode selects the intermediate gain setting of the internal inverter-amplifier. XT mode current consumption is the medium of the three modes. This mode is best suited to drive resonators with a medium drive level specification.

HS Oscillator mode selects the highest gain setting of the internal inverter-amplifier. HS mode current consumption is the highest of the three modes. This mode is best suited for resonators that require a high drive setting.

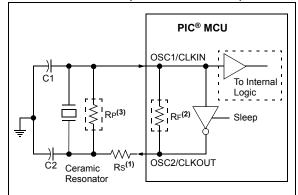
Figure 5-3 and Figure 5-4 show typical circuits for quartz crystal and ceramic resonators, respectively.

FIGURE 5-3: QUARTZ CRYSTAL OPERATION (LP, XT OR HS MODE)



- Note 1: A series resistor (Rs) may be required for quartz crystals with low drive level.
 - 2: The value of RF varies with the Oscillator mode selected (typically between 2 M Ω to 10 M Ω).
- Note 1: Quartz crystal characteristics vary according to type, package and manufacturer. The user should consult the manufacturer data sheets for specifications and recommended application.
 - **2:** Always verify oscillator performance over the VDD and temperature range that is expected for the application.
 - **3:** For oscillator design assistance, reference the following Microchip Application Notes:
 - AN826, "Crystal Oscillator Basics and Crystal Selection for rfPIC[®] and PIC[®] Devices" (DS00826)
 - AN849, "Basic PIC® Oscillator Design" (DS00849)
 - AN943, "Practical PIC® Oscillator Analysis and Design" (DS00943)
 - AN949, "Making Your Oscillator Work" (DS00949)

FIGURE 5-4: CERAMIC RESONATOR
OPERATION
(XT OR HS MODE)



- **Note 1:** A series resistor (Rs) may be required for ceramic resonators with low drive level.
 - 2: The value of RF varies with the Oscillator mode selected (typically between 2 M Ω to 10 M Ω).
 - An additional parallel feedback resistor (RP) may be required for proper ceramic resonator operation.

5.2.1.3 Oscillator Start-up Timer (OST)

If the oscillator module is configured for LP, XT or HS modes, the Oscillator Start-up Timer (OST) counts 1024 oscillations from OSC1. This occurs following a Power-on Reset (POR) and when the Power-up Timer (PWRT) has expired (if configured), or a wake-up from Sleep. During this time, the program counter does not increment and program execution is suspended unless either FSCM or Two-Speed Start-up are enabled. In this case, the code will continue to execute at the selected INTOSC frequency while the OST is counting. The OST ensures that the oscillator circuit, using a quartz crystal resonator or ceramic resonator, has started and is providing a stable system clock to the oscillator module.

In order to minimize latency between external oscillator start-up and code execution, the Two-Speed Clock Start-up mode can be selected (see Section 5.4 "Two-Speed Clock Start-up Mode").

5.2.1.4 4xPLL

The oscillator module contains a 4xPLL that can be used with both external and internal clock sources to provide a system clock source. The input frequency for the 4xPLL must fall within specifications. See the PLL Clock Timing Specifications in Section 30.0 "Electrical Specifications".

The 4xPLL may be enabled for use by one of two methods:

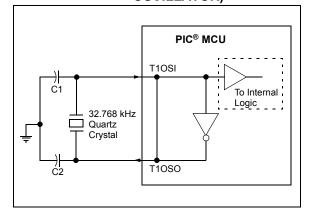
- Program the PLLEN bit in Configuration Word 2 to a '1'.
- Write the SPLLEN bit in the OSCCON register to a '1'. If the PLLEN bit in Configuration Word 2 is programmed to a '1', then the value of SPLLEN is ignored.

5.2.1.5 TIMER1 Oscillator

The Timer1 Oscillator is a separate crystal oscillator that is associated with the Timer1 peripheral. It is optimized for timekeeping operations with a 32.768 kHz crystal connected between the T1OSO and T1OSI device pins.

The Timer1 Oscillator can be used as an alternate system clock source and can be selected during run-time using clock switching. Refer to **Section 5.3** "Clock Switching" for more information.

FIGURE 5-5: QUARTZ CRYSTAL OPERATION (TIMER1 OSCILLATOR)



- Note 1: Quartz crystal characteristics vary according to type, package and manufacturer. The user should consult the manufacturer data sheets for specifications and recommended application.
 - **2:** Always verify oscillator performance over the VDD and temperature range that is expected for the application.
 - **3:** For oscillator design assistance, reference the following Microchip Application Notes:
 - AN826, "Crystal Oscillator Basics and Crystal Selection for rfPIC[®] and PIC[®] Devices" (DS00826)
 - AN849, "Basic PIC® Oscillator Design" (DS00849)
 - AN943, "Practical PIC[®] Oscillator Analysis and Design" (DS00943)
 - AN949, "Making Your Oscillator Work" (DS00949)
 - TB097, "Interfacing a Micro Crystal MS1V-T1K 32.768 kHz Tuning Fork Crystal to a PIC16F690/SS" (DS91097)
 - AN1288, "Design Practices for Low-Power External Oscillators" (DS01288)

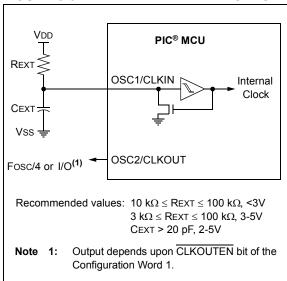
5.2.1.6 External RC Mode

The external Resistor-Capacitor (RC) modes support the use of an external RC circuit. This allows the designer maximum flexibility in frequency choice while keeping costs to a minimum when clock accuracy is not required.

The RC circuit connects to OSC1. OSC2/CLKOUT is available for general purpose I/O or CLKOUT. The function of the OSC2/CLKOUT pin is determined by the state of the CLKOUTEN bit in Configuration Word 1.

Figure 5-6 shows the external RC mode connections.

FIGURE 5-6: EXTERNAL RC MODES



The RC oscillator frequency is a function of the supply voltage, the resistor (REXT) and capacitor (CEXT) values and the operating temperature. Other factors affecting the oscillator frequency are:

- · threshold voltage variation
- · component tolerances
- · packaging variations in capacitance

The user also needs to take into account variation due to tolerance of external RC components used.

5.2.2 INTERNAL CLOCK SOURCES

The device may be configured to use the internal oscillator block as the system clock by performing one of the following actions:

- Program the FOSC<2:0> bits in Configuration
 Word 1 to select the INTOSC clock source, which
 will be used as the default system clock upon a
 device Reset.
- Write the SCS<1:0> bits in the OSCCON register to switch the system clock source to the internal oscillator during run-time. See Section 5.3 "Clock Switching"for more information.

In **INTOSC** mode, OSC1/CLKIN is available for general purpose I/O. OSC2/CLKOUT is available for general purpose I/O or CLKOUT.

The function of the OSC2/CLKOUT pin is determined by the state of the CLKOUTEN bit in Configuration Word 1.

The internal oscillator block has two independent oscillators and a dedicated Phase-Lock Loop, HFPLL that can produce one of three internal system clock sources.

- The HFINTOSC (High-Frequency Internal Oscillator) is factory calibrated and operates at 16 MHz. The HFINTOSC source is generated from the 500 kHz MFINTOSC source and the dedicated Phase-Lock Loop, HFPLL. The frequency of the HFINTOSC can be user-adjusted via software using the OSCTUNE register (Register 5-3).
- The MFINTOSC (Medium-Frequency Internal Oscillator) is factory calibrated and operates at 500 kHz. The frequency of the MFINTOSC can be user-adjusted via software using the OSCTUNE register (Register 5-3).
- The LFINTOSC (Low-Frequency Internal Oscillator) is uncalibrated and operates at 31 kHz.

5.2.2.1 HFINTOSC

The High-Frequency Internal Oscillator (HFINTOSC) is a factory calibrated 16 MHz internal clock source. The frequency of the HFINTOSC can be altered via software using the OSCTUNE register (Register 5-3).

The output of the HFINTOSC connects to a postscaler and multiplexer (see Figure 5-1). One of nine frequencies derived from the HFINTOSC can be selected via software using the IRCF<3:0> bits of the OSCCON register. See Section 5.2.2.7 "Internal Oscillator Clock Switch Timing" for more information.

The HFINTOSC is enabled by:

- Configure the IRCF<3:0> bits of the OSCCON register for the desired HF frequency, and
- FOSC<2:0> = 100, or
- Set the System Clock Source (SCS) bits of the OSCCON register to '1x'.

The High-Frequency Internal Oscillator Ready bit (HFIOFR) of the OSCSTAT register indicates when the HFINTOSC is running and can be utilized.

The High-Frequency Internal Oscillator Status Locked bit (HFIOFL) of the OSCSTAT register indicates when the HFINTOSC is running within 2% of its final value.

The High-Frequency Internal Oscillator Status Stable bit (HFIOFS) of the OSCSTAT register indicates when the HFINTOSC is running within 0.5% of its final value.

5.2.2.2 MFINTOSC

The Medium-Frequency Internal Oscillator (MFINTOSC) is a factory calibrated 500 kHz internal clock source. The frequency of the MFINTOSC can be altered via software using the OSCTUNE register (Register 5-3).

The output of the MFINTOSC connects to a postscaler and multiplexer (see Figure 5-1). One of nine frequencies derived from the MFINTOSC can be selected via software using the IRCF<3:0> bits of the OSCCON register. See Section 5.2.2.7 "Internal Oscillator Clock Switch Timing" for more information.

The MFINTOSC is enabled by:

- Configure the IRCF<3:0> bits of the OSCCON register for the desired HF frequency, and
- FOSC<2:0> = 100, or
- Set the System Clock Source (SCS) bits of the OSCCON register to '1x'

The Medium-Frequency Internal Oscillator Ready bit (MFIOFR) of the OSCSTAT register indicates when the MFINTOSC is running and can be utilized.

5.2.2.3 Internal Oscillator Frequency Adjustment

The 500 kHz internal oscillator is factory calibrated. This internal oscillator can be adjusted in software by writing to the OSCTUNE register (Register 5-3). Since the HFINTOSC and MFINTOSC clock sources are derived from the 500 kHz internal oscillator a change in the OSCTUNE register value will apply to both.

The default value of the OSCTUNE register is '0'. The value is a 6-bit two's complement number. A value of 1Fh will provide an adjustment to the maximum frequency. A value of 20h will provide an adjustment to the minimum frequency.

When the OSCTUNE register is modified, the oscillator frequency will begin shifting to the new frequency. Code execution continues during this shift. There is no indication that the shift has occurred.

OSCTUNE does not affect the LFINTOSC frequency. Operation of features that depend on the LFINTOSC clock source frequency, such as the Power-up Timer (PWRT), Watchdog Timer (WDT), Fail-Safe Clock Monitor (FSCM) and peripherals, are *not* affected by the change in frequency.

5.2.2.4 LFINTOSC

The Low-Frequency Internal Oscillator (LFINTOSC) is an uncalibrated 31 kHz internal clock source.

The output of the LFINTOSC connects to a multiplexer (see Figure 5-1). Select 31 kHz, via software, using the IRCF<3:0> bits of the OSCCON register. See Section 5.2.2.7 "Internal Oscillator Clock Switch Timing" for more information. The LFINTOSC is also the frequency for the Power-up Timer (PWRT), Watchdog Timer (WDT) and Fail-Safe Clock Monitor (FSCM).

The LFINTOSC is enabled by selecting 31 kHz (IRCF<3:0> bits of the OSCCON register = 000) as the system clock source (SCS bits of the OSCCON register = $1\times$), or when any of the following are enabled:

- Configure the IRCF<3:0> bits of the OSCCON register for the desired LF frequency, and
- FOSC<2:0> = 100, or
- Set the System Clock Source (SCS) bits of the OSCCON register to '1x'

Peripherals that use the LFINTOSC are:

- Power-up Timer (PWRT)
- Watchdog Timer (WDT)
- · Fail-Safe Clock Monitor (FSCM)

The Low-Frequency Internal Oscillator Ready bit (LFIOFR) of the OSCSTAT register indicates when the LFINTOSC is running and can be utilized.

5.2.2.5 Internal Oscillator Frequency Selection

The system clock speed can be selected via software using the Internal Oscillator Frequency Select bits IRCF<3:0> of the OSCCON register.

The outputs of the 16 MHz HFINTOSC postscaler and the LFINTOSC connect to multiplexer (see Figure 5-1). The Internal Oscillator Frequency Select bits IRCF<3:0> of the OSCCON register select the frequency output of the internal oscillators. One of the following frequencies can be selected via software:

- 32 MHz (requires 4xPLL)
- 16 MHz
- 8 MHz
- 4 MHz
- 2 MHz
- 1 MHz
- 500 kHz (default after Reset)
- 250 kHz
- 125 kHz
- 62.5 kHz
- 31.25 kHz
- · 31 kHz (LFINTOSC)

Note: Following any Reset, the IRCF<3:0> bits of the OSCCON register are set to '0111' and the frequency selection is set to 500 kHz. The user can modify the IRCF bits to select a different frequency.

The IRCF<3:0> bits of the OSCCON register allow duplicate selections for some frequencies. These duplicate choices can offer system design trade-offs. Lower power consumption can be obtained when changing oscillator sources for a given frequency. Faster transition times can be obtained between frequency changes that use the same oscillator source.

5.2.2.6 32 MHz Internal Oscillator Frequency Selection

The Internal Oscillator Block can be used with the 4xPLL associated with the External Oscillator Block to produce a 32 MHz internal system clock source. The following settings are required to use the 32 MHz internal clock source:

- The FOSC bits in Configuration Word 1 must be set to use the INTOSC source as the device system clock (FOSC<2:0> = 100).
- The SCS bits in the OSCCON register must be cleared to use the clock determined by FOSC<2:0> in Configuration Word 1 (SCS<1:0> = 00).
- The IRCF bits in the OSCCON register must be set to the 8 MHz HFINTOSC set to use (IRCF<3:0> = 1110).
- The SPLLEN bit in the OSCCON register must be set to enable the 4xPLL, or the PLLEN bit of the Configuration Word 2 must be programmed to a '1'.

Note: When using the PLLEN bit of the Configuration Word 2, the 4xPLL cannot be disabled by software and the 8 MHz HFINTOSC option will no longer be available.

The 4xPLL is not available for use with the internal oscillator when the SCS bits of the OSCCON register are set to '1x'. The SCS bits must be set to '00' to use the 4xPLL with the internal oscillator.

5.2.2.7 Internal Oscillator Clock Switch Timing

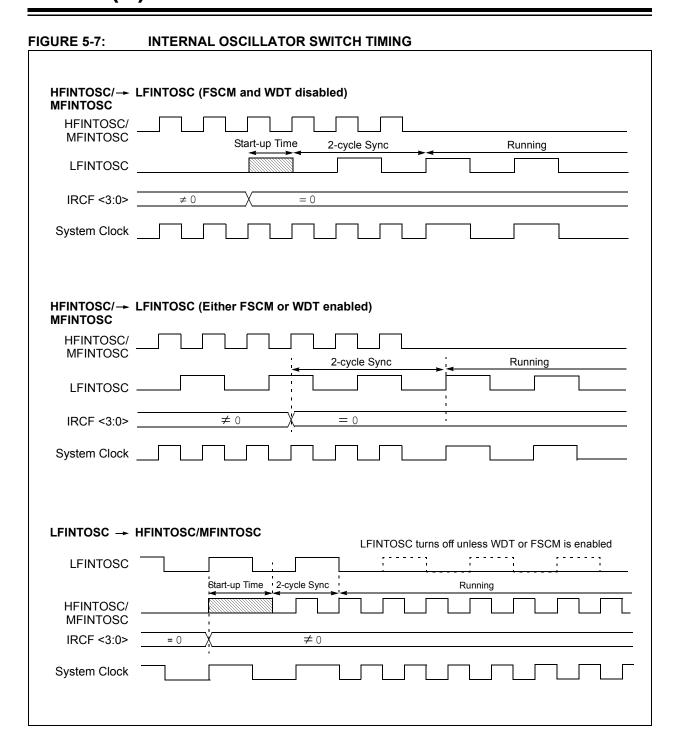
When switching between the HFINTOSC, MFINTOSC and the LFINTOSC, the new oscillator may already be shut down to save power (see Figure 5-7). If this is the case, there is a delay after the IRCF<3:0> bits of the OSCCON register are modified before the frequency selection takes place. The OSCSTAT register will reflect the current active status of the HFINTOSC, MFINTOSC and LFINTOSC oscillators. The sequence of a frequency selection is as follows:

- IRCF<3:0> bits of the OSCCON register are modified.
- If the new clock is shut down, a clock start-up delay is started.
- 3. Clock switch circuitry waits for a falling edge of the current clock.
- The current clock is held low and the clock switch circuitry waits for a rising edge in the new clock.
- 5. The new clock is now active.
- 6. The OSCSTAT register is updated as required.
- 7. Clock switch is complete.

See Figure 5-7 for more details.

If the internal oscillator speed is switched between two clocks of the same source, there is no start-up delay before the new frequency is selected. Clock switching time delays are shown in Table 5-1.

Start-up delay specifications are located in the oscillator tables of Section 30.0 "Electrical Specifications"



5.3 Clock Switching

The system clock source can be switched between external and internal clock sources via software using the System Clock Select (SCS) bits of the OSCCON register. The following clock sources can be selected using the SCS bits:

- Default system oscillator determined by FOSC bits in Configuration Word 1
- Timer1 32 kHz crystal oscillator
- Internal Oscillator Block (INTOSC)

5.3.1 SYSTEM CLOCK SELECT (SCS)

The System Clock Select (SCS) bits of the OSCCON register selects the system clock source that is used for the CPU and peripherals.

- When the SCS bits of the OSCCON register = 00, the system clock source is determined by value of the FOSC<2:0> bits in the Configuration Word 1.
- When the SCS bits of the OSCCON register = 01, the system clock source is the Timer1 oscillator.
- When the SCS bits of the OSCCON register = 1x, the system clock source is chosen by the internal oscillator frequency selected by the IRCF<3:0> bits of the OSCCON register. After a Reset, the SCS bits of the OSCCON register are always cleared.

Note: Any automatic clock switch, which may occur from Two-Speed Start-up or Fail-Safe Clock Monitor, does not update the SCS bits of the OSCCON register. The user can monitor the OSTS bit of the OSCSTAT register to determine the current system clock source.

When switching between clock sources, a delay is required to allow the new clock to stabilize. These oscillator delays are shown in Table 5-1.

5.3.2 OSCILLATOR START-UP TIMER STATUS (OSTS) BIT

The Oscillator Start-up Timer Status (OSTS) bit of the OSCSTAT register indicates whether the system clock is running from the external clock source, as defined by the FOSC<2:0> bits in the Configuration Word 1, or from the internal clock source. In particular, OSTS indicates that the Oscillator Start-up Timer (OST) has timed out for LP, XT or HS modes. The OST does not reflect the status of the Timer1 Oscillator.

5.3.3 TIMER1 OSCILLATOR

The Timer1 Oscillator is a separate crystal oscillator associated with the Timer1 peripheral. It is optimized for timekeeping operations with a 32.768 kHz crystal connected between the T1OSO and T1OSI device pins.

The Timer1 oscillator is enabled using the T1OSCEN control bit in the T1CON register. See **Section 21.0** "Timer1 Module with Gate Control" for more information about the Timer1 peripheral.

5.3.4 TIMER1 OSCILLATOR READY (T1OSCR) BIT

The user must ensure that the Timer1 Oscillator is ready to be used before it is selected as a system clock source. The Timer1 Oscillator Ready (T1OSCR) bit of the OSCSTAT register indicates whether the Timer1 oscillator is ready to be used. After the T1OSCR bit is set, the SCS bits can be configured to select the Timer1 oscillator.

5.4 Two-Speed Clock Start-up Mode

Two-Speed Start-up mode provides additional power savings by minimizing the latency between external oscillator start-up and code execution. In applications that make heavy use of the Sleep mode, Two-Speed Start-up will remove the external oscillator start-up time from the time spent awake and can reduce the overall power consumption of the device. This mode allows the application to wake-up from Sleep, perform a few instructions using the INTOSC internal oscillator block as the clock source and go back to Sleep without waiting for the external oscillator to become stable.

Two-Speed Start-up provides benefits when the oscillator module is configured for LP, XT or HS modes. The Oscillator Start-up Timer (OST) is enabled for these modes and must count 1024 oscillations before the oscillator can be used as the system clock source.

If the oscillator module is configured for any mode other than LP, XT or HS mode, then Two-Speed Start-up is disabled. This is because the external clock oscillator does not require any stabilization time after POR or an exit from Sleep.

If the OST count reaches 1024 before the device enters Sleep mode, the OSTS bit of the OSCSTAT register is set and program execution switches to the external oscillator. However, the system may never operate from the external oscillator if the time spent awake is very short.

Note:

Executing a SLEEP instruction will abort the oscillator start-up time and will cause the OSTS bit of the OSCSTAT register to remain clear.

5.4.1 TWO-SPEED START-UP MODE CONFIGURATION

Two-Speed Start-up mode is configured by the following settings:

- IESO (of the Configuration Word 1) = 1; Internal/External Switchover bit (Two-Speed Start-up mode enabled).
- SCS (of the OSCCON register) = 00.
- FOSC<2:0> bits in the Configuration Word 1 configured for LP, XT or HS mode.

Two-Speed Start-up mode is entered after:

- Power-on Reset (POR) and, if enabled, after Power-up Timer (PWRT) has expired, or
- · Wake-up from Sleep.

Note:

When FSCM is enabled, Two-Speed Start-up will automatically be enabled.

TABLE 5-1: OSCILLATOR SWITCHING DELAYS

Switch From	Switch To	Frequency	Oscillator Delay
Sleep/POR	LFINTOSC ⁽¹⁾ MFINTOSC ⁽¹⁾ HFINTOSC ⁽¹⁾	31 kHz 31.25 kHz-500 kHz 31.25 kHz-16 MHz	Oscillator Warm-up Delay (TWARM)
Sleep/POR	EC, RC ⁽¹⁾	DC – 32 MHz	2 cycles
LFINTOSC	EC, RC ⁽¹⁾	DC – 32 MHz	1 cycle of each
Sleep/POR	Timer1 Oscillator LP, XT, HS ⁽¹⁾	32 kHz-20 MHz	1024 Clock Cycles (OST)
Any clock source	MFINTOSC ⁽¹⁾ HFINTOSC ⁽¹⁾	31.25 kHz-500 kHz 31.25 kHz-16 MHz	2 μs (approx.)
Any clock source	LFINTOSC ⁽¹⁾	31 kHz	1 cycle of each
Any clock source	Timer1 Oscillator	32 kHz	1024 Clock Cycles (OST)
PLL inactive	PLL active	16-32 MHz	2 ms (approx.)

Note 1: PLL inactive.

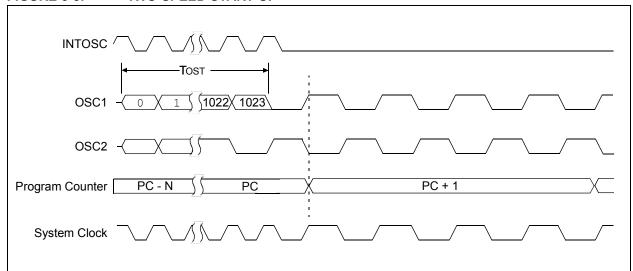
5.4.2 TWO-SPEED START-UP SEQUENCE

- 1. Wake-up from Power-on Reset or Sleep.
- 2. Instructions begin execution by the internal oscillator at the frequency set in the IRCF<3:0> bits of the OSCCON register.
- 3. OST enabled to count 1024 clock cycles.
- 4. OST timed out, wait for falling edge of the internal oscillator.
- 5. OSTS is set.
- 6. System clock held low until the next falling edge of new clock (LP, XT or HS mode).
- System clock is switched to external clock source.

5.4.3 CHECKING TWO-SPEED CLOCK STATUS

Checking the state of the OSTS bit of the OSCSTAT register will confirm if the microcontroller is running from the external clock source, as defined by the FOSC<2:0> bits in the Configuration Word 1, or the internal oscillator.

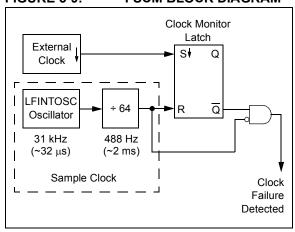




5.5 Fail-Safe Clock Monitor

The Fail-Safe Clock Monitor (FSCM) allows the device to continue operating should the external oscillator fail. The FSCM can detect oscillator failure any time after the Oscillator Start-up Timer (OST) has expired. The FSCM is enabled by setting the FCMEN bit in the Configuration Word 1. The FSCM is applicable to all external Oscillator modes (LP, XT, HS, EC, Timer1 Oscillator and RC).

FIGURE 5-9: FSCM BLOCK DIAGRAM



5.5.1 FAIL-SAFE DETECTION

The FSCM module detects a failed oscillator by comparing the external oscillator to the FSCM sample clock. The sample clock is generated by dividing the LFINTOSC by 64. See Figure 5-9. Inside the fail detector block is a latch. The external clock sets the latch on each falling edge of the external clock. The sample clock clears the latch on each rising edge of the sample clock. A failure is detected when an entire half-cycle of the sample clock elapses before the external clock goes low.

5.5.2 FAIL-SAFE OPERATION

When the external clock fails, the FSCM switches the device clock to an internal clock source and sets the bit flag OSFIF of the PIR2 register. Setting this flag will generate an interrupt if the OSFIE bit of the PIE2 register is also set. The device firmware can then take steps to mitigate the problems that may arise from a failed clock. The system clock will continue to be sourced from the internal clock source until the device firmware successfully restarts the external oscillator and switches back to external operation.

The internal clock source chosen by the FSCM is determined by the IRCF<3:0> bits of the OSCCON register. This allows the internal oscillator to be configured before a failure occurs.

5.5.3 FAIL-SAFE CONDITION CLEARING

The Fail-Safe condition is cleared after a Reset, executing a SLEEP instruction or changing the SCS bits of the OSCCON register. When the SCS bits are changed, the OST is restarted. While the OST is running, the device continues to operate from the INTOSC selected in OSCCON. When the OST times out, the Fail-Safe condition is cleared after successfully switching to the external clock source. The OSFIF bit should be cleared prior to switching to the external clock source. If the Fail-Safe condition still exists, the OSFIF flag will again become set by hardware.

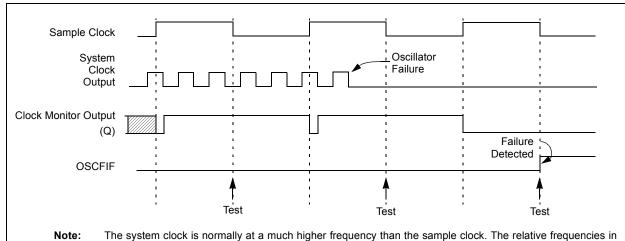
5.5.4 RESET OR WAKE-UP FROM SLEEP

The FSCM is designed to detect an oscillator failure after the Oscillator Start-up Timer (OST) has expired. The OST is used after waking up from Sleep and after any type of Reset. The OST is not used with the EC or RC Clock modes so that the FSCM will be active as soon as the Reset or wake-up has completed. When the FSCM is enabled, the Two-Speed Start-up is also enabled. Therefore, the device will always be executing code while the OST is operating.

Note: Due to the wide range of oscillator start-up times, the Fail-Safe circuit is not active during oscillator start-up (i.e., after exiting Reset or Sleep). After an appropriate amount of time, the user should check the Status bits in the OSCSTAT register to verify the oscillator start-up and that the system clock switchover has successfully completed.



this example have been chosen for clarity.



© 2010-2012 Microchip Technology Inc.

5.6 Oscillator Control Registers

REGISTER 5-1: OSCCON: OSCILLATOR CONTROL REGISTER

R/W-0/0	R/W-0/0	R/W-1/1	R/W-1/1	R/W-1/1	U-0	R/W-0/0	R/W-0/0
SPLLEN		IRCF	<3:0>	_	SCS	<1:0>	
bit 7							bit 0

Legend:R = Readable bitW = Writable bitU = Unimplemented bit, read as '0'u = Bit is unchangedx = Bit is unknown-n/n = Value at POR and BOR/Value at all other Resets'1' = Bit is set'0' = Bit is cleared

bit 7 SPLLEN: Software PLL Enable bit

If PLLEN in Configuration Word 2 = 1:

SPLLEN bit is ignored. 4xPLL is always enabled (subject to oscillator requirements)

If PLLEN in Configuration Word 2 = 0:

1 = 4xPLL Is enabled 0 = 4xPLL is disabled

bit 6-3 IRCF<3:0>: Internal Oscillator Frequency Select bits

000x = 31 kHz LF 0010 = 31.25 kHz MF 0011 = 31.25 kHz HF⁽¹⁾

0100 = 62.5 kHz MF 0101 = 125 kHz MF 0110 = 250 kHz MF

0111 = 500 kHz MF (default upon Reset)

1000 = 125 kHz HF⁽¹⁾ 1001 = 250 kHz HF⁽¹⁾ 1010 = 500 kHz HF⁽¹⁾ 1011 = 1 MHz HF

1100 = 2 MHz HF 1101 = 4 MHz HF

1110 = 8 MHz or 32 MHz HF(see Section 5.2.2.1 "HFINTOSC")

1111 = 16 MHz HF

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

bit 1-0 SCS<1:0>: System Clock Select bits

1x = Internal oscillator block

01 = Timer1 oscillator

00 = Clock determined by FOSC<2:0> in Configuration Word 1.

Note 1: Duplicate frequency derived from HFINTOSC.

REGISTER 5-2: OSCSTAT: OSCILLATOR STATUS REGISTER

R-1/q	R-0/q	R-q/q	R-0/q	R-0/q	R-q/q	R-0/0	R-0/q
T10SCR	PLLR	OSTS	HFIOFR	HFIOFL	MFIOFR	LFIOFR	HFIOFS
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared q = Conditional

bit 7 T10SCR: Timer1 Oscillator Ready bit

<u>If T10SCEN = 1</u>:

1 = Timer1 oscillator is ready

0 = Timer1 oscillator is not ready

If T1OSCEN = 0:

1 = Timer1 clock source is always ready

bit 6 PLLR 4xPLL Ready bit

1 = 4xPLL is ready

0 = 4xPLL is not ready

bit 5 OSTS: Oscillator Start-up Timer Status bit

1 = Running from the clock defined by the FOSC<2:0> bits of the Configuration Word 1

0 = Running from an internal oscillator (FOSC<2:0> = 100)

bit 4 **HFIOFR:** High-Frequency Internal Oscillator Ready bit

1 = HFINTOSC is ready

0 = HFINTOSC is not ready

bit 3 **HFIOFL:** High-Frequency Internal Oscillator Locked bit

1 = HFINTOSC is at least 2% accurate

0 = HFINTOSC is not 2% accurate

bit 2 MFIOFR: Medium-Frequency Internal Oscillator Ready bit

1 = MFINTOSC is ready

0 = MFINTOSC is not ready

bit 1 LFIOFR: Low-Frequency Internal Oscillator Ready bit

1 = LFINTOSC is ready

0 = LFINTOSC is not ready

bit 0 **HFIOFS**: High-Frequency Internal Oscillator Stable bit

1 = HFINTOSC is at least 0.5% accurate

0 = HFINTOSC is not 0.5% accurate

REGISTER 5-3: OSCTUNE: OSCILLATOR TUNING REGISTER

U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0			
_	_		TUN<5:0>							
bit 7							bit 0			

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

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

bit 5-0 **TUN<5:0>:** Frequency Tuning bits

011111 = Maximum frequency

011110 =

•

•

000001 =

000000 = Oscillator module is running at the factory-calibrated frequency.

111111 =

•

•

100000 = Minimum frequency

TABLE 5-2: SUMMARY OF REGISTERS ASSOCIATED WITH CLOCK SOURCES

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page	
OSCCON	SPLLEN		IRCF	<3:0>		_	SCS	70		
OSCSTAT	T10SCR	PLLR	OSTS	HFIOFR	HFIOFL	MFIOFR	LFIOFR	HFIOFS	71	
OSCTUNE	_	_		TUN<5:0>						
PIE2	OSFIE	C2IE	C1IE	EEIE	BCL1IE	_	_	CCP2IE	92	
PIR2	OSFIF	C2IF	C1IF	EEIF	BCL1IF	_	_	CCP2IF	96	
T1CON	TMR1C	S<1:0>	T1CKPS<1:0>		T10SCEN	T1SYNC	_	TMR10N	195	

Legend: — Unimplemented location, read as '0'. Shaded cells are not used by clock sources.

TABLE 5-3: SUMMARY OF CONFIGURATION WORD WITH CLOCK SOURCES

Name	Bits	Bit -/7	Bit -/6	Bit 13/5	Bit 12/4	Bit 11/3	Bit 10/2	Bit 9/1	Bit 8/0	Register on Page
CONFICA	13:8	_	_	FCMEN	IESO	CLKOUTEN	BOREN<1:0>		CPD	50
CONFIG1	7:0	CP	MCLRE	PWRTE	WDTE	E<1:0>	FOSC<2:0>		•	50

Legend: — Unimplemented location, read as '0'. Shaded cells are not used by clock sources.

6.0 REFERENCE CLOCK MODULE

The Reference Clock module provides the ability to send a divided clock to the clock output pin of the device (CLKR) and provide a secondary internal clock source to the modulator module. This module is available in all oscillator configurations and allows the user to select a greater range of clock submultiples to drive external devices in the application. The Reference Clock module includes the following features:

- · System clock is the source
- · Available in all oscillator configurations
- · Programmable clock divider
- · Output enable to a port pin
- · Selectable duty cycle
- · Slew rate control

The Reference Clock module is controlled by the CLKRCON register (Register 6-1) and is enabled when setting the CLKREN bit. To output the divided clock signal to the CLKR port pin, the CLKROE bit must be set. The CLKRDIV<2:0> bits enable the selection of eight different clock divider options. The CLKRDC<1:0> bits can be used to modify the duty cycle of the output clock⁽¹⁾. The CLKRSLR bit controls slew rate limiting.

Note 1: If the base clock rate is selected without a divider, the output clock will always have a duty cycle equal to that of the source clock, unless a 0% duty cycle is selected. If the clock divider is set to base clock/2, then 25% and 75% duty cycle accuracy will be dependent upon the source clock.

For information on using the reference clock output with the modulator module, see **Section 23.0 "Data Signal Modulator"**.

6.1 Slew Rate

The slew rate limitation on the output port pin can be disabled. The slew rate limitation can be removed by clearing the CLKRSLR bit in the CLKRCON register.

6.2 Effects of a Reset

Upon any device Reset, the Reference Clock module is disabled. The user's firmware is responsible for initializing the module before enabling the output. The registers are reset to their default values.

6.3 Conflicts with the CLKR Pin

There are two cases when the reference clock output signal cannot be output to the CLKR pin, if:

- · LP, XT or HS Oscillator mode is selected.
- · CLKOUT function is enabled.

Even if either of these cases are true, the module can still be enabled and the reference clock signal may be used in conjunction with the modulator module.

6.3.1 OSCILLATOR MODES

If LP, XT or HS oscillator modes are selected, the OSC2/CLKR pin must be used as an oscillator input pin and the CLKR output cannot be enabled. See Section 5.2 "Clock Source Types" for more information on different oscillator modes.

6.3.2 CLKOUT FUNCTION

The CLKOUT function has a higher priority than the Reference Clock module. Therefore, if the CLKOUT function is enabled by the CLKOUTEN bit in Configuration Word 1, Fosc/4 will always be output on the port pin. Reference **Section 4.0 "Device Configuration"** for more information.

6.4 Operation During Sleep

As the Reference Clock module relies on the system clock as its source, and the system clock is disabled in Sleep, the module does not function in Sleep, even if an external clock source or the Timer1 clock source is configured as the system clock. The module outputs will remain in their current state until the device exits Sleep.

6.5 Reference Clock Control Register

REGISTER 6-1: CLKRCON: REFERENCE CLOCK CONTROL REGISTER

R/W-0/0	R/W-0/0	R/W-1/1	R/W-1/1	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
CLKREN	CLKROE	CLKRSLR	CLKRD	C<1:0>	(CLKRDIV<2:0>	•
bit 7							bit 0

Legend:R = Readable bitW = Writable bitU = Unimplemented bit, read as '0'u = Bit is unchangedx = Bit is unknown-n/n = Value at POR and BOR/Value at all other Resets'1' = Bit is set'0' = Bit is cleared

bit 7 **CLKREN:** Reference Clock Module Enable bit 1 = Reference Clock module is enabled 0 = Reference Clock module is disabled bit 6 **CLKROE**: Reference Clock Output Enable bit⁽³⁾ 1 = Reference Clock output is enabled on CLKR pin 0 = Reference Clock output disabled on CLKR pin CLKRSLR: Reference Clock Slew Rate Control Limiting Enable bit bit 5 1 = Slew Rate limiting is enabled 0 = Slew Rate limiting is disabled bit 4-3 CLKRDC<1:0>: Reference Clock Duty Cycle bits 11 = Clock outputs duty cycle of 75% 10 = Clock outputs duty cycle of 50% 01 = Clock outputs duty cycle of 25% 00 = Clock outputs duty cycle of 0% bit 2-0 CLKRDIV<2:0> Reference Clock Divider bits 111 = Base clock value divided by 128 110 = Base clock value divided by 64 101 = Base clock value divided by 32 100 = Base clock value divided by 16 011 = Base clock value divided by 8 010 = Base clock value divided by 4

001 = Base clock value divided by 2⁽¹⁾

000 = Base clock value⁽²⁾

Note 1: In this mode, the 25% and 75% duty cycle accuracy will be dependent on the source clock duty cycle.

- 2: In this mode, the duty cycle will always be equal to the source clock duty cycle, unless a duty cycle of 0% is selected.
- 3: To route CLKR to pin, CLKOUTEN of Configuration Word 1 = 1 is required. CLKOUTEN of Configuration Word 1 = 0 will result in Fosc/4. See Section 6.3 "Conflicts with the CLKR Pin" for details.

TABLE 6-1: SUMMARY OF REGISTERS ASSOCIATED WITH REFERENCE CLOCK SOURCES

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
CLKRCON	CLKREN	CLKROE	CLKRSLR	CLKRDC<1:0>		:0> CLKRDIV<2:0>			

Legend: — Unimplemented locations read as '0'. Shaded cells are not used by reference clock sources.

TABLE 6-2: SUMMARY OF CONFIGURATION WORD WITH REFERENCE CLOCK SOURCES

Name	Bits	Bit -/7	Bit -/6	Bit 13/5	Bit 12/4	Bit 11/3	Bit 10/2	Bit 9/1	Bit 8/0	Register on Page
CONFIG1	13:8	I	_	FCMEN	IESO	IESO CLKOUTEN		EN BOREN<1:0>		50
CONFIGI	7:0	CP	MCLRE	PWRTE	WDTE<1:0>			FOSC<2:0>		50

Legend: — Unimplemented locations read as '0'. Shaded cells are not used by reference clock sources.

NOTES:

7.0 RESETS

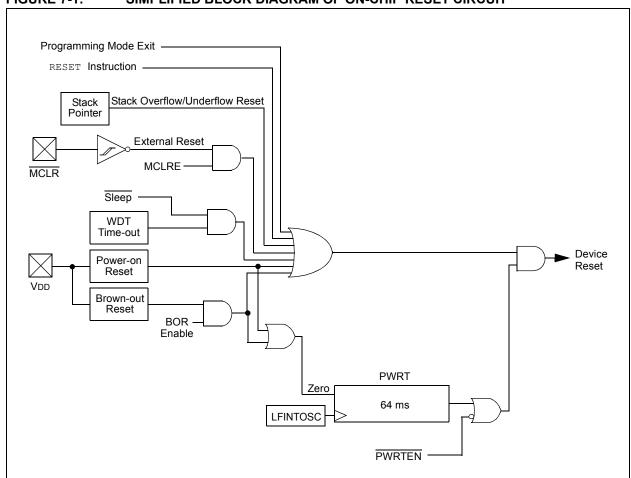
There are multiple ways to reset this device:

- Power-on Reset (POR)
- · Brown-out Reset (BOR)
- MCLR Reset
- · WDT Reset
- RESET instruction
- · Stack Overflow
- · Stack Underflow
- · Programming mode exit

To allow VDD to stabilize, an optional power-up timer can be enabled to extend the Reset time after a BOR or POR event.

A simplified block diagram of the On-Chip Reset Circuit is shown in Figure 7-1.

FIGURE 7-1: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT



7.1 Power-on Reset (POR)

The POR circuit holds the device in Reset until VDD has reached an acceptable level for minimum operation. Slow rising VDD, fast operating speeds or analog performance may require greater than minimum VDD. The PWRT, BOR or MCLR features can be used to extend the start-up period until all device operation conditions have been met.

7.1.1 POWER-UP TIMER (PWRT)

The Power-up Timer provides a nominal 64 ms timeout on POR or Brown-out Reset.

The device is held in Reset as long as PWRT is active. The PWRT delay allows additional time for the VDD to rise to an acceptable level. The Power-up Timer is enabled by clearing the PWRTE bit in Configuration Word 1.

The Power-up Timer starts after the release of the POR and BOR.

For additional information, refer to Application Note AN607, "Power-up Trouble Shooting" (DS00607).

7.2 Brown-Out Reset (BOR)

The BOR circuit holds the device in Reset when VDD reaches a selectable minimum level. Between the POR and BOR, complete voltage range coverage for execution protection can be implemented.

The Brown-out Reset module has four operating modes controlled by the BOREN<1:0> bits in Configuration Word 1. The four operating modes are:

- · BOR is always on
- · BOR is off when in Sleep
- · BOR is controlled by software
- · BOR is always off

Refer to Table 7-2 for more information.

The Brown-out Reset voltage level is selectable by configuring the BORV bit in Configuration Word 2.

A VDD noise rejection filter prevents the BOR from triggering on small events. If VDD falls below VBOR for a duration greater than parameter TBORDC, the device will reset. See Figure 7-2 for more information.

TABLE 7-1: BOR OPERATING MODES

BOREN<1:0>	SBOREN	Device Mode	BOR Mode	Device Operation upon Release of POR	Device Operation upon Wake-up from Sleep	
11	Х	Х	X Active Waits for B		OR ready ⁽¹⁾	
1.0			Active	Waits for BOR ready		
10	X	Sleep	Disabled	vvaits ioi	BOR leady	
0.1	1	Х	Active	Begins immediately		
01	0	X	Disabled	Begins immediately		
00	Х	X	Disabled	Begins in	nmediately	

Note 1: In these specific cases, "Release of POR" and the "Wake-up from Sleep", there is no delay in start-up. The BOR Ready flag (BORRDY = 1) will be set before the CPU is ready to execute instructions because the BOR circuit is forced on by the BOREN<1:0> bits.

7.2.1 BOR IS ALWAYS ON

When the BOREN bits of Configuration Word 1 are set to '11', the BOR is always on. The device start-up will be delayed until the BOR is ready and VDD is higher than the BOR threshold.

BOR protection is active during Sleep. The BOR does not delay wake-up from Sleep.

7.2.2 BOR IS OFF IN SLEEP

When the BOREN bits of Configuration Word 1 are set to '10', the BOR is on, except in Sleep. The device start-up will be delayed until the BOR is ready and VDD is higher than the BOR threshold.

BOR protection is not active during Sleep. The device wake-up will be delayed until the BOR is ready.

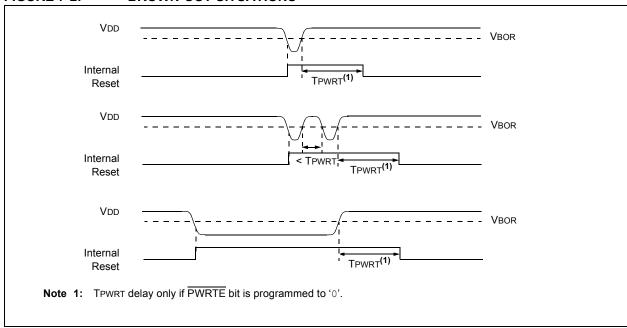
7.2.3 BOR CONTROLLED BY SOFTWARE

When the BOREN bits of Configuration Word 1 are set to '01', the BOR is controlled by the SBOREN bit of the BORCON register. The device start-up is not delayed by the BOR ready condition or the VDD level.

BOR protection begins as soon as the BOR circuit is ready. The status of the BOR circuit is reflected in the BORRDY bit of the BORCON register.

BOR protection is unchanged by Sleep.

FIGURE 7-2: BROWN-OUT SITUATIONS



REGISTER 7-1: BORCON: BROWN-OUT RESET CONTROL REGISTER

R/W-1/u	U-0	U-0	U-0	U-0	U-0	U-0	R-q/u
SBOREN	_	_	_	_	_	_	BORRDY
bit 7	•	•	•	•	•	•	bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	q = Value depends on condition

bit 7 SBOREN: Software Brown-out Reset Enable bit

If BOREN <1:0> in Configuration Word 1 ≠ 01:

SBOREN is read/write, but has no effect on the BOR.

If BOREN <1:0> in Configuration Word 1 = 01:

1 = BOR Enabled

0 = BOR Disabled

bit 6-1 **Unimplemented:** Read as '0'

bit 0 BORRDY: Brown-out Reset Circuit Ready Status bit

1 = The Brown-out Reset circuit is active

0 = The Brown-out Reset circuit is inactive

7.3 MCLR

The MCLR is an optional external input that can reset the device. The MCLR function is controlled by the MCLRE bit of Configuration Word 1 and the LVP bit of Configuration Word 2 (Table 7-2).

TABLE 7-2: MCLR CONFIGURATION

MCLRE	LVP	MCLR
0	0	Disabled
1	0	Enabled
Х	1	Enabled

7.3.1 MCLR ENABLED

When $\overline{\text{MCLR}}$ is enabled and the pin is held low, the device is held in Reset. The $\overline{\text{MCLR}}$ pin is connected to VDD through an internal weak pull-up.

The device has a noise filter in the MCLR Reset path. The filter will detect and ignore small pulses.

7.3.2 MCLR DISABLED

When MCLR is disabled, the pin functions as a general purpose input and the internal weak pull-up is under software control. See Section 12.2 "PORTA Registers" for more information.

7.4 Watchdog Timer (WDT) Reset

The Watchdog Timer generates a Reset if the firmware does not issue a CLRWDT instruction within the time-out period. The \overline{TO} and \overline{PD} bits in the STATUS register are changed to indicate the WDT Reset. See **Section 10.0** "Watchdog Timer" for more information.

7.5 RESET Instruction

A RESET instruction will cause a device Reset. The $\overline{\text{RI}}$ bit in the PCON register will be set to '0'. See Table 7-4 for default conditions after a RESET instruction has occurred.

7.6 Stack Overflow/Underflow Reset

The device can reset when the Stack Overflows or Underflows. The STKOVF or STKUNF bits of the PCON register indicate the Reset condition. These Resets are enabled by setting the STVREN bit in Configuration Word 2. See Section 3.4.2 "Overflow/Underflow Reset" for more information.

7.7 Programming Mode Exit

Upon exit of Programming mode, the device will behave as if a POR had just occurred.

7.8 Power-Up Timer

The Power-up Timer optionally delays device execution after a BOR or POR event. This timer is typically used to allow VDD to stabilize before allowing the device to start running.

The Power-up Timer is controlled by the PWRTE bit of Configuration Word 1.

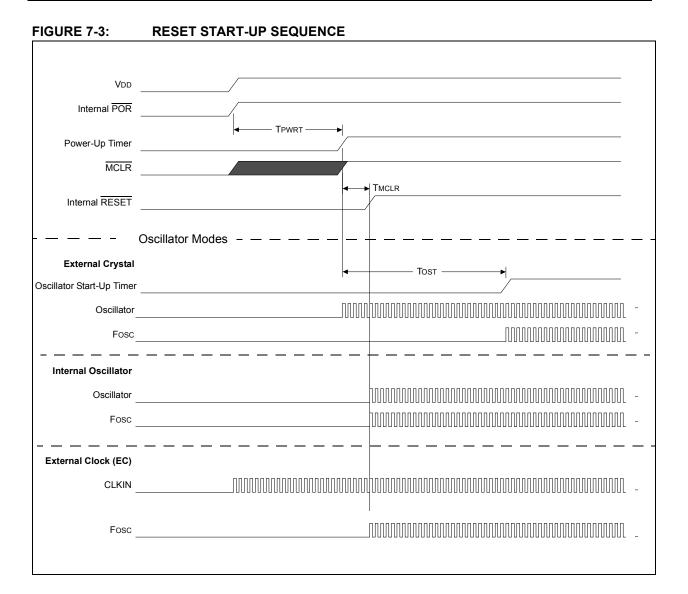
7.9 Start-up Sequence

Upon the release of a POR or BOR, the following must occur before the device will begin executing:

- Power-up Timer runs to completion (if enabled).
- Oscillator start-up timer runs to completion (if required for oscillator source).
- 3. MCLR must be released (if enabled).

The total time-out will vary based on oscillator configuration and Power-up Timer configuration. See Section 5.0 "Oscillator Module (With Fail-Safe Clock Monitor)" for more information.

The Power-up Timer and oscillator start-up timer run independently of MCLR Reset. If MCLR is kept low long enough, the Power-up Timer and oscillator start-up timer will expire. Upon bringing MCLR high, the device will begin execution immediately (see Figure 7-3). This is useful for testing purposes or to synchronize more than one device operating in parallel.



7.10 Determining the Cause of a Reset

Upon any Reset, multiple bits in the STATUS and PCON register are updated to indicate the cause of the Reset. Table 7-3 and Table 7-4 show the Reset conditions of these registers.

TABLE 7-3: RESET STATUS BITS AND THEIR SIGNIFICANCE

STKOVF	STKUNF	RMCLR	RI	POR	BOR	TO	PD	Condition
0	0	1	1	0	Х	1	1	Power-on Reset
0	0	1	1	0	Х	0	Х	Illegal, TO is set on POR
0	0	1	1	0	Х	Х	0	Illegal, PD is set on POR
0	0	1	1	u	0	1	1	Brown-out Reset
u	u	u	u	u	u	0	u	WDT Reset
u	u	u	u	u	u	0	0	WDT Wake-up from Sleep
u	u	u	u	u	u	1	0	Interrupt Wake-up from Sleep
u	u	0	u	u	u	u	u	MCLR Reset during normal operation
u	u	0	u	u	u	1	0	MCLR Reset during Sleep
u	u	u	0	u	u	u	u	RESET Instruction Executed
1	u	u	u	u	u	u	u	Stack Overflow Reset (STVREN = 1)
u	1	u	u	u	u	u	u	Stack Underflow Reset (STVREN = 1)

TABLE 7-4: RESET CONDITION FOR SPECIAL REGISTERS⁽²⁾

Condition	Program Counter	STATUS Register	PCON Register
Power-on Reset	0000h	1 1000	00 110x
MCLR Reset during normal operation	0000h	u uuuu	uu Ouuu
MCLR Reset during Sleep	0000h	1 Ouuu	uu Ouuu
WDT Reset	0000h	0 uuuu	uu uuuu
WDT Wake-up from Sleep	PC + 1	0 Ouuu	uu uuuu
Brown-out Reset	0000h	1 1uuu	00 11u0
Interrupt Wake-up from Sleep	PC + 1 ⁽¹⁾	1 Ouuu	uu uuuu
RESET Instruction Executed	0000h	u uuuu	uu u0uu
Stack Overflow Reset (STVREN = 1)	0000h	u uuuu	1u uuuu
Stack Underflow Reset (STVREN = 1)	0000h	u uuuu	u1 uuuu

Legend: u = unchanged, x = unknown, - = unimplemented bit, reads as '0'.

Note 1: When the wake-up is due to an interrupt and Global Enable bit (GIE) is set, the return address is pushed on the stack and PC is loaded with the interrupt vector (0004h) after execution of PC + 1.

2: If a Status bit is not implemented, that bit will be read as '0'.

7.11 Power Control (PCON) Register

The Power Control (PCON) register contains flag bits to differentiate between a:

- Power-on Reset (POR)
- Brown-out Reset (BOR)
- Reset Instruction Reset (RI)
- Stack Overflow Reset (STKOVF)
- Stack Underflow Reset (STKUNF)
- MCLR Reset (RMCLR)

The PCON register bits are shown in Register 7-2.

REGISTER 7-2: PCON: POWER CONTROL REGISTER

R/W/HS-0/q	R/W/HS-0/q	U-0	U-0	R/W/HC-1/q	R/W/HC-1/q	R/W/HC-q/u	R/W/HC-q/u
STKOVF	STKUNF	-	-	RMCLR	RI	POR	BOR
bit 7							bit 0

Legend:		
HC = Bit is cleared by ha	rdware	HS = Bit is set by hardware
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-m/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	q = Value depends on condition

bit 7	STKOVF: Stack Overflow Flag bit
	1 = A Stack Overflow occurred
	0 = A Stack Overflow has not occurred or set to '0' by firmware
bit 6	STKUNF: Stack Underflow Flag bit
	1 = A Stack Underflow occurred
	0 = A Stack Underflow has not occurred or set to '0' by firmware
bit 5-4	Unimplemented: Read as '0'
bit 3	RMCLR: MCLR Reset Flag bit
	1 = A MCLR Reset has not occurred or set to '1' by firmware
	0 = A MCLR Reset has occurred (set to '0' in hardware when a MCLR Reset occurs)
bit 2	RI: RESET Instruction Flag bit
	1 = A RESET instruction has not been executed or set to '1' by firmware
	0 = A RESET instruction has been executed (set to '0' in hardware upon executing a RESET instruction)
bit 1	POR: Power-on Reset Status bit
	1 = No Power-on Reset occurred
	0 = A Power-on Reset occurred (must be set in software after a Power-on Reset occurs)
bit 0	BOR: Brown-out Reset Status bit
	1 = No Brown-out Reset occurred
	0 = A Brown-out Reset occurred (must be set in software after a Power-on Reset or Brown-out Reset occurs)

TABLE 7-5: SUMMARY OF REGISTERS ASSOCIATED WITH RESETS⁽¹⁾

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
BORCON	SBOREN	_		_	_	_	_	BORRDY	79
PCON	STKOVF	STKUNF	-	_	RMCLR	RI	POR	BOR	83
STATUS	_	1	1	TO	PD	Z	DC	С	24
WDTCON	_	_	WDTPS<4:0>					SWDTEN	103

Legend: — Unimplemented bit, reads as '0'. Shaded cells are not used by Resets.

Note 1: Other (non Power-up) Resets include MCLR Reset and Watchdog Timer Reset during normal operation.

8.0 INTERRUPTS

The interrupt feature allows certain events to preempt normal program flow. Firmware is used to determine the source of the interrupt and act accordingly. Some interrupts can be configured to wake the MCU from Sleep mode.

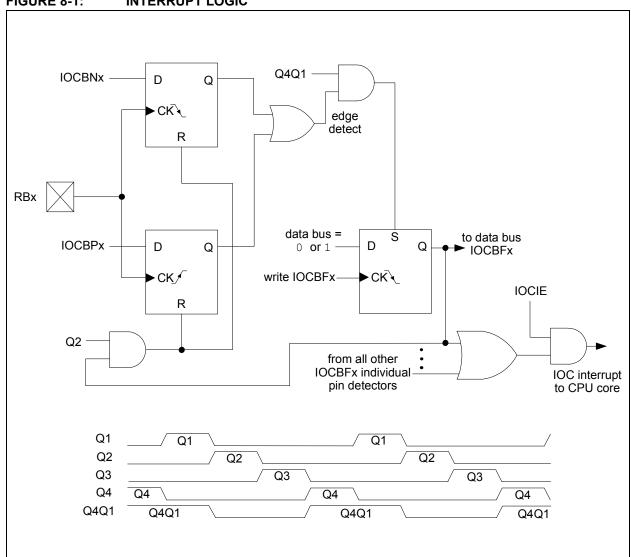
This chapter contains the following information for Interrupts:

- Operation
- · Interrupt Latency
- · Interrupts During Sleep
- INT Pin
- · Automatic Context Saving

Many peripherals produce Interrupts. Refer to the corresponding chapters for details.

A block diagram of the interrupt logic is shown in Figure 8-1.

FIGURE 8-1: INTERRUPT LOGIC



8.1 Operation

Interrupts are disabled upon any device Reset. They are enabled by setting the following bits:

- · GIE bit of the INTCON register
- Interrupt Enable bit(s) for the specific interrupt event(s)
- PEIE bit of the INTCON register (if the Interrupt Enable bit of the interrupt event is contained in the PIEx register)

The INTCON, PIR1, PIR2, PIR3 and PIR4 registers record individual interrupts via interrupt flag bits. Interrupt flag bits will be set, regardless of the status of the GIE, PEIE and individual interrupt enable bits.

The following events happen when an interrupt event occurs while the GIE bit is set:

- · Current prefetched instruction is flushed
- · GIE bit is cleared
- Current Program Counter (PC) is pushed onto the stack
- Critical registers are automatically saved to the shadow registers (See Section 8.5 "Automatic Context Saving")
- · PC is loaded with the interrupt vector 0004h

The firmware within the Interrupt Service Routine (ISR) should determine the source of the interrupt by polling the interrupt flag bits. The interrupt flag bits must be cleared before exiting the ISR to avoid repeated interrupts. Because the GIE bit is cleared, any interrupt that occurs while executing the ISR will be recorded through its interrupt flag, but will not cause the processor to redirect to the interrupt vector.

The RETFIE instruction exits the ISR by popping the previous address from the stack, restoring the saved context from the shadow registers and setting the GIE bit.

For additional information on a specific interrupt's operation, refer to its peripheral chapter.

- Note 1: Individual interrupt flag bits are set, regardless of the state of any other enable bits.
 - 2: All interrupts will be ignored while the GIE bit is cleared. Any interrupt occurring while the GIE bit is clear will be serviced when the GIE bit is set again.

8.2 Interrupt Latency

Interrupt latency is defined as the time from when the interrupt event occurs to the time code execution at the interrupt vector begins. The latency for synchronous interrupts is three or four instruction cycles. For asynchronous interrupts, the latency is three to five instruction cycles, depending on when the interrupt occurs. See Figure 8-2 and Figure 8.3 for more details.

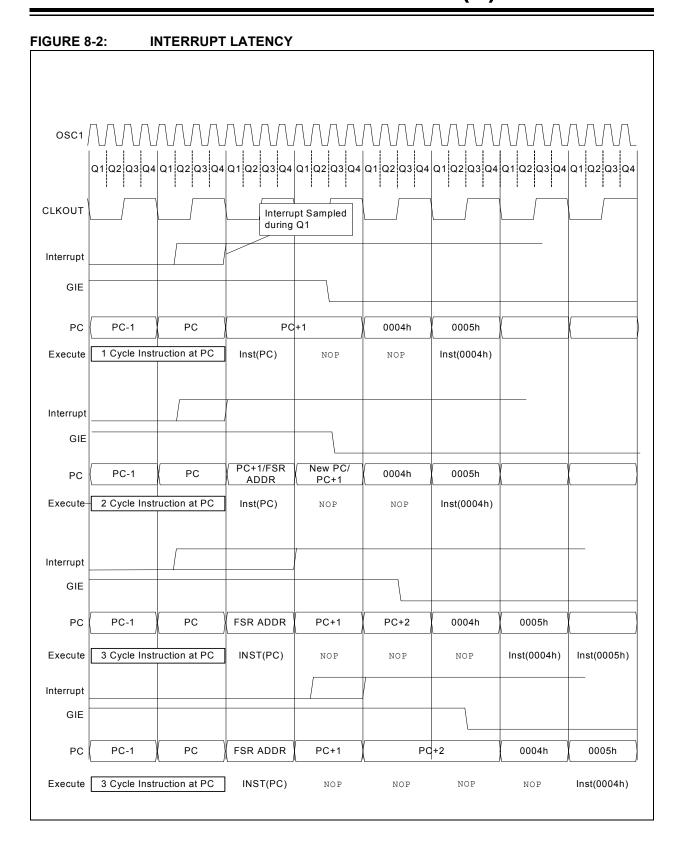
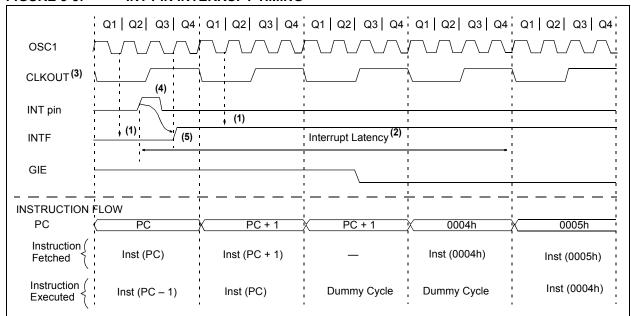


FIGURE 8-3: INT PIN INTERRUPT TIMING



- Note 1: INTF flag is sampled here (every Q1).
 - 2: Asynchronous interrupt latency = 3-5 TcY. Synchronous latency = 3-4 TcY, where TcY = instruction cycle time. Latency is the same whether Inst (PC) is a single cycle or a 2-cycle instruction.
 - 3: CLKOUT not available in all oscillator modes.
 - 4: For minimum width of INT pulse, refer to AC specifications in Section 30.0 "Electrical Specifications".
 - 5: INTF is enabled to be set any time during the Q4-Q1 cycles.

8.3 Interrupts During Sleep

Some interrupts can be used to wake from Sleep. To wake from Sleep, the peripheral must be able to operate without the system clock. The interrupt source must have the appropriate Interrupt Enable bit(s) set prior to entering Sleep.

On waking from Sleep, if the GIE bit is also set, the processor will branch to the interrupt vector. Otherwise, the processor will continue executing instructions after the SLEEP instruction. The instruction directly after the SLEEP instruction will always be executed before branching to the ISR. Refer to the Section 9.0 "Power-Down Mode (Sleep)" for more details.

8.4 INT Pin

The INT pin can be used to generate an asynchronous edge-triggered interrupt. This interrupt is enabled by setting the INTE bit of the INTCON register. The INTEDG bit of the OPTION_REG register determines on which edge the interrupt will occur. When the INTEDG bit is set, the rising edge will cause the interrupt. When the INTEDG bit is clear, the falling edge will cause the interrupt. The INTF bit of the INTCON register will be set when a valid edge appears on the INT pin. If the GIE and INTE bits are also set, the processor will redirect program execution to the interrupt vector.

8.5 Automatic Context Saving

Upon entering an interrupt, the return PC address is saved on the stack. Additionally, the following registers are automatically saved in the shadow registers:

- · W register
- STATUS register (except for TO and PD)
- · BSR register
- · FSR registers
- · PCLATH register

Upon exiting the Interrupt Service Routine, these registers are automatically restored. Any modifications to these registers during the ISR will be lost. If modifications to any of these registers are desired, the corresponding shadow register should be modified and the value will be restored when exiting the ISR. The shadow registers are available in Bank 31 and are readable and writable. Depending on the user's application, other registers may also need to be saved.

8.6 Interrupt Control Registers

8.6.1 INTCON REGISTER

The INTCON register is a readable and writable register, that contains the various enable and flag bits for TMR0 register overflow, interrupt-on-change and external INT pin interrupts.

Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the Global Enable bit, GIE, of the INTCON register. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 8-1: INTCON: INTERRUPT CONTROL REGISTER

R/W-0/0	R-0/0						
GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF ⁽¹⁾
bit 7							bit 0

Note:

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7	GIE: Global Interrupt Enable bit
	1 = Enables all active interrupts0 = Disables all interrupts
bit 6	PEIE: Peripheral Interrupt Enable bit 1 = Enables all active peripheral interrupts 0 = Disables all peripheral interrupts
bit 5	TMR0IE: Timer0 Overflow Interrupt Enable bit 1 = Enables the Timer0 interrupt 0 = Disables the Timer0 interrupt
bit 4	INTE: INT External Interrupt Enable bit 1 = Enables the INT external interrupt 0 = Disables the INT external interrupt
bit 3	IOCIE: Interrupt-on-Change Enable bit 1 = Enables the interrupt-on-change 0 = Disables the interrupt-on-change
bit 2	TMR0IF: Timer0 Overflow Interrupt Flag bit 1 = TMR0 register has overflowed 0 = TMR0 register did not overflow
bit 1	INTF: INT External Interrupt Flag bit 1 = The INT external interrupt occurred 0 = The INT external interrupt did not occur
bit 0	IOCIF: Interrupt-on-Change Interrupt Flag bit ⁽¹⁾ 1 = When at least one of the interrupt-on-change pins changed state 0 = None of the interrupt-on-change pins have changed state

Note 1: The IOCIF Flag bit is read-only and cleared when all the Interrupt-on-Change flags in the IOCxF register have been cleared by software.

8.6.2 PIE1 REGISTER

The PIE1 register contains the interrupt enable bits, as shown in Register 8-2.

Note: Bit PEIE of the INTCON register must be set to enable any peripheral interrupt.

REGISTER 8-2: PIE1: PERIPHERAL INTERRUPT ENABLE REGISTER 1

| R/W-0/0 |
|---------|---------|---------|---------|---------|---------|---------|---------|
| TMR1GIE | ADIE | RCIE | TXIE | SSP1IE | CCP1IE | TMR2IE | TMR1IE |
| bit 7 | | | | | | | bit 0 |

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7	TMR1GIE: Timer1 Gate Interrupt Enable bit
	1 = Enables the Timer1 Gate Acquisition interrupt
	0 = Disables the Timer1 Gate Acquisition interrupt
bit 6	ADIE: A/D Converter (ADC) Interrupt Enable bit
	1 = Enables the ADC interrupt
	0 = Disables the ADC interrupt
bit 5	RCIE: USART Receive Interrupt Enable bit
	1 = Enables the USART receive interrupt
	0 = Disables the USART receive interrupt
bit 4	TXIE: USART Transmit Interrupt Enable bit
	1 = Enables the USART transmit interrupt
	0 = Disables the USART transmit interrupt
bit 3	SSP1IE: Synchronous Serial Port (MSSP) Interrupt Enable bit
	1 = Enables the MSSP interrupt
	0 = Disables the MSSP interrupt
bit 2	CCP1IE: CCP1 Interrupt Enable bit
	1 = Enables the CCP1 interrupt
	0 = Disables the CCP1 interrupt
bit 1	TMR2IE: TMR2 to PR2 Match Interrupt Enable bit
	1 = Enables the Timer2 to PR2 match interrupt
	0 = Disables the Timer2 to PR2 match interrupt
bit 0	TMR1IE: Timer1 Overflow Interrupt Enable bit
	1 = Enables the Timer1 overflow interrupt0 = Disables the Timer1 overflow interrupt

8.6.3 PIE2 REGISTER

The PIE2 register contains the interrupt enable bits, as shown in Register 8-3.

Note: Bit PEIE of the INTCON register must be set to enable any peripheral interrupt.

REGISTER 8-3: PIE2: PERIPHERAL INTERRUPT ENABLE REGISTER 2

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	U-0	R/W-0/0
OSFIE	C2IE ⁽¹⁾	C1IE	EEIE	BCL1IE	_	_	CCP2IE
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7	OSFIE: Oscillator Fail Interrupt Enable bit
	1 = Enables the Oscillator Fail interrupt
	0 = Disables the Oscillator Fail interrupt
bit 6	C2IE: Comparator C2 Interrupt Enable bit ⁽¹⁾
	1 = Enables the Comparator C2 interrupt
	0 = Disables the Comparator C2 interrupt
bit 5	C1IE: Comparator C1 Interrupt Enable bit
	1 = Enables the Comparator C1 interrupt
	0 = Disables the Comparator C1 interrupt
bit 4	EEIE: EEPROM Write Completion Interrupt Enable bit
	1 = Enables the EEPROM write completion interrupt
	0 = Disables the EEPROM write completion interrupt
bit 3	BCL1IE: MSSP Bus Collision Interrupt Enable bit
	1 = Enables the MSSP bus collision interrupt
	0 = Disables the MSSP bus collision interrupt
bit 2-1	Unimplemented: Read as '0'
bit 0	CCP2IE: CCP2 Interrupt Enable bit
	1 = Enables the CCP2 interrupt
	0 = Disables the CCP2 interrupt

Note 1: PIC16(L)F1829 only.

8.6.4 PIE3 REGISTER

The PIE3 register contains the interrupt enable bits, as shown in Register 8-4.

Note 1: Bit PEIE of the INTCON register must be set to enable any peripheral interrupt.

REGISTER 8-4: PIE3: PERIPHERAL INTERRUPT ENABLE REGISTER 3

U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	R/W-0/0	U-0
_	_	CCP4IE	CCP3IE	TMR6IE	_	TMR4IE	_
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7-6	Unimplemented: Read as '0'
bit 5	CCP4IE: CCP4 Interrupt Enable bit
	1 = Enables the CCP4 interrupt0 = Disables the CCP4 interrupt
bit 4	CCP3IE: CCP3 Interrupt Enable bit
	1 = Enables the CCP3 interrupt0 = Disables the CCP3 interrupt
bit 3	TMR6IE: TMR6 to PR6 Match Interrupt Enable bit 1 = Enables the TMR6 to PR6 match interrupt 0 = Disables the TMR6 to PR6 match interrupt
bit 2	Unimplemented: Read as '0'
bit 1	TMR4IE: TMR4 to PR4 Match Interrupt Enable bit
	1 = Enables the TMR4 to PR4 match interrupt0 = Disables the TMR4 to PR4 match interrupt
bit 0	Unimplemented: Read as '0'

8.6.5 PIE4 REGISTER⁽¹⁾

The PIE4 register contains the interrupt enable bits, as shown in Register 8-5.

- **Note 1:** The PIE4 register is available only on the PIC16(L)F1829 device.
 - **2:** Bit PEIE of the INTCON register must be set to enable any peripheral interrupt.

REGISTER 8-5: PIE4: PERIPHERAL INTERRUPT ENABLE REGISTER 4⁽¹⁾

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0/0	R/W-0/0
_	_	_	_	_	_	BCL2IE	SSP2IE
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7-2 **Unimplemented:** Read as '0'

bit 1 BCL2IE: MSSP2 Bus Collision Interrupt Enable bit

1 = Enables the MSSP2 bus collision interrupt0 = Disables the MSSP2 bus collision interrupt

bit 0 SSP2IE: Master Synchronous Serial Port 2 (MSSP2) Interrupt Enable bit

1 = Enables the MSSP2 interrupt0 = Disables the MSSP2 interrupt

Note 1: This register is only available on PIC16(L)F1829.

8.6.6 PIR1 REGISTER

Legend:

R = Readable bit

'1' = Bit is set

u = Bit is unchanged

The PIR1 register contains the interrupt flag bits, as shown in Register 8-6.

Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the Global Enable bit, GIE, of the INTCON register. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 8-6: PIR1: PERIPHERAL INTERRUPT REQUEST REGISTER 1

R/W-0/0	R/W-0/0	R-0/0	R-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF
bit 7							bit 0

Note:

U = Unimplemented bit, read as '0'

-n/n = Value at POR and BOR/Value at all other Resets

bit 7	TMR1GIF: Timer1 Gate Interrupt Flag bit
	1 = Interrupt is pending
	0 = Interrupt is not pending
bit 6	ADIF: A/D Converter Interrupt Flag bit
	1 = Interrupt is pending
	0 = Interrupt is not pending
bit 5	RCIF: USART Receive Interrupt Flag bit
	1 = Interrupt is pending
	0 = Interrupt is not pending
bit 4	TXIF: USART Transmit Interrupt Flag bit
	1 = Interrupt is pending
	0 = Interrupt is not pending
bit 3	SSP1IF: Synchronous Serial Port (MSSP) Interrupt Flag bit
	1 = Interrupt is pending
	0 = Interrupt is not pending
bit 2	CCP1IF: CCP1 Interrupt Flag bit
	1 = Interrupt is pending
	0 = Interrupt is not pending
bit 1	TMR2IF: Timer2 to PR2 Interrupt Flag bit
	1 = Interrupt is pending
	0 = Interrupt is not pending
bit 0	TMR1IF: Timer1 Overflow Interrupt Flag bit

W = Writable bit

x = Bit is unknown

'0' = Bit is cleared

1 = Interrupt is pending0 = Interrupt is not pending

8.6.7 PIR2 REGISTER

The PIR2 register contains the interrupt flag bits, as shown in Register 8-7.

Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the Global Enable bit, GIE, of the INTCON register. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 8-7: PIR2: PERIPHERAL INTERRUPT REQUEST REGISTER 2

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	U-0	U-0
OSFIF	C2IF ⁽¹⁾	C1IF	EEIF	BCL1IF	_	_	CCP2IF
bit 7							bit 0

Note:

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7	OSFIF: Oscillator Fail Interrupt Flag bit
	1 = Interrupt is pending
	0 = Interrupt is not pending
bit 6	C2IF: Comparator C2 Interrupt Flag bit ⁽¹⁾
	1 = Interrupt is pending
	0 = Interrupt is not pending
bit 5	C1IF: Comparator C1 Interrupt Flag bit
	1 = Interrupt is pending
	0 = Interrupt is not pending
bit 4	EEIF: EEPROM Write Completion Interrupt Flag bit
	1 = Interrupt is pending
	0 = Interrupt is not pending
bit 3	BCL1IF: MSSP Bus Collision Interrupt Flag bit
	1 = Interrupt is pending
	0 = Interrupt is not pending
bit 2-1	CCP2IF: CCP2 Interrupt Flag bit
	1 = Interrupt is pending
	0 = Interrupt is not pending
bit 0	Unimplemented: Read as '0'

Note 1: PIC16(L)F1829 only.

8.6.8 PIR3 REGISTER

The PIR3 register contains the interrupt flag bits, as shown in Register 8-8.

Note 1: Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the Global Enable bit, GIE, of the INTCON register. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 8-8: PIR3: PERIPHERAL INTERRUPT REQUEST REGISTER 3

U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	R/W-0/0	U-0
_	_	CCP4IF	CCP3IF	TMR6IF	_	TMR4IF	_
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7-6	Unimplemented: Read as '0'
bit 5	CCP4IF: CCP4 Interrupt Flag bit
	1 = Interrupt is pending0 = Interrupt is not pending
bit 4	CCP3IF: CCP3 Interrupt Flag bit
	1 = Interrupt is pending
	0 = Interrupt is not pending
bit 3	TMR6IF: TMR6 to PR6 Match Interrupt Flag bit
	1 = Interrupt is pending
	0 = Interrupt is not pending
bit 2	Unimplemented: Read as '0'
bit 1	TMR4IF: TMR4 to PR4 Match Interrupt Flag bit
	1 = Interrupt is pending
	0 = Interrupt is not pending
bit 0	Unimplemented: Read as '0'

8.6.9 PIR4 REGISTER⁽¹⁾

The PIR4 register contains the interrupt flag bits, as shown in Register 8-9.

- **Note 1:** The PIR4 register is available only on the PIC16(L)F1829 device.
 - 2: Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the Global Enable bit, GIE, of the INTCON register. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 8-9: PIR4: PERIPHERAL INTERRUPT REQUEST REGISTER 4⁽¹⁾

U-0	U-0	U-0	U-0	U-0	U-0	R/W/HS-0/0	R/W/HS-0/0
_	_	_	_	_	_	BCL2IF	SSP2IF
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared HS = Bit is set by hardware

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

bit 1 BCL2IF: MSSP2 Bus Collision Interrupt Flag bit

1 = A Bus Collision was detected (must be cleared in software)

0 = No Bus collision was detected

bit 0 SSP2IF: Master Synchronous Serial Port 2 (MSSP2) Interrupt Flag bit

1 = The Transmission/Reception/Bus Condition is complete (must be cleared in software)

0 = Waiting to Transmit/Receive/Bus Condition in progress

Note 1: This register is only available on PIC16(L)F1829.

TABLE 8-1: SUMMARY OF REGISTERS ASSOCIATED WITH INTERRUPTS

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	90
OPTION_REG	WPUEN	INTEDG	TMR0CS	TMR0SE	PSA		PS<2:0>		
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	91
PIE2	OSFIE	C2IE	C1IE	EEIE	BCL1IE	_	1	CCP2IE	92
PIE3	_	_	CCP4IE	CCP3IE	TMR6IE	_	TMR4IE	_	93
PIE4 ⁽¹⁾	_	_	_	_	_	_	BCL2IE	SSP2IE	94
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	95
PIR2	OSFIF	C2IF	C1IF	EEIF	BCL1IF	_	_	CCP2IF	96
PIR3	_	_	CCP4IF	CCP3IF	TMR6IF	_	TMR4IF	_	97
PIR4 ⁽¹⁾	_	_	1	1	- 1	_	BCL2IF	SSP2IF	98

Legend: — Unimplemented locations read as '0'. Shaded cells are not used by interrupts.

Note 1: PIC16(L)F1829 only.

9.0 POWER-DOWN MODE (SLEEP)

The Power-Down mode is entered by executing a SLEEP instruction.

Upon entering Sleep mode, the following conditions exist:

- WDT will be cleared but keeps running, if enabled for operation during Sleep.
- PD bit of the STATUS register is cleared.
- 3. TO bit of the STATUS register is set.
- 4. CPU clock is disabled.
- 31 kHz LFINTOSC is unaffected and peripherals that operate from it may continue operation in Sleep.
- Timer1 oscillator is unaffected and peripherals that operate from it may continue operation in Sleep.
- ADC is unaffected, if the dedicated FRC clock is selected.
- 8. Capacitive Sensing oscillator is unaffected.
- I/O ports maintain the status they had before SLEEP was executed (driving high, low or high-impedance).
- 10. Resets other than WDT are not affected by Sleep mode.

Refer to individual chapters for more details on peripheral operation during Sleep.

To minimize current consumption, the following conditions should be considered:

- · I/O pins should not be floating
- · External circuitry sinking current from I/O pins
- Internal circuitry sourcing current from I/O pins
- · Current draw from pins with internal weak pull-ups
- · Modules using 31 kHz LFINTOSC
- · Modules using Timer1 oscillator

I/O pins that are high-impedance inputs should be pulled to VDD or Vss externally to avoid switching currents caused by floating inputs.

Examples of internal circuitry that might be sourcing current include modules such as the DAC and FVR modules. See Section 17.0 "Digital-to-Analog Converter (DAC) Module" and Section 14.0 "Fixed Voltage Reference (FVR)" for more information on these modules.

9.1 Wake-up from Sleep

The device can wake-up from Sleep through one of the following events:

- 1. External Reset input on MCLR pin, if enabled
- 2. BOR Reset, if enabled
- 3. POR Reset
- 4. Watchdog Timer, if enabled
- 5. Any external interrupt
- 6. Interrupts by peripherals capable of running during Sleep (see individual peripheral for more information)

The first three events will cause a device Reset. The last three events are considered a continuation of program execution. To determine whether a device Reset or wake-up event occurred, refer to Section 7.10 "Determining the Cause of a Reset".

When the SLEEP instruction is being executed, the next instruction (PC + 1) is prefetched. For the device to wake-up through an interrupt event, the corresponding interrupt enable bit must be enabled. Wake-up will occur regardless of the state of the GIE bit. If the GIE bit is disabled, the device continues execution at the instruction after the SLEEP instruction. If the GIE bit is enabled, the device executes the instruction after the SLEEP instruction, the device will then call the Interrupt Service Routine. In cases where the execution of the instruction following SLEEP is not desirable, the user should have a NOP after the SLEEP instruction.

The WDT is cleared when the device wakes up from Sleep, regardless of the source of wake-up.

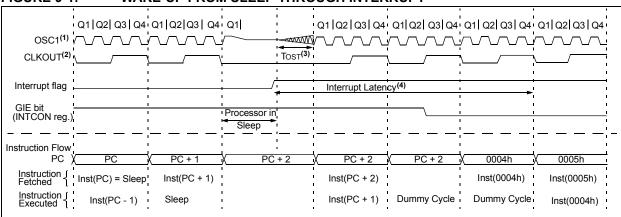
9.1.1 WAKE-UP USING INTERRUPTS

When global interrupts are disabled (GIE cleared) and any interrupt source has both its interrupt enable bit and interrupt flag bit set, one of the following will occur:

- If the interrupt occurs before the execution of a SLEEP instruction
 - SLEEP instruction will execute as a NOP.
 - WDT and WDT prescaler will not be cleared
 - TO bit of the STATUS register will not be set
 - PD bit of the STATUS register will not be cleared.
- If the interrupt occurs during or after the execution of a SLEEP instruction
 - SLEEP instruction will be completely
 - Device will immediately wake-up from Sleep
 - WDT and WDT prescaler will be cleared
 - TO bit of the STATUS register will be set
 - PD bit of the STATUS register will be cleared.

Even if the flag bits were checked before executing a ${\tt SLEEP}$ instruction, it may be possible for flag bits to become set before the ${\tt SLEEP}$ instruction completes. To determine whether a ${\tt SLEEP}$ instruction executed, test the $\overline{\tt PD}$ bit. If the $\overline{\tt PD}$ bit is set, the ${\tt SLEEP}$ instruction was executed as a NOP.





Note 1: XT, HS or LP Oscillator mode assumed.

- 2: CLKOUT is not available in XT, HS, or LP Oscillator modes, but shown here for timing reference.
- 3: Tost = 1024 Tosc (drawing not to scale). This delay applies only to XT, HS or LP Oscillator modes.
- 4: GIE = 1 assumed. In this case after wake-up, the processor calls the ISR at 0004h. If GIE = 0, execution will continue in-line.

TABLE 9-1: SUMMARY OF REGISTERS ASSOCIATED WITH POWER-DOWN MODE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	90
IOCAF	_	_	IOCAF5	IOCAF4	IOCAF3	IOCAF2	IOCAF1	IOCAF0	141
IOCAN	_	_	IOCAN5	IOCAN4	IOCAN3	IOCAN2	IOCAN1	IOCAN0	141
IOCAP	_	_	IOCAP5	IOCAP4	IOCAP3	IOCAP2	IOCAP1	IOCAP0	140
IOCBF ⁽¹⁾	IOCBF7	IOCBF6	IOCBF5	IOCBF4	_	_	_	_	142
IOCBN ⁽¹⁾	IOCBN7	IOCBN6	IOCBN5	IOCBN4	_	_	_	_	142
IOCBP ⁽¹⁾	IOCBP7	IOCBP6	IOCBP5	IOCBP4			_	_	141
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	91
PIE2	OSFIE	C2IE	C1IE	EEIE	BCL1IE	_	_	CCP2IE	92
PIE4 ⁽¹⁾	_	_	_	_	_	_	BCL2IE	SSP2IE	94
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	95
PIR2	OSFIF	C2IF	C1IF	EEIF	BCL1IF	_	_	CCP2IF	96
PIR4 ⁽¹⁾	_	_	_	_	_	_	BCL2IF	SSP2IF	98
STATUS	_	_	_	TO	PD	Z	DC	С	24
WDTCON	_			SWDTEN	103				

Legend: — Unimplemented, read as '0'. Shaded cells are not used in Power-down mode.

Note 1: PIC16(L)F1829 only.

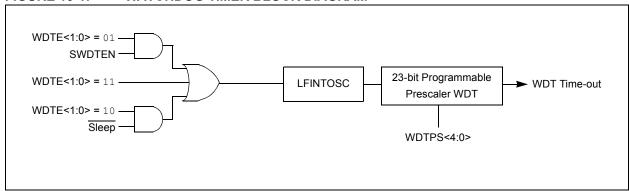
10.0 WATCHDOG TIMER

The Watchdog Timer is a system timer that generates a Reset if the firmware does not issue a CLRWDT instruction within the time-out period. The Watchdog Timer is typically used to recover the system from unexpected events.

The WDT has the following features:

- · Independent clock source
- · Multiple operating modes
 - WDT is always on
 - WDT is off when in Sleep
 - WDT is controlled by software
 - WDT is always off
- Configurable time-out period is from 1 ms to 256 seconds (typical)
- · Multiple Reset conditions
- · Operation during Sleep

FIGURE 10-1: WATCHDOG TIMER BLOCK DIAGRAM



10.1 Independent Clock Source

The WDT derives its time base from the 31 kHz LFINTOSC internal oscillator. Time intervals in this chapter are based on a nominal interval of 1 ms. See **Section 25.0 "Electrical Specifications"** for the LFINTOSC tolerances.

10.2 WDT Operating Modes

The Watchdog Timer module has four operating modes controlled by the WDTE<1:0> bits in Configuration Word 1. See Table 10-1.

10.2.1 WDT IS ALWAYS ON

When the WDTE bits of Configuration Word 1 are set to '11', the WDT is always on.

WDT protection is active during Sleep.

10.2.2 WDT IS OFF IN SLEEP

When the WDTE bits of Configuration Word 1 are set to '10', the WDT is on, except in Sleep.

WDT protection is not active during Sleep.

10.2.3 WDT CONTROLLED BY SOFTWARE

When the WDTE bits of Configuration Word 1 are set to '01', the WDT is controlled by the SWDTEN bit of the WDTCON register.

WDT protection is unchanged by Sleep. See Table 10-1 for more details.

TABLE 10-1: WDT OPERATING MODES

WDTE<1:0>	SWDTEN	Device Mode	WDT Mode
11	Х	Х	Active
1.0	77	Awake	Active
10	X	Sleep	Disabled
0.1	1	Х	Active
01	0	^	Disabled
00	Х	Х	Disabled

10.3 Time-Out Period

The WDTPS bits of the WDTCON register set the time-out period from 1 ms to 256 seconds (nominal). After a Reset, the default time-out period is two seconds.

10.4 Clearing the WDT

The WDT is cleared when any of the following conditions occur:

- · Any Reset
- CLRWDT instruction is executed
- · Device enters Sleep
- · Device wakes up from Sleep
- · Oscillator fail event
- · WDT is disabled
- · Oscillator Start-up TImer (OST) is running

See Table 10-2 for more information.

10.5 Operation During Sleep

When the device enters Sleep, the WDT is cleared. If the WDT is enabled during Sleep, the WDT resumes counting.

When the device exits Sleep, the WDT is cleared again. The WDT remains clear until the OST, if enabled, completes. See Section 5.0 "Oscillator Module (With Fail-Safe Clock Monitor)" for more information on the OST.

When a WDT time-out occurs while the device is in Sleep, no Reset is generated. Instead, the device wakes up and resumes operation. The TO and PD bits in the STATUS register are changed to indicate the event. See **Section 3.0 "Memory Organization"** for more information.

TABLE 10-2: WDT CLEARING CONDITIONS

Conditions	WDT
WDTE<1:0> = 00	
WDTE<1:0> = 01 and SWDTEN = 0	
WDTE<1:0> = 10 and enter Sleep	Cleared
CLRWDT Command	Cleared
Oscillator Fail Detected	
Exit Sleep + System Clock = T1OSC, EXTRC, INTOSC, EXTCLK	
Exit Sleep + System Clock = XT, HS, LP	Cleared until the end of OST

TABLE 10-2: WDT CLEARING CONDITIONS

Conditions	WDT
Change INTOSC divider (IRCF bits)	Unaffected

10.6 Watchdog Control Register

REGISTER 10-1: WDTCON: WATCHDOG TIMER CONTROL REGISTER

U-0	U-0	R/W-0/0	R/W-1/1	R/W-0/0	R/W-1/1	R/W-1/1	R/W-0/0
_	_		SWDTEN				
bit 7							bit 0

```
Legend:

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

u = Bit is unchanged x = Bit is unknown -m/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

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

```
bit 5-1
               WDTPS<4:0>: Watchdog Timer Period Select bits(1)
               Bit Value = Prescale Rate
               00000 = 1:32 (Interval 1 ms nominal)
               00001 = 1:64 (Interval 2 ms nominal)
               00010 = 1:128 (Interval 4 ms nominal)
               00011 = 1:256 (Interval 8 ms nominal)
               00100 = 1:512 (Interval 16 ms nominal)
               00101 = 1:1024 (Interval 32 ms nominal)
               00110 = 1:2048 (Interval 64 ms nominal)
               00111 = 1:4096 (Interval 128 ms nominal)
               01000 = 1:8192 (Interval 256 ms nominal)
               01001 = 1:16384 (Interval 512 ms nominal)
               01010 = 1:32768 (Interval 1s nominal)
               01011 = 1:65536 (Interval 2s nominal) (Reset value)
               01100 = 1:131072 (2^{17}) (Interval 4s nominal)
               01101 = 1:262144 (2<sup>18</sup>) (Interval 8s nominal)
               01110 = 1:524288 (2<sup>19</sup>) (Interval 16s nominal)
               01111 = 1:1048576 (2^{20}) (Interval 32s nominal)
               10000 = 1:2097152 (2^{21}) (Interval 64s nominal)
               10001 = 1:4194304 (2^{22})  (Interval 128s nominal)
               10010 = 1.8388608 (2^{23}) (Interval 256s nominal)
               10011 = Reserved. Results in minimum interval (1:32)
               11111 = Reserved. Results in minimum interval (1:32)
```

bit 0 **SWDTEN:** Software Enable/Disable for Watchdog Timer bit If WDTE<1:0> = 00:

This bit is ignored.

If WDTE<1:0> = 01:

1 = WDT is turned on

0 = WDT is turned off

If WDTE<1:0> = 1x:

This bit is ignored.

Note 1: Times are approximate. WDT time is based on 31 kHz LFINTOSC.

TABLE 10-3: SUMMARY OF REGISTERS ASSOCIATED WITH WATCHDOG TIMER

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
OSCCON	SPLLEN		IRCF<3:0>			_	SCS<1:0>		70
STATUS	_	_	_	TO	PD	Z	DC	С	24
WDTCON	_	1	WDTPS<4:0>					SWDTEN	103

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by Watchdog Timer.

TABLE 10-4: SUMMARY OF CONFIGURATION WORD WITH WATCHDOG TIMER

Name	Bits	Bit -/7	Bit -/6	Bit 13/5	Bit 12/4	Bit 11/3	Bit 10/2	Bit 9/1	Bit 8/0	Register on Page
CONITION	13:8	_	_	FCMEN	IESO	CLKOUTEN	BOREN<1:0>		CPD	50
CONFIG1	7:0	CP	MCLRE	PWRTE	WDTE<1:0>		FOSC<2:0>			50

Legend: — Unimplemented location, read as '0'. Shaded cells are not used by Watchdog Timer.

11.0 DATA EEPROM AND FLASH PROGRAM MEMORY CONTROL

The data EEPROM and Flash program memory are readable and writable during normal operation (full VDD range). These memories are not directly mapped in the register file space. Instead, they are indirectly addressed through the Special Function Registers (SFRs). There are six SFRs used to access these memories:

- EECON1
- EECON2
- EEDATL
- EEDATH
- EEADRL
- EEADRH

When interfacing the data memory block, EEDATL holds the 8-bit data for read/write, and EEADRL holds the address of the EEDATL location being accessed. These devices have 256 bytes of data EEPROM with an address range from 0h to 0FFh.

When accessing the program memory block, the EED-ATH:EEDATL register pair forms a 2-byte word that holds the 14-bit data for read/write, and the EEADRL and EEADRH registers form a 2-byte word that holds the 15-bit address of the program memory location being read.

The EEPROM data memory allows byte read and write. An EEPROM byte write automatically erases the location and writes the new data (erase before write).

The write time is controlled by an on-chip timer. The write/erase voltages are generated by an on-chip charge pump rated to operate over the voltage range of the device for byte or word operations.

Depending on the setting of the Flash Program Memory Self Write Enable bits WRT<1:0> of the Configuration Word 2, the device may or may not be able to write certain blocks of the program memory. However, reads from the program memory are always allowed.

When the device is code-protected, the device programmer can no longer access data or program memory. When code-protected, the CPU may continue to read and write the data EEPROM memory and Flash program memory.

11.1 EEADRL and EEADRH Registers

The EEADRH:EEADRL register pair can address up to a maximum of 256 bytes of data EEPROM or up to a maximum of 32K words of program memory.

When selecting a program address value, the MSB of the address is written to the EEADRH register and the LSB is written to the EEADRL register. When selecting a EEPROM address value, only the LSB of the address is written to the EEADRL register.

11.1.1 EECON1 AND EECON2 REGISTERS

EECON1 is the control register for EE memory accesses.

Control bit EEPGD determines if the access will be a program or data memory access. When clear, any subsequent operations will operate on the EEPROM memory. When set, any subsequent operations will operate on the program memory. On Reset, EEPROM is selected by default.

Control bits RD and WR initiate read and write, respectively. These bits cannot be cleared, only set, in software. They are cleared in hardware at completion of the read or write operation. The inability to clear the WR bit in software prevents the accidental, premature termination of a write operation.

The WREN bit, when set, will allow a write operation to occur. On power-up, the WREN bit is clear. The WRERR bit is set when a write operation is interrupted by a Reset during normal operation. In these situations, following Reset, the user can check the WRERR bit and execute the appropriate error handling routine.

Interrupt flag bit EEIF of the PIR2 register is set when write is complete. It must be cleared in the software.

Reading EECON2 will read all '0's. The EECON2 register is used exclusively in the data EEPROM write sequence. To enable writes, a specific pattern must be written to EECON2.

11.2 Using the Data EEPROM

The data EEPROM is a high-endurance, byte addressable array that has been optimized for the storage of frequently changing information (e.g., program variables or other data that are updated often). When variables in one section change frequently, while variables in another section do not change, it is possible to exceed the total number of write cycles to the EEPROM without exceeding the total number of write cycles to a single byte. Refer to Section 30.0 "Electrical Specifications". If this is the case, then a refresh of the array must be performed. For this reason, variables that change infrequently (such as constants, IDs, calibration, etc.) should be stored in Flash program memory.

11.2.1 READING THE DATA EEPROM MEMORY

To read a data memory location, the user must write the address to the EEADRL register, clear the EEPGD and CFGS control bits of the EECON1 register, and then set control bit RD. The data is available at the very next cycle, in the EEDATL register; therefore, it can be read in the next instruction. EEDATL will hold this value until another read or until it is written to by the user (during a write operation).

EXAMPLE 11-1: DATA EEPROM READ

```
BANKSEL EEADRL ;

MOVLW DATA_EE_ADDR;

MOVWF EEADRL ; Data Memory ; Address to read

BCF EECON1, CFGS; Deselect Config space
BCF EECON1, EEPGD; Point to DATA memory
BSF EECON1, RD ; EE Read

MOVF EEDATL, W ; W = EEDATL
```

Note: Data EEPROM can be read regardless of the setting of the CPD bit.

11.2.2 WRITING TO THE DATA EEPROM MEMORY

To write an EEPROM data location, the user must first write the address to the EEADRL register and the data to the EEDATL register. Then the user must follow a specific sequence to initiate the write for each byte.

The write will not initiate if the above sequence is not followed exactly (write 55h to EECON2, write AAh to EECON2, then set the WR bit) for each byte. Interrupts should be disabled during this code segment.

Additionally, the WREN bit in EECON1 must be set to enable write. This mechanism prevents accidental writes to data EEPROM due to errant (unexpected) code execution (i.e., lost programs). The user should keep the WREN bit clear at all times, except when updating EEPROM. The WREN bit is not cleared by hardware.

After a write sequence has been initiated, clearing the WREN bit will not affect this write cycle. The WR bit will be inhibited from being set unless the WREN bit is set.

At the completion of the write cycle, the WR bit is cleared in hardware and the EE Write Complete Interrupt Flag bit (EEIF) is set. The user can either enable this interrupt or poll this bit. EEIF must be cleared by software.

11.2.3 PROTECTION AGAINST SPURIOUS WRITE

There are conditions when the user may not want to write to the data EEPROM memory. To protect against spurious EEPROM writes, various mechanisms have been built-in. On power-up, WREN is cleared. Also, the Power-up Timer (64 ms duration) prevents EEPROM write.

The write initiate sequence and the WREN bit together help prevent an accidental write during:

- · Brown-out
- · Power glitch
- · Software malfunction

11.2.4 DATA EEPROM OPERATION DURING CODE-PROTECT

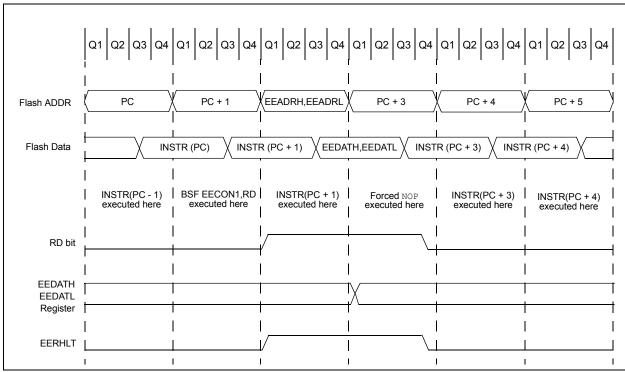
Data memory can be code-protected by programming the $\overline{\text{CPD}}$ bit in the Configuration Word 1 (Register 5-1) to '0'

When the data memory is code-protected, only the CPU is able to read and write data to the data EEPROM. It is recommended to code-protect the program memory when code-protecting data memory. This prevents anyone from replacing your program with a program that will access the contents of the data EEPROM.

EXAMPLE 11-2: DATA EEPROM WRITE

```
BANKSEL EEADRL
MOVLW DATA EE ADDR ;
MOVWF
        EEADRL ; Data Memory Address to write
        DATA_EE_DATA ;
MOVLW
        EEDATL ;Data Memory Value to write
EECON1, CFGS ;Deselect Configuration space
MOVWF
BCF
        EECON1, EEPGD ; Point to DATA memory
BCF
        EECON1, WREN ; Enable writes
BSF
BCF
        INTCON, GIE
                       ;Disable INTs.
MOVLW
        55h
MOVWF
        EECON2
                        ;Write 55h
MOVLW
        0AAh
MOVWF
        EECON2
                       ;Write AAh
BSF
        EECON1, WR
                       ;Set WR bit to begin write
        INTCON, GIE
BSF
                        ;Enable Interrupts
        EECON1, WREN
BCF
                       ;Disable writes
BTFSC
        EECON1, WR
                        ;Wait for write to complete
GOTO
                        ;Done
```

FIGURE 11-1: FLASH PROGRAM MEMORY READ CYCLE EXECUTION



11.3 Flash Program Memory Overview

It is important to understand the Flash program memory structure for erase and programming operations. Flash program memory is arranged in rows. A row consists of a fixed number of 14-bit program memory words. A row is the minimum block size that can be erased by user software.

Flash program memory may only be written or erased if the destination address is in a segment of memory that is not write-protected, as defined in bits WRT<1:0> of Configuration Word 2.

After a row has been erased, the user can reprogram all or a portion of this row. Data to be written into the program memory row is written to 14-bit wide data write latches. These write latches are not directly accessible to the user, but may be loaded via sequential writes to the EEDATH:EEDATL register pair.

Note: If the user wants to modify only a portion of a previously programmed row, then the contents of the entire row must be read and saved in RAM prior to the erase.

The number of data write latches is not equivalent to the number of row locations. During programming, user software will need to fill the set of write latches and initiate a programming operation multiple times in order to fully reprogram an erased row. For example, a device with a row size of 32 words and eight write latches will need to load the write latches with data and initiate a programming operation four times.

The size of a program memory row and the number of program memory write latches may vary by device. See Table 11-1 for details.

TABLE 11-1: FLASH MEMORY ORGANIZATION BY DEVICE

Device	Erase Block (Row) Size/ Boundary	Number of Write Latches/ Boundary		
PIC16(L)F1825	32 words,	32 words,		
PIC16(L)F1829	EEADRL<4:0>	EEADRL<4:0>		
	= 00000	= 00000		

11.3.1 READING THE FLASH PROGRAM MEMORY

To read a program memory location, the user must:

- Write the Least and Most Significant address bits to the EEADRH:EEADRL register pair.
- Clear the CFGS bit of the EECON1 register.
- Set the EEPGD control bit of the EECON1 register.
- 4. Then, set control bit RD of the EECON1 register.

Once the read control bit is set, the program memory Flash controller will use the second instruction cycle to read the data. This causes the second instruction immediately following the "BSF EECON1, RD" instruction to be ignored. The data is available in the very next cycle, in the EEDATH:EEDATL register pair; therefore, it can be read as two bytes in the following instructions.

EEDATH:EEDATL register pair will hold this value until another read or until it is written to by the user.

- Note 1: The two instructions following a program memory read are required to be NOPs. This prevents the user from executing a two-cycle instruction on the next instruction after the RD bit is set.
 - **2:** Flash program memory can be read regardless of the setting of the CP bit.

EXAMPLE 11-3: FLASH PROGRAM MEMORY READ

```
* This code block will read 1 word of program
* memory at the memory address:
   PROG ADDR HI : PROG ADDR LO
   data will be returned in the variables;
   PROG_DATA_HI, PROG_DATA_LO
   BANKSEL EEADRL ; Select Bank for EEPROM registers
MOVLW PROG_ADDR_LO ;
MOVWF EEADRL ; Store LSB of address
MOVLW PROG_ADDR_HI ;
MOVWL EEADRH ; Store MSB of address
            EECON1,CFGS ; Do not select Configuration Space EECON1,EEPGD ; Select Program Memory
   BCF
            EECON1, CFGS
    BSF
              INTCON, GIE ; Disable interrupts
    BCF
                                 ; Initiate read
    BSF
              EECON1,RD
    NOP
                                  ; Executed (Figure 11-1)
   NOP
                                  ; Ignored (Figure 11-1)
                                 ; Restore interrupts
   BSF
             INTCON, GIE
             EEDATL,W
    MOVF
                                ; Get LSB of word
    MOVWF
            PROG_DATA_LO ; Store in user location
            EEDATH,W ; Get MSB of word
PROG_DATA_HI ; Store in user location
    MOVE
    MOVWF
```

11.3.2 ERASING FLASH PROGRAM MEMORY

While executing code, program memory can only be erased by rows. To erase a row:

- Load the EEADRH:EEADRL register pair with the address of new row to be erased.
- 2. Clear the CFGS bit of the EECON1 register.
- Set the EEPGD, FREE, and WREN bits of the EECON1 register.
- Write 55h, then AAh, to EECON2 (Flash programming unlock sequence).
- Set control bit WR of the EECON1 register to begin the erase operation.
- Poll the FREE bit in the EECON1 register to determine when the row erase has completed.

See Example 11-4.

After the "BSF EECON1, WR" instruction, the processor requires two cycles to set up the erase operation. The user must place two NOP instructions after the WR bit is set. The processor will halt internal operations for the typical 2 ms erase time. This is not Sleep mode as the clocks and peripherals will continue to run. After the erase cycle, the processor will resume operation with the third instruction after the EECON1 write instruction.

11.3.3 WRITING TO FLASH PROGRAM MEMORY

Program memory is programmed using the following steps:

- Load the starting address of the word(s) to be programmed.
- 2. Load the write latches with data.
- 3. Initiate a programming operation.
- 4. Repeat steps 1 through 3 until all data is written.

Before writing to program memory, the word(s) to be written must be erased or previously unwritten. Program memory can only be erased one row at a time. No automatic erase occurs upon the initiation of the write.

Program memory can be written one or more words at a time. The maximum number of words written at one time is equal to the number of write latches. See Figure 11-2 for more details. The write latches are aligned to the address boundary defined by EEADRL as shown in Table 11-1. Write operations do not cross these boundaries. At the completion of a program memory write operation, the write latches are reset to contain 0x3FFF.

The following steps should be completed to load the write latches and program a block of program memory. These steps are divided into two parts. First, all write latches are loaded with data except for the last program memory location. Then, the last write latch is loaded and the programming sequence is initiated. A special unlock sequence is required to load a write latch with data or initiate a Flash programming operation. This unlock sequence should not be interrupted.

- Set the EEPGD and WREN bits of the EECON1 register.
- 2. Clear the CFGS bit of the EECON1 register.
- Set the LWLO bit of the EECON1 register. When the LWLO bit of the EECON1 register is '1', the write sequence will only load the write latches and will not initiate the write to Flash program memory.
- 4. Load the EEADRH:EEADRL register pair with the address of the location to be written.
- 5. Load the EEDATH:EEDATL register pair with the program memory data to be written.
- Write 55h, then AAh, to EECON2, then set the WR bit of the EECON1 register (Flash programming unlock sequence). The write latch is now loaded.
- Increment the EEADRH:EEADRL register pair to point to the next location.
- Repeat steps 5 through 7 until all but the last write latch has been loaded.
- Clear the LWLO bit of the EECON1 register. When the LWLO bit of the EECON1 register is '0', the write sequence will initiate the write to Flash program memory.
- 10. Load the EEDATH:EEDATL register pair with the program memory data to be written.
- 11. Write 55h, then AAh, to EECON2, then set the WR bit of the EECON1 register (Flash programming unlock sequence). The entire latch block is now written to Flash program memory.

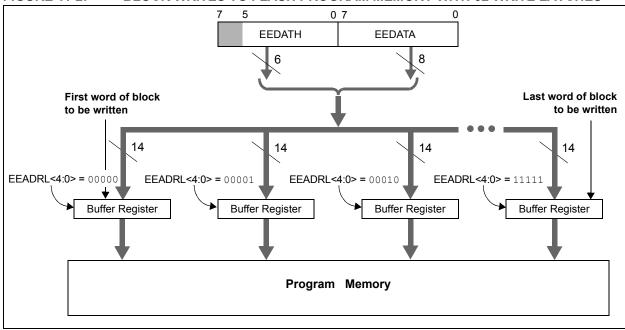
It is not necessary to load the entire write latch block with user program data. However, the entire write latch block will be written to program memory.

An example of the complete write sequence for eight words is shown in Example 11-5. The initial address is loaded into the EEADRH:EEADRL register pair; the eight words of data are loaded using indirect addressing.

After the "BSF EECON1, WR" instruction, the processor requires two cycles to set up the write operation. The user must place two NOP instructions after the WR bit is set. The processor will halt internal operations for the typical 2 ms, only during the cycle in which the write takes place (i.e., the last word of the block write). This is not Sleep mode as the clocks and peripherals will

continue to run. The processor does not stall when LWLO = 1, loading the write latches. After the write cycle, the processor will resume operation with the third instruction after the EECON1 write instruction.

FIGURE 11-2: BLOCK WRITES TO FLASH PROGRAM MEMORY WITH 32 WRITE LATCHES



EXAMPLE 11-4: ERASING ONE ROW OF PROGRAM MEMORY

```
; This row erase routine assumes the following:
; 1. A valid address within the erase block is loaded in ADDRH:ADDRL
; 2. ADDRH and ADDRL are located in shared data memory 0x70 - 0x7F
       BCF
                  INTCON, GIE
                               ; Disable ints so required sequences will execute properly
       BANKSEL
                  EEADRL
       MOVF
                 ADDRL, W
                                ; Load lower 8 bits of erase address boundary
       MOVWF
                 EEADRL
       MOVF
                 ADDRH,W
                                ; Load upper 6 bits of erase address boundary
       MOVWF
                 EEADRH
                  EECON1, EEPGD ; Point to program memory
       BSF
                  EECON1,CFGS ; Not configuration space
       BCF
       BSF
                  EECON1, FREE
                                ; Specify an erase operation
       BSF
                  EECON1, WREN
                                ; Enable writes
       MOVLW
                 55h
                                ; Start of required sequence to initiate erase
       MOVWF
                 EECON2
                                ; Write 55h
      MOVLW
                  0AAh
       MOVWF
                  EECON2
                                ; Write AAh
       BSF
                  EECON1,WR
                                ; Set WR bit to begin erase
       NOP
                                 ; Any instructions here are ignored as processor
                                 ; halts to begin erase sequence
       NOP
                                 ; Processor will stop here and wait for erase complete.
                                 ; after erase processor continues with 3rd instruction
       BCF
                  EECON1, WREN
                                 ; Disable writes
       BSF
                  INTCON, GIE
                                ; Enable interrupts
```

EXAMPLE 11-5: WRITING TO FLASH PROGRAM MEMORY

```
; This write routine assumes the following:
; 1. The 16 bytes of data are loaded, starting at the address in DATA ADDR
; 2. Each word of data to be written is made up of two adjacent bytes in DATA ADDR,
     stored in little endian format
; 3. A valid starting address (the least significant bits = 000) is loaded in ADDRH: ADDRL
; 4. ADDRH and ADDRL are located in shared data memory 0x70 - 0x7F
                  INTCON, GIE
                                ; Disable ints so required sequences will execute properly
                                ; Bank 3
       BANKSEL EEADRH
       MOVF
                  ADDRH, W
                                 ; Load initial address
       MOVWF
                  EEADRH
       MOVF
                  ADDRL, W
       MOVWF
                  EEADRL
       W.TVOM
                  LOW DATA_ADDR ; Load initial data address
       MOVWF
                  FSR01
                 HIGH DATA ADDR ; Load initial data address
       MOVWF
                 FSR0H
                 EECON1, EEPGD ; Point to program memory
       BSF
       BCF
                 EECON1,CFGS ; Not configuration space
                  EECON1, WREN ; Enable writes
                  EECON1, LWLO
                                 ; Only Load Write Latches
TIOOP
              FSR0++
EEDATL
FSR0++
       MTVOM
                                 ; Load first data byte into lower
       MOVWF
       MOVIW
                                 ; Load second data byte into upper
       MOVWF
                  EEDATH
                 EEADRL,W
       MOVF
                                 ; Check if lower bits of address are '000'
       XORLW
                  0x07
                                  ; Check if we're on the last of 8 addresses
       ANDIW
                  0 \times 0.7
       BTFSC
                  STATUS, Z
                                 ; Exit if last of eight words,
       GOTO
                  START WRITE
       MOVLW
                 55h
                                 ; Start of required write sequence:
       MOVWF
                 EECON2
                                 ; Write 55h
       MOVLW
                  0AAh
  Required
Sequence
       MOVWF
                                 ; Write AAh
                  EECON2
       BSF
                  EECON1.WR
                                 ; Set WR bit to begin write
       NOP
                                  ; Any instructions here are ignored as processor
                                  ; halts to begin write sequence
       NOP
                                  ; Processor will stop here and wait for write to complete.
                                  ; After write processor continues with 3rd instruction.
       INCF
                  EEADRL, F
                                 ; Still loading latches Increment address
       GOTO
                  LOOP
                                  ; Write next latches
START_WRITE
                  EECON1, LWLO
                                 ; No more loading latches - Actually start Flash program
                                  ; memory write
       MOVLW
                  55h
                                  ; Start of required write sequence:
       MOVWF
                  EECON2
                                  ; Write 55h
  Rednited
NOVLW
MOVWF
BSF
NOP
                  OAAh
                  EECON2
                                 ; Write AAh
                  EECON1,WR
                                 ; Set WR bit to begin write
       NOP
                                  ; Any instructions here are ignored as processor
                                  ; halts to begin write sequence
       NOP
                                  ; Processor will stop here and wait for write complete.
                                  ; after write processor continues with 3rd instruction
       BCF
                   EECON1, WREN
                                ; Disable writes
       BSF
                   INTCON, GIE
                                  ; Enable interrupts
```

11.4 Modifying Flash Program Memory

When modifying existing data in a program memory row, and data within that row must be preserved, it must first be read and saved in a RAM image. Program memory is modified using the following steps:

- Load the starting address of the row to be modified.
- Read the existing data from the row into a RAM image.
- Modify the RAM image to contain the new data to be written into program memory.
- Load the starting address of the row to be rewritten.
- 5. Erase the program memory row.
- Load the write latches with data from the RAM image.
- 7. Initiate a programming operation.
- 8. Repeat steps 6 and 7 as many times as required to reprogram the erased row.

11.5 User ID, Device ID and Configuration Word Access

Instead of accessing program memory or EEPROM data memory, the User ID's, Device ID/Revision ID and Configuration Words can be accessed when CFGS = 1 in the EECON1 register. This is the region that would be pointed to by PC<15> = 1, but not all addresses are accessible. Different access may exist for reads and writes. Refer to Table 11-2.

When read access is initiated on an address outside the parameters listed in Table 11-2, the EEDATH:EEDATL register pair is cleared.

TABLE 11-2: USER ID, DEVICE ID AND CONFIGURATION WORD ACCESS (CFGS = 1)

Address	Function	Read Access	Write Access
8000h-8003h	User IDs	Yes	Yes
8006h	Device ID/Revision ID	Yes	No
8007h-8008h	Configuration Words 1 and 2	Yes	No

EXAMPLE 11-3: CONFIGURATION WORD AND DEVICE ID ACCESS

```
^{\star} This code block will read 1 word of program memory at the memory address:
   PROG ADDR LO (must be 00h-08h) data will be returned in the variables;
   PROG DATA HI, PROG DATA LO
   BANKSEL EEADRL
                            ; Select correct Bank
   MOVLW PROG ADDR LO
                          ; Store LSB of address
   MOVWF EEADRL
   CLRF
          EEADRH
                           ; Clear MSB of address
          EECON1, CFGS
   BSF
                          ; Select Configuration Space
          INTCON, GIE
   BCF
                           ; Disable interrupts
           EECON1,RD
                            ; Initiate read
   BSF
                            ; Executed (See Figure 11-1)
   NOP
   NOP
                            ; Ignored (See Figure 11-1)
           INTCON, GIE
   BSF
                            ; Restore interrupts
   MOVF
           EEDATL, W
                           ; Get LSB of word
           PROG DATA LO
                          ; Store in user location
   MOVWF
   MOVE
           EEDATH,W
                           ; Get MSB of word
   MOVWF
           PROG DATA HI
                          ; Store in user location
```

11.6 Write Verify

Depending on the application, good programming practice may dictate that the value written to the data EEPROM or program memory should be verified (see Example 11-6) to the desired value to be written. Example 11-6 shows how to verify a write to EEPROM.

EXAMPLE 11-6: EEPROM WRITE VERIFY

```
BANKSEL EEDATL ;

MOVF EEDATL, W ; EEDATL not changed ; from previous write

BSF EECON1, RD ; YES, Read the ; value written

XORWF EEDATL, W ;

BTFSS STATUS, Z ; Is data the same

GOTO WRITE_ERR ; No, handle error ; Yes, continue
```

11.7 EEPROM and Flash Control Registers

REGISTER 11-1: EEDATL: EEPROM LOW BYTE DATA REGISTER

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
	EEDAT<7:0>						
bit 7 bit 0							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

bit 7-0 **EEDAT<7:0>**: Read/write value for EEPROM data byte or Least Significant bits of program memory

REGISTER 11-2: EEDATH: EEPROM DATA HIGH BYTE REGISTER

U-0	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
_	_			EEDA	T<13:8>		
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

bit 7-6 Unimplemented: Read as '0'

bit 5-0 **EEDAT<13:8>**: Read/write value for Most Significant bits of program memory

REGISTER 11-3: EEADRL: EEPROM ADDRESS REGISTER

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

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

bit 7-0 **EEADR<7:0>**: Specifies the Least Significant bits for program memory address or EEPROM address

REGISTER 11-4: EEADRH: EEPROM ADDRESS HIGH BYTE REGISTER

U-1	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
_				EEADR<14:8>	>		
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

bit 7 **Unimplemented:** Read as '1'

bit 6-0 **EEADR<14:8>**: Specifies the Most Significant bits for program memory address or EEPROM address

REGISTER 11-5: EECON1: EEPROM CONTROL 1 REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W/HC-0/0	R/W-x/q	R/W-0/0	R/S/HC-0/0	R/S/HC-0/0
EEPGD	CFGS	LWLO	FREE	WRERR	WREN	WR	RD
bit 7							bit 0

Legend:

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

S = Bit can only be set x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared HC = Bit is cleared by hardware

bit 7 **EEPGD:** Flash Program/Data EEPROM Memory Select bit

1 = Accesses program space Flash memory

0 = Accesses data EEPROM memory

bit 6 CFGS: Flash Program/Data EEPROM or Configuration Select bit

1 = Accesses Configuration, User ID and Device ID Registers

0 = Accesses Flash Program or data EEPROM Memory

bit 5 LWLO: Load Write Latches Only bit

If CFGS = 1 (Configuration space) OR CFGS = 0 and EEPGD = 1 (program Flash):

- 1 = The next WR command does not initiate a write; only the program memory latches are updated.
- 0 = The next WR command writes a value from EEDATH:EEDATL into program memory latches and initiates a write of all the data stored in the program memory latches.

<u>If CFGS = 0 and EEPGD = 0:</u> (Accessing data EEPROM)

LWLO is ignored. The next WR command initiates a write to the data EEPROM.

bit 4 FREE: Program Flash Erase Enable bit

If CFGS = 1 (Configuration space) OR CFGS = 0 and EEPGD = 1 (program Flash):

- 1 = Performs an erase operation on the next WR command (cleared by hardware after completion of erase).
- 0 = Performs a write operation on the next WR command.

If EEPGD = 0 and CFGS = 0: (Accessing data EEPROM)

FREE is ignored. The next WR command will initiate both a erase cycle and a write cycle.

- bit 3 WRERR: EEPROM Error Flag bit
 - 1 = Condition indicates an improper program or erase sequence attempt or termination (bit is set automatically on any set attempt (write '1') of the WR bit).
 - 0 = The program or erase operation completed normally.
- bit 2 WREN: Program/Erase Enable bit
 - 1 = Allows program/erase cycles
 - 0 = Inhibits programming/erasing of program Flash and data EEPROM
- bit 1 WR: Write Control bit
 - 1 = Initiates a program Flash or data EEPROM program/erase operation.

 The operation is self-timed and the bit is cleared by hardware once operation is complete.

The WR bit can only be set (not cleared) in software.

0 = Program/erase operation to the Flash or data EEPROM is complete and inactive.

- bit 0 RD: Read Control bit
 - 1 = Initiates an program Flash or data EEPROM read. Read takes one cycle. RD is cleared in hardware. The RD bit can only be set (not cleared) in software.
 - 0 = Does not initiate a program Flash or data EEPROM data read.

REGISTER 11-6: EECON2: EEPROM CONTROL 2 REGISTER

W-0/0	W-0/0	W-0/0	W-0/0	W-0/0	W-0/0	W-0/0	W-0/0
EEPROM Control Register 2							
bit 7 bit 0							

Legend:

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

S = Bit can only be set x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

bit 7-0 Data EEPROM Unlock Pattern bits

To unlock writes, a 55h must be written first, followed by an AAh, before setting the WR bit of the EECON1 register. The value written to this register is used to unlock the writes. There are specific timing requirements on these writes. Refer to **Section 11.2.2 "Writing to the Data EEPROM Memory"** for more information.

TABLE 11-3: SUMMARY OF REGISTERS ASSOCIATED WITH DATA EEPROM

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
EECON1	EEPGD	CFGS	LWLO	FREE	WRERR	WREN	WR	RD	117
EECON2	EEPROM Control Register 2 (not a physical register)								118*
EEADRL	EEADRL<7:0>							116	
EEADRH	(1) EEADRH<6:0						116		
EEDATL	EEDATL<7:0>								116
EEDATH	_	_			EEDAT	H<5:0>			116
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	90
PIE2	OSFIE	C2IE	C1IE	EEIE	BCL1IE	_	_	CCP2IE	92
PIR2	OSFIF	C2IF	C1IF	EEIF	BCL1IF	_	_	CCP2IF	96

Legend: — Unimplemented location, read as '0'. Shaded cells are not used by data EEPROM module.

Note 1: Unimplemented, read as '1'.

^{*} Page provides register information.

12.0 I/O PORTS

Depending on the device selected and peripherals enabled, there are up to two ports available. In general, when a peripheral is enabled on a port pin, that pin cannot be used as a general purpose output. However, the pin can still be read.

Each port has three standard registers for its operation. These registers are:

- · TRISx registers (data direction)
- PORTx registers (reads the levels on the pins of the device)
- · LATx registers (output latch)

Some ports may have one or more of the following additional registers. These registers are:

- · ANSELx (analog select)
- WPUx (weak pull-up)
- · INLVLx (input level control)

TABLE 12-1: PORT AVAILABILITY PER DEVICE

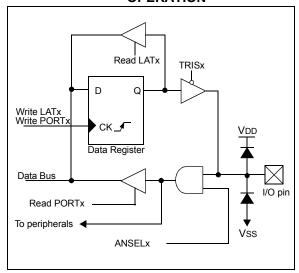
Device	PORTA	PORTB	PORTC
PIC16(L)F1825	•		•
PIC16(L)F1829	•	•	•

The Data Latch (LATx registers) is useful for read-modify-write operations on the value that the I/O pins are driving.

A write operation to the LATx register has the same effect as a write to the corresponding PORTx register. A read of the LATx register reads of the values held in the I/O PORT latches, while a read of the PORTx register reads the actual I/O pin value.

Ports that support analog inputs have an associated ANSELx register. When an ANSEL bit is set, the digital input buffer associated with that bit is disabled. Disabling the input buffer prevents analog signal levels on the pin between a logic high and low from causing excessive current in the logic input circuitry. A simplified model of a generic I/O port, without the interfaces to other peripherals, is shown in Figure 12-1.

FIGURE 12-1: GENERIC I/O PORT OPERATION



EXAMPLE 12-1: INITIALIZING PORTA

```
; This code example illustrates
; initializing the PORTA register. The
; other ports are initialized in the same
: manner.
BANKSEL PORTA
CLRF
        PORTA
                    ;Init PORTA
BANKSEL LATA
                    ;Data Latch
CLRF
       T.ATA
BANKSEL ANSELA
CLRF
      ANSELA
                    ;digital I/O
BANKSEL TRISA
MOVLW B'00111000'; Set RA<5:3> as inputs
MOVWF
        TRISA
                    ; and set RA<2:0> as
                    ;outputs
```

12.1 Alternate Pin Function

The Alternate Pin Function Control 0 (APFCON0) and Alternate Pin Function Control 1 (APFCON1) registers are used to steer specific peripheral input and output functions between different pins. The APFCON0 and APFCON1 registers are shown in Register 12-1 and Register 12-2. For this device family, the following functions can be moved between different pins.

- RX/DT/TX/CK
- SDO1
- SS (Slave Select)
- T1G
- P1B/P1C/P1D/P2B
- CCP1/P1A/CCP2

These bits have no effect on the values of any TRIS register. PORT and TRIS overrides will be routed to the correct pin. The unselected pin will be unaffected.

REGISTER 12-1: APFCON0: ALTERNATE PIN FUNCTION CONTROL REGISTER 0

R/W-0/0	R/W-0/0	R/W-0/0	U-0	R/W-0/0	R/W-0/0	U-0	U-0
RXDTSEL	SDO1SEL	SS1SEL	_	T1GSEL	TXCKSEL	_	_
bit 7							bit 0

Legend:R = Readable bitW = Writable bitU = Unimplemented bit, read as '0'u = Bit is unchangedx = Bit is unknown-n/n = Value at POR and BOR/Value at all other Resets'1' = Bit is set'0' = Bit is cleared

bit 7 **RXDTSEL:** Pin Selection bit For 14 Pin Devices (PIC16(L)F1825): 0 = RX/DT function is on RC5 1 = RX/DT function is on RA1 For 20 Pin Devices (PIC16(L)F1829): 0 = RX/DT function is on RB5 1 = RX/DT function is on RC5 bit 6 SDO1SEL: Pin Selection bit For 14 Pin Devices (PIC16(L)F1825): 0 = SDO1 function is on RC2 1 = SDO1 function is on RA4 For 20 Pin Devices (PIC16(L)F1829): Bit is read-only, '0' SDO1 function is always on RC7. bit 5 SS1SEL: Pin Selection bit For 14 Pin Devices (PIC16(L)F1825): $0 = \overline{SS1}$ function is on RC3 $1 = \overline{SS1}$ function is on RA3 For 20 Pin Devices (PIC16(L)F1829): Bit is read-only, '0' SS1 function is always on RC6. bit 4 Unimplemented: Read as '0' bit 3 T1GSEL: Pin Selection bit 0 = T1G function is on RA4 1 = T1G function is on RA3 bit 2 TXCKSEL: Pin Selection bit For 14 Pin Devices (PIC16(L)F1825): 0 = TX/CK function is on RC4 1 = TX/CK function is on RA0 For 20 Pin Devices (PIC16(L)F1829): 0 = TX/CK function is on RB7 1 = TX/CK function is on RC4 bit 1-0 Unimplemented: Read as '0'

REGISTER 12-2: APFCON1: ALTERNATE PIN FUNCTION CONTROL REGISTER 1

U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
_	_	SDO2SEL ⁽¹⁾	SS2SEL ⁽¹⁾	P1DSEL	P1CSEL	P2BSEL	CCP2SEL
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7-6 bit 5	Unimplemented: Read as '0' SDO2SEL: Pin Selection bit ⁽¹⁾
	0 = SDO2 function is on RC1 1 = SDO2 function is on RA5
bit 4	SS2SEL: Pin Selection bit ⁽¹⁾ 0 = SS2 function is on RC0
	$1 = \frac{332}{\text{SS2}} \text{ function is on RA4}$
bit 3	P1DSEL: Pin Selection bit
	0 = P1D function is on RC2 1 = P1D function is on RC0
	1 12 10110110111100111100
bit 2	P1CSEL: Pin Selection bit
	0 = P1C function is on RC3
	1 = P1C function is on RC1
bit 1	P2BSEL: Pin Selection bit
	0 = P2B function is on RC2
	1 = P2B function is on RA4
bit 0	CCP2SEL: Pin Selection bit
	0 = CCP2 function is on RC3
	1 = CCP2 function is on RA5

Note 1: PIC16(L)F1829 only.

12.2 PORTA Registers

PORTA is a 6-bit wide, bidirectional port. The corresponding data direction register is TRISA (Register 12-4). Setting a TRISA bit (= 1) will make the corresponding PORTA pin an input (i.e., disable the output driver). Clearing a TRISA bit (= 0) will make the corresponding PORTA pin an output (i.e., enables output driver and puts the contents of the output latch on the selected pin). The exception is RA3, which is input only and its TRIS bit will always read as '1'. Example 12-2 shows how to initialize a port.

Reading the PORTA register (Register 12-3) reads the status of the pins, whereas writing to it will write to the PORT latch. All write operations are read-modify-write operations. Therefore, a write to a port implies that the port pins are read, this value is modified and then written to the PORT data latch (LATA).

The TRISA register (Register 12-4) controls the PORTA pin output drivers, even when they are being used as analog inputs. The user should ensure the bits in the TRISA register are maintained set when using them as analog inputs. I/O pins configured as analog input always read '0'.

The INLVLA register (Register 12-8) controls the input voltage threshold for each of the available PORTA input pins. A selection between the Schmitt Trigger CMOS or the TTL Compatible thresholds is available. The input threshold is important in determining the value of a read of the PORTA register and also the level at which an Interrupt-on-Change occurs, if that feature is enabled. See Section 30.4 "DC Characteristics: PIC16(L)F1825/1829-I/E" for more information on threshold levels.

Note: Changing the input threshold selection should be performed while all peripheral modules are disabled. Changing the threshold level during the time a module is active may inadvertently generate a transition associated with an input pin, regardless of the actual voltage level on that pin.

12.2.1 ANSELA REGISTER

The ANSELA register (Register 12-6) is used to configure the Input mode of an I/O pin to analog. Setting the appropriate ANSELA bit high will cause all digital reads on the pin to be read as '0' and allow analog functions on the pin to operate correctly.

The state of the ANSELA bits has no effect on digital output functions. A pin with TRIS clear and ANSEL set will still operate as a digital output, but the Input mode will be analog. This can cause unexpected behavior when executing read-modify-write instructions on the affected port.

Note: The ANSELA bits default to the Analog mode after Reset. To use any pins as digital general purpose or peripheral inputs, the corresponding ANSEL bits must be initialized to '0' by user software.

EXAMPLE 12-2: INITIALIZING PORTA

BANKSEL	PORTA	;
CLRF	PORTA	;Init PORTA
BANKSEL	LATA	;Data Latch
CLRF	LATA	;
BANKSEL	ANSELA	;
CLRF	ANSELA	;digital I/O
BANKSEL	TRISA	;
MOVLW	B'00111000'	;Set RA<5:3> as inputs
MOVWF	TRISA	;and set RA<2:0> as
		;outputs

12.2.2 PORTA FUNCTIONS AND OUTPUT PRIORITIES

Each PORTA pin is multiplexed with other functions. The pins, their combined functions and their output priorities are briefly described here. For additional information, refer to the appropriate section in this data sheet.

When multiple outputs are enabled, the actual pin control goes to the peripheral with the lowest number in the following lists.

Analog input functions, such as ADC, comparator and Cap Sense inputs, are not shown in the priority lists. These inputs are active when the I/O pin is set for Analog mode using the ANSELx registers. Digital output functions may control the pin when it is in Analog mode with the priority shown below.

TABLE 12-2: PORTA OUTPUT PRIORITY

Pin Name	Function Priority ⁽¹⁾
RA0	ICSPDAT ICDDAT DACOUT
RA1	ICSPCLK ICDCLK RX/DT ⁽²⁾
RA2	SRQ C1OUT CCP3
RA3	None (input only)
RA4	CLKOUT T10SO CLKR SDO1 P2B ⁽²⁾
RA5	SDO2 (PIC16(L)F1829 only) CCP2 ⁽²⁾ /P2A ⁽²⁾

Note 1: Priority listed from highest to lowest.

2: Pin function is selectable via the APFCON0 or APFCON1 register.

REGISTER 12-3: PORTA: PORTA REGISTER

U-0	U-0	R/W-x/x	R/W-x/x	R-x/x	R/W-x/x	R/W-x/x	R/W-x/x
_	_	RA5	RA4	RA3	RA2	RA1	RA0
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

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

bit 5-0 RA<5:0>: PORTA I/O Value bits⁽¹⁾

1 = Port pin is \geq VIH 0 = Port pin is \leq VIL

Note 1: Writes to PORTA are actually written to corresponding LATA register. Reads from PORTA register is return of actual I/O pin values.

REGISTER 12-4: TRISA: PORTA TRI-STATE REGISTER

U-0	U-0	R/W-1/1	R/W-1/1	R-1/1	R/W-1/1	R/W-1/1	R/W-1/1
_	_	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

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

bit 5-4 TRISA<5:4>: PORTA Tri-State Control bit

1 = PORTA pin configured as an input (tri-stated)

0 = PORTA pin configured as an output

bit 3 TRISA3: RA3 Port Tri-State Control bit

This bit is always '1' as RA3 is an input only

bit 2-0 TRISA<2:0>: PORTA Tri-State Control bit

1 = PORTA pin configured as an input (tri-stated)

0 = PORTA pin configured as an output

REGISTER 12-5: LATA: PORTA DATA LATCH REGISTER

U-0	U-0	R/W-x/u	R/W-x/u	U-0	R/W-x/u	R/W-x/u	R/W-x/u
_	_	LATA5	LATA4	_	LATA2	LATA1	LATA0
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets '1' = Bit is set '0' = Bit is cleared

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

bit 5-4 LATA<5:4>: RA<5:4> Output Latch Value bits⁽¹⁾

bit 3 Unimplemented: Read as '0'

bit 2-0 LATA<2:0>: RA<2:0> Output Latch Value bits⁽¹⁾

Note 1: Writes to PORTA are actually written to corresponding LATA register. Reads from PORTA register is return

of actual I/O pin values.

REGISTER 12-6: ANSELA: PORTA ANALOG SELECT REGISTER

U-0	U-0	U-0	R/W-1/1	U-0	R/W-1/1	R/W-1/1	R/W-1/1
_	_	_	ANSA4	_	ANSA2	ANSA1	ANSA0
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets '1' = Bit is set '0' = Bit is cleared

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

bit 4 ANSA4: Analog Select between Analog or Digital Function on pins RA4, respectively

0 = Digital I/O. Pin is assigned to port or digital special function.

1 = Analog input. Pin is assigned as analog input⁽¹⁾. Digital input buffer disabled.

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

bit 2-0 ANSA<2:0>: Analog Select between Analog or Digital Function on pins RA<2:0>, respectively

0 = Digital I/O. Pin is assigned to port or digital special function.

1 = Analog input. Pin is assigned as analog input (1). Digital input buffer disabled.

Note 1: When setting a pin to an analog input, the corresponding TRIS bit must be set to Input mode in order to allow external control of the voltage on the pin.

REGISTER 12-7: WPUA: WEAK PULL-UP PORTA REGISTER

U-0	U-0	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
_	_	WPUA5	WPUA4	WPUA3	WPUA2	WPUA1	WPUA0
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets '1' = Bit is set '0' = Bit is cleared

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

bit 5-0 WPUA<5:0>: Weak Pull-up Register bits

1 = Pull-up enabled0 = Pull-up disabled

Note 1: Global WPUEN bit of the OPTION_REG register must be cleared for individual pull-ups to be enabled.

2: The weak pull-up device is automatically disabled if the pin is in configured as an output.

REGISTER 12-8: INLVLA: PORTA INPUT LEVEL CONTROL REGISTER

U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-1/1	R/W-0/0	R/W-0/0
_	_	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

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

bit 5-0 INLVLA<5:0>: PORTA Input Level Select bits

For RA<5:0> pins, respectively

1 = ST input used for port reads and interrupt-on-change 0 = TTL input used for port reads and interrupt-on-change

TABLE 12-3: SUMMARY OF REGISTERS ASSOCIATED WITH PORTA

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ANSELA	_	_		ANSA4	_	ANSA2	ANSA1	ANSA0	126
APFCON0	RXDTSEL	SDO1SEL ⁽²⁾	SS1SEL ⁽²⁾	_	T1GSEL	TXCKSEL	_	_	121
APFCON1	_	_	SDO2SEL ⁽¹⁾	SS2SEL ⁽¹⁾	P1DSEL	P1CSEL	P2BSEL	CCP2SEL	122
INLVLA	_	_	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	127
LATA	_	_	LATA5	LATA4	_	LATA2	LATA1	LATA0	126
OPTION_REG	WPUEN	INTEDG	TMR0CS	TMR0SE	PSA		PS<2:0>		185
PORTA	_	_	RA5	RA4	RA3	RA2	RA1	RA0	125
TRISA	_	_	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	125
WPUA	_	_	WPUA5	WPUA4	WPUA3	WPUA2	WPUA1	WPUA0	127

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by PORTA.

Note 1: PIC16(L)F1829 only.
2: PIC16(L)F1825 only.

TABLE 12-4: SUMMARY OF CONFIGURATION WORD WITH PORTA

Name	Bits	Bit -/7	Bit -/6	Bit 13/5	Bit 12/4	Bit 11/3	Bit 10/2	Bit 9/1	Bit 8/0	Register on Page
CONTION	13:8	-	_	FCMEN	IESO	CLKOUTEN	BORE	V<1:0>	CPD	
CONFIG1	7:0	CP	MCLRE	PWRTE	WDTE	E<1:0>		FOSC<2:0>		50

Legend: — Unimplemented location, read as '0'. Shaded cells are not used by PORTA.

12.3 PORTB Registers (PIC16(L)F1829 only)

PORTB is a 4-bit wide, bidirectional port. The corresponding data direction register is TRISB (Register 12-10). Setting a TRISB bit (= 1) will make the corresponding PORTB pin an input (i.e., put the corresponding output driver in a High-Impedance mode). Clearing a TRISB bit (= 0) will make the corresponding PORTB pin an output (i.e., enable the output driver and put the contents of the output latch on the selected pin). Example 12-2 shows how to initialize a port.

Reading the PORTB register (Register 12-9) reads the status of the pins, whereas writing to it will write to the PORT latch. All write operations are read-modify-write operations. Therefore, a write to a port implies that the port pins are read, this value is modified and then written to the PORT data latch (LATB).

The TRISB register (Register 12-10) controls the PORTB pin output drivers, even when they are being used as analog inputs. The user should ensure the bits in the TRISB register are maintained set when using them as analog inputs. I/O pins configured as analog input always read '0'.

The INLVLB register (Register 12-14) controls the input voltage threshold for each of the available PORTB input pins. A selection between the Schmitt Trigger CMOS or the TTL Compatible thresholds is available. The input threshold is important in determining the value of a read of the PORTB register and also the level at which an Interrupt-on-Change occurs, if that feature is enabled. See Section 30.4 "DC Characteristics: PIC16(L)F1825/1829-I/E" for more information on threshold levels.

Note: Changing the input threshold selection should be performed while all peripheral modules are disabled. Changing the threshold level during the time a module is active may inadvertently generate a transition associated with an input pin, regardless of the actual voltage level on that pin.

12.3.1 ANSELB REGISTER

Note:

The ANSELB register (Register 12-12) is used to configure the Input mode of an I/O pin to analog. Setting the appropriate ANSELB bit high will cause all digital reads on the pin to be read as '0' and allow analog functions on the pin to operate correctly.

The state of the ANSELB bits has no effect on digital output functions. A pin with TRIS clear and ANSELB set will still operate as a digital output, but the Input mode will be analog. This can cause unexpected behavior when executing read-modify-write instructions on the affected port.

The ANSELB bits default to the Analog mode after Reset. To use any pins as digital general purpose or peripheral inputs, the corresponding ANSEL bits must be initialized to '0' by user software.

12.3.2 PORTB FUNCTIONS AND OUTPUT PRIORITIES

Each PORTB pin is multiplexed with other functions. The pins, their combined functions and their output priorities are briefly described here. For additional information, refer to the appropriate section in this data sheet.

When multiple outputs are enabled, the actual pin control goes to the peripheral with the lowest number in the following lists.

Analog input and some digital input functions are not included in the list below. These input functions can remain active when the pin is configured as an output. Certain digital input functions override other port functions and are included in the priority list.

TABLE 12-5: PORTB OUTPUT PRIORITY

Pin Name	Function Priority ⁽¹⁾
RB4	SDA
RB5	SDA2 RX ⁽²⁾ /DT ⁽²⁾
RB6	SCL/SCK
RB7	TX ⁽²⁾ /CK ⁽²⁾

Note 1: Priority listed from highest to lowest.

2: Pin function is selectable via the APFCON0 or APFCON1 register.

REGISTER 12-9: PORTB: PORTB REGISTER

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	U-0	U-0	U-0	U-0
RB7	RB6	RB5	RB4	_	_	_	_
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

bit 7-4 RB<7:4>: PORTB General Purpose I/O Pin bits

1 = Port pin is \geq VIH 0 = Port pin is \leq VIL

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

REGISTER 12-10: TRISB: PORTB TRI-STATE REGISTER

R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	U-0	U-0	U-0	U-0
TRISB7	TRISB6	TRISB5	TRISB4	_	_	_	_
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

bit 7-4 TRISB<7:4>: PORTB Tri-State Control bits

1 = PORTB pin configured as an input (tri-stated)

0 = PORTB pin configured as an output

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

REGISTER 12-11: LATB: PORTB DATA LATCH REGISTER

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	U-0	U-0	U-0	U-0
LATB7	LATB6	LATB5	LATB4	_	_	_	_
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

bit 7-4 LATB<7:4>: PORTB Output Latch Value bits⁽¹⁾

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

Note 1: Writes to PORTB are actually written to corresponding LATB register. Reads from PORTB register is return of actual I/O pin values.

REGISTER 12-12: ANSELB: PORTB ANALOG SELECT REGISTER

R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	U-0	U-0	U-0	U-0
ANSB7	ANSB6	ANSB5	ANSB4	_	_	_	_
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

bit 7-4 ANSB<7:4>: Analog Select between Analog or Digital Function on pins RB<7:4>, respectively

0 = Digital I/O. Pin is assigned to port or digital special function.

1 = Analog input. Pin is assigned as analog input⁽¹⁾. Digital input buffer disabled.

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

Note 1: When setting a pin to an analog input, the corresponding TRIS bit must be set to Input mode in order to allow external control of the voltage on the pin.

REGISTER 12-13: WPUB: WEAK PULL-UP PORTB REGISTER

R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	U-0	U-0	U-0	U-0
WPUB7	WPUB6	WPUB5	WPUB4	_	_		_
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

bit 7-4 WPUB<7:4>: Weak Pull-up Register bits

1 = Pull-up enabled

0 = Pull-up disabled

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

Note 1: Global WPUEN bit of the OPTION_REG register must be cleared for individual pull-ups to be enabled.

2: The weak pull-up device is automatically disabled if the pin is in configured as an output.

REGISTER 12-14: INLVLB: PORTB INPUT LEVEL CONTROL REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	U-0	U-0	U-0
INLVLB7	INLVLB6	INLVLB5	INLVLB4	_	_	_	_
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

bit 7-4 INLVLB<7:4>: PORTB Input Level Select bits

For RB<7:4> pins, respectively

1 = ST input used for PORT reads and interrupt-on-change 0 = TTL input used for PORT reads and interrupt-on-change

bit 7-6 Unimplemented: Read as '0'

SUMMARY OF REGISTERS ASSOCIATED WITH PORTB⁽¹⁾ **TABLE 12-6:**

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ANSELB	ANSB7	ANSB6	ANSB5	ANSB4	_	_	_	_	132
INLVLB	INLVLB7	INLVLB6	INLVLB5	INLVLB4	_	_	_	_	132
LATB	LATB7	LATB6	LATB5	LATB4	_	_	_	_	131
PORTB	RB7	RB6	RB5	RB4	_	_	_	_	131
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	_	_	_	_	131
WPUB	WPUB7	WPUB6	WPUB5	WPUB4	1	1	_	-	132

x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by PORTB. PIC16(L)F1829 only. Legend: Note 1:

12.4 PORTC Registers

PORTC is a 6-bit wide (8-bit wide for PIC16(L)F1829), bidirectional port. The corresponding data direction register is TRISC (Register 12-16). Setting a TRISC bit (= 1) will make the corresponding PORTC pin an input (i.e., put the corresponding output driver in a High-Impedance mode). Clearing a TRISC bit (= 0) will make the corresponding PORTC pin an output (i.e., enable the output driver and put the contents of the output latch on the selected pin). Example 12-2 shows how to initialize a port.

Reading the PORTC register (Register 12-15) reads the status of the pins, whereas writing to it will write to the PORT latch. All write operations are read-modify-write operations. Therefore, a write to a port implies that the port pins are read, this value is modified and then written to the PORT data latch (LATC).

The TRISC register (Register 12-16) controls the PORTC pin output drivers, even when they are being used as analog inputs. The user should ensure the bits in the TRISC register are maintained set when using them as analog inputs. I/O pins configured as analog input always read '0'.

The INLVLC register (Register 12-20) controls the input voltage threshold for each of the available PORTC input pins. A selection between the Schmitt Trigger CMOS or the TTL Compatible thresholds is available. The input threshold is important in determining the value of a read of the PORTC register and also the level at which an Interrupt-on-Change occurs, if that feature is enabled. See Section 30.4 "DC Characteristics: PIC16(L)F1825/1829-I/E" for more information on threshold levels.

Note: Changing the input threshold selection should be performed while all peripheral modules are disabled. Changing the threshold level during the time a module is active may inadvertently generate a transition associated with an input pin, regardless of the actual voltage level on

that pin.

12.4.1 ANSELC REGISTER

The ANSELC register (Register 12-18) is used to configure the Input mode of an I/O pin to analog. Setting the appropriate ANSELC bit high will cause all digital reads on the pin to be read as '0' and allow analog functions on the pin to operate correctly.

The state of the ANSELC bits has no effect on digital output functions. A pin with TRIS clear and ANSELC set will still operate as a digital output, but the Input mode will be analog. This can cause unexpected behavior when executing read-modify-write instructions on the affected port.

Note: The ANSELC bits default to the Analog mode after Reset. To use any pins as digital general purpose or peripheral inputs, the corresponding ANSEL bits must be initialized to '0' by user software.

12.4.2 PORTC FUNCTIONS AND OUTPUT PRIORITIES

Each PORTC pin is multiplexed with other functions. The pins, their combined functions and their output priorities are briefly described here. For additional information, refer to the appropriate section in this data sheet.

When multiple outputs are enabled, the actual pin control goes to the peripheral with the lowest number in the following lists.

Analog input and some digital input functions are not included in the list below. These input functions can remain active when the pin is configured as an output. Certain digital input functions override other port functions and are included in the priority list.

TABLE 12-7: PORTC OUTPUT PRIORITY

Pin Name	Function Priority ⁽¹⁾
RC0	SCL (PIC16(L)F1825 only) SCK (PIC16(L)F1825 only) P1D ⁽²⁾
RC1	SDA1 (PIC16(L)F1825 only) SDA1 (PIC16(L)F1825 only) P1C ⁽²⁾
RC2	SDO1 ⁽²⁾ (PIC16(L)F1825 only) P1D ⁽²⁾ P2B ⁽²⁾
RC3	SS1 ⁽²⁾ (PIC16(L)F1825 only) CCP2 ⁽²⁾ P1C ⁽²⁾ P2A ⁽²⁾
RC4	MDOUT SRNQ C2OUT TX ⁽²⁾ /CK ⁽²⁾ P1B
RC5	RX ⁽²⁾ /DT ⁽²⁾ CCP1/P1A
RC6 ⁽³⁾	SS1 CCP4
RC7 ⁽³⁾	SDO1

Note 1: Priority listed from highest to lowest.

Pin function is selectable via the APFCON0 or APFCON1 register.

3: PIC16(L)F1829 only.

REGISTER 12-15: PORTC: PORTC REGISTER

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
RC7 ⁽¹⁾	RC6 ⁽¹⁾	RC5	RC4	RC3	RC2	RC1	RC0
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

bit 7-0 RC<7:0>: PORTC General Purpose I/O Pin bits⁽¹⁾

1 = Port pin is \geq VIH 0 = Port pin is \leq VIL

Note 1: RC<7:6> available on PIC16(L)F1829 only. Otherwise, they are unimplemented and read as '0'.

REGISTER 12-16: TRISC: PORTC TRI-STATE REGISTER

R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
TRISC7 ⁽¹⁾	TRISC6 ⁽¹⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

bit 7-0 TRISC<7:0>: PORTC Tri-State Control bits⁽¹⁾

1 = PORTC pin configured as an input (tri-stated)

0 = PORTC pin configured as an output

Note 1: TRISC<7:6> available on PIC16(L)F1829 only. Otherwise, they are unimplemented and read as '0'.

REGISTER 12-17: LATC: PORTC DATA LATCH REGISTER

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
LATC7 ⁽²⁾	LATC6 ⁽²⁾	LATC5	LATC4	LATC3	LATC2	LATC1	LATC0
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

bit 7-0 LATC<7:0>: PORTC Output Latch Value bits^(1, 2)

Note 1: Writes to PORTC are actually written to corresponding LATC register. Reads from PORTC register is return of actual I/O pin values.

2: LATC<7:6> available on PIC16(L)F1829 only. Otherwise, they are unimplemented and read as '0'.

REGISTER 12-18: ANSELC: PORTC ANALOG SELECT REGISTER

R/W-1/1	R/W-1/1	U-0	U-0	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
ANSC7 ⁽²⁾	ANSC6 ⁽²⁾	_	_	ANSC3	ANSC2	ANSC1	ANSC0
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7-6 ANSC<7:6>: Analog Select between Analog or Digital Function on pins RC<7:6>, respectively(2)

0 = Digital I/O. Pin is assigned to port or digital special function.

1 = Analog input. Pin is assigned as analog input $^{(1)}$. Digital input buffer disabled.

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

bit 3-0 ANSC<3:0>: Analog Select between Analog or Digital Function on pins RC<3:0>, respectively

0 = Digital I/O. Pin is assigned to port or digital special function.

1 = Analog input. Pin is assigned as analog input (1). Digital input buffer disabled.

Note 1: When setting a pin to an analog input, the corresponding TRIS bit must be set to Input mode in order to allow external control of the voltage on the pin.

2: ANSELC<7:6> available on PIC16(L)F1829 only. Otherwise, they are unimplemented and read as '0'.

REGISTER 12-19: WPUC: WEAK PULL-UP PORTC REGISTER

R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
WPUC7 ⁽³⁾	WPUC6 ⁽³⁾	WPUC5	WPUC4	WPUC3	WPUC2	WPUC1	WPUC0
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7-0 WPUC<7:0>: Weak Pull-up Register bits^(1, 2)

1 = Pull-up enabled

0 = Pull-up disabled

Note 1: Global WPUEN bit of the OPTION_REG register must be cleared for individual pull-ups to be enabled.

2: The weak pull-up device is automatically disabled if the pin is in configured as an output.

3: WPUC<7:6> available on PIC16(L)F1829 only. Otherwise, they are unimplemented and read as '0'.

REGISTER 12-20: INLVLC: PORTC INPUT LEVEL CONTROL REGISTER

| _ | _ | R/W-0/0 ⁽³⁾ |
|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|
| R/W-1/1 ⁽²⁾ |
| INLVLC7 ⁽¹⁾ | INLVLC6 ⁽¹⁾ | INLVLC5 | INLVLC4 | INLVLC3 | INLVLC2 | INLVLC1 | INLVLC0 |
| bit 7 | | | | | | | bit 0 |

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

bit 7-0 INLVLC<7:0>: PORTC Input Level Select bits⁽¹⁾

For RC<7:0> pins, respectively

1 = ST input used for port reads and interrupt-on-change

0 = TTL input used for port reads and interrupt-on-change

Note 1: INLVLC<7:6> available on PIC16(L)F1829 only. Otherwise, they are unimplemented and read as '0'.

2: PIC16(L)F1829 only, reset default value.

3: PIC16(L)F1825 only, reset default value.

TABLE 12-8: SUMMARY OF REGISTERS ASSOCIATED WITH PORTC

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ANSELC	ANSC7 ⁽¹⁾	ANSC6 ⁽¹⁾	_	_	ANSC3	ANSC2	ANSC1	ANSC0	132
INLVLC	INLVLC7 ⁽¹⁾	INLVLC6 ⁽¹⁾	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	138
LATC	LATC7 ⁽¹⁾	LATC6 ⁽¹⁾	LATC5	LATC4	LATC3	LATC2	LATC1	LATC0	131
PORTC	RC7 ⁽¹⁾	RC6 ⁽¹⁾	RC5	RC4	RC3	RC2	RC1	RC0	131
TRISC	TRISC7 ⁽¹⁾	TRISC6 ⁽¹⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	131
WPUC	WPUC7 ⁽¹⁾	WPUC6 ⁽¹⁾	WPUC5	WPUC4	WPUC3	WPUC2	WPUC1	WPUC0	132

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by PORTC.

Note 1: PIC16(L)F1829 only.

13.0 INTERRUPT-ON-CHANGE

The PORTA pins can be configured to operate as Interrupt-on-Change (IOC) pins. On the PIC16(L)F1829 devices, the PORTB pins can also be configured to operate as IOC pins. An interrupt can be generated by detecting a signal that has either a rising edge or a falling edge. Any individual port pin, or combination of port pins, can be configured to generate an interrupt. The interrupt-on-change module has the following features:

- · Interrupt-on-change enable (Master Switch)
- Individual pin configuration
- · Rising and falling edge detection
- · Individual pin interrupt flags

Figure 13-1 is a block diagram of the IOC module.

13.1 Enabling the Module

To allow individual port pins to generate an interrupt, the IOCIE bit of the INTCON register must be set. If the IOCIE bit is disabled, the edge detection on the pin will still occur, but an interrupt will not be generated.

13.2 Individual Pin Configuration

For each port pin, a rising edge detector and a falling edge detector are present. To enable a pin to detect a rising edge, the associated bit of the IOCxP register is set. To enable a pin to detect a falling edge, the associated bit of the IOCxN register is set.

A pin can be configured to detect rising and falling edges simultaneously by setting both associated bits of the IOCxP and IOCxN registers, respectively.

13.3 Interrupt Flags

The IOCAFx and IOCBFx bits located in the IOCAF and IOCBF registers, respectively, are status flags that correspond to the interrupt-on-change pins of the associated port. If an expected edge is detected on an appropriately enabled pin, then the status flag for that pin will be set, and an interrupt will be generated if the IOCIE bit is set. The IOCIF bit of the INTCON register reflects the status of all IOCAFx and IOCBFx bits.

13.4 Clearing Interrupt Flags

The individual status flags, (IOCAFx and IOCBFx bits), can be cleared by resetting them to zero. If another edge is detected during this clearing operation, the associated status flag will be set at the end of the sequence, regardless of the value actually being written.

In order to ensure that no detected edge is lost while clearing flags, only AND operations masking out known changed bits should be performed. The following sequence is an example of what should be performed.

EXAMPLE 13-1: CLEARING INTERRUPT FLAGS (PORTA EXAMPLE)

MOVLW 0xff XORWF IOCAF, W ANDWF IOCAF, F

13.5 Operation in Sleep

The interrupt-on-change interrupt sequence will wake the device from Sleep mode, if the IOCIE bit is set.

If an edge is detected while in Sleep mode, the IOCxF register will be updated prior to the first instruction executed out of Sleep.

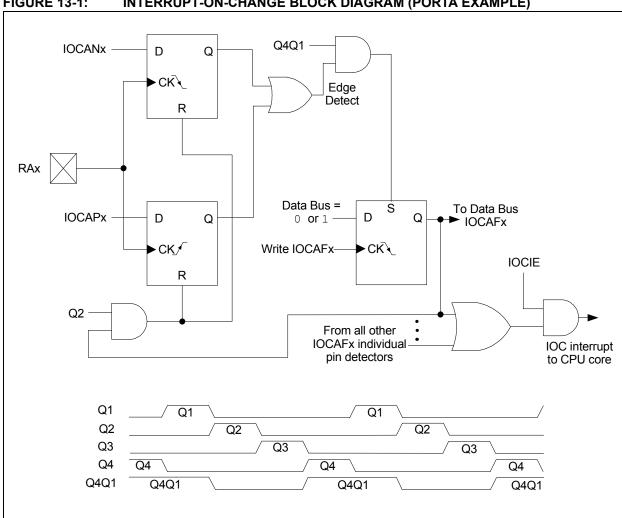


FIGURE 13-1: INTERRUPT-ON-CHANGE BLOCK DIAGRAM (PORTA EXAMPLE)

13.6 **Interrupt-on-Change Registers**

REGISTER 13-1: IOCAP: INTERRUPT-ON-CHANGE PORTA POSITIVE EDGE REGISTER

U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
_		IOCAP5	IOCAP4	IOCAP3	IOCAP2	IOCAP1	IOCAP0
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7-6 Unimplemented: Read as '0'

bit 5-0 IOCAP<5:0>: Interrupt-on-Change PORTA Positive Edge Enable bits

- 1 = Interrupt-on-change enabled on the pin for a positive going edge. Associated Status bit and interrupt flag will be set upon detecting an edge.
- 0 = Interrupt-on-change disabled for the associated pin.

REGISTER 13-2: IOCAN: INTERRUPT-ON-CHANGE PORTA NEGATIVE EDGE REGISTER

U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
_	_	IOCAN5	IOCAN4	IOCAN3	IOCAN2	IOCAN1	IOCAN0
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

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

bit 5-0 IOCAN<5:0>: Interrupt-on-Change PORTA Negative Edge Enable bits

1 = Interrupt-on-change enabled on the pin for a negative going edge. Associated Status bit and interrupt flag will be set upon detecting an edge.

0 = Interrupt-on-change disabled for the associated pin.

REGISTER 13-3: IOCAF: INTERRUPT-ON-CHANGE PORTA FLAG REGISTER

U-0	U-0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0
_	_	IOCAF5	IOCAF4	IOCAF3	IOCAF2	IOCAF1	IOCAF0
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared HS - Bit is set in hardware

bit 7-6 Unimplemented: Read as '0'

bit 5-0 **IOCAF<5:0>:** Interrupt-on-Change PORTA Flag bits

1 = An enabled change was detected on the associated pin.

Set when IOCAPx = 1 and a rising edge was detected on RAx, or when IOCANx = 1 and a falling edge was

detected on RAx.

0 = No change was detected, or the user cleared the detected change.

REGISTER 13-4: IOCBP: INTERRUPT-ON-CHANGE PORTB POSITIVE EDGE REGISTER (PIC16(L)F1829 ONLY)

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	U-0	U-0	U-0
IOCBP7	IOCBP6	IOCBP5	IOCBP4	_	_	_	_
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

bit 7-4 IOCBP<7:4>: Interrupt-on-Change PORTB Positive Edge Enable bits

1 = Interrupt-on-change enabled on the pin for a positive going edge. Associated Status bit and interrupt flag will be set upon detecting an edge.

0 = Interrupt-on-change disabled for the associated pin.

bit 3-0 Unimplemented: Read as '0'

REGISTER 13-5: IOCBN: INTERRUPT-ON-CHANGE PORTB NEGATIVE EDGE REGISTER (PIC16(L)F1829 ONLY)

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	U-0	U-0	U-0
IOCBN7	IOCBN6	IOCBN5	IOCBN4	_	_	_	_
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

bit 7-4 IOCAN<7:4>: Interrupt-on-Change PORTB Negative Edge Enable bits

1 = Interrupt-on-change enabled on the pin for a negative going edge. Associated Status bit and interrupt flag will be set upon detecting an edge.

0 = Interrupt-on-change disabled for the associated pin.

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

REGISTER 13-6: IOCBF: INTERRUPT-ON-CHANGE PORTB FLAG REGISTER (PIC16(L)F1829 ONLY)

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	U-0	U-0	U-0
IOCBF7	IOCBF6	IOCBF5	IOCBF4	_	_	_	_
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared HS - Bit is set in hardware

bit 7-4 **IOCBF<7:4>:** Interrupt-on-Change PORTB Flag bits

1 = An enabled change was detected on the associated pin. Set when IOCBPx = 1 and a rising edge was detected on RAx, or when IOCANx = 1 and a falling edge was detected on RBx.

0 = No change was detected, or the user cleared the detected change.

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

TABLE 13-1: SUMMARY OF REGISTERS ASSOCIATED WITH INTERRUPT-ON-CHANGE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ANSELA	_	-	-	ANSA4	_	ANSA2	ANSA1	ANSA0	126
ANSELB ⁽¹⁾	ANSB7	ANSB6	ANSB5	ANSB4	_	_	_	_	132
INLVLA	_	_	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	127
INLVLB ⁽¹⁾	INLVLB7	INLVLB6	INLVLB5	INLVLB4	_	_	_	_	132
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	90
IOCAF	_	_	IOCAF5	IOCAF4	IOCAF3	IOCAF2	IOCAF1	IOCAF0	141
IOCAN	_	_	IOCAN5	IOCAN4	IOCAN3	IOCAN2	IOCAN1	IOCAN0	141
IOCAP	_	_	IOCAP5	IOCAP4	IOCAP3	IOCAP2	IOCAP1	IOCAP0	140
IOCBF ⁽¹⁾	IOCBF7	IOCBF6	IOCBF5	IOCBF4	_	_	_	_	142
IOCBN ⁽¹⁾	IOCBN7	IOCBN6	IOCBN5	IOCBN4	_	_	_	_	142
IOCBP ⁽¹⁾	IOCBP7	IOCBP6	IOCBP5	IOCBP4	_	_	_	_	141
TRISA	_	_	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	125
TRISB ⁽¹⁾	TRISB7	TRISB6	TRISB5	TRISB4	_	_	_	_	131

Legend: — Unimplemented location, read as '0'. Shaded cells are not used by interrupt-on-change.

Note 1: PIC16(L)F1829 only.

NOTES:

14.0 FIXED VOLTAGE REFERENCE (FVR)

The Fixed Voltage Reference, or FVR, is a stable voltage reference, independent of VDD, with 1.024V, 2.048V or 4.096V selectable output levels. The output of the FVR can be configured to supply a reference voltage to the following:

- · ADC input channel
- · ADC positive reference
- · Comparator positive input
- Digital-to-Analog Converter (DAC)
- · Capacitive Sensing (CPS) module

The FVR can be enabled by setting the FVREN bit of the FVRCON register.

14.1 Independent Gain Amplifiers

The output of the FVR supplied to the ADC, Comparators, DAC and CPS are routed through two independent programmable gain amplifiers. Each amplifier can be configured to amplify the reference voltage by 1x, 2x or 4x, to produce the three possible voltage levels.

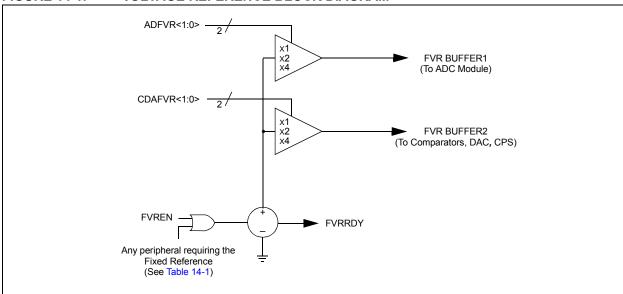
The ADFVR<1:0> bits of the FVRCON register are used to enable and configure the gain amplifier settings for the reference supplied to the ADC module. Reference Section 16.0 "Analog-to-Digital Converter (ADC) Module" for additional information.

The CDAFVR<1:0> bits of the FVRCON register are used to enable and configure the gain amplifier settings for the reference supplied to the DAC, CPS and comparator module. Reference Section 17.0 "Digital-to-Analog Converter (DAC) Module" and Section 19.0 "Comparator Module" for additional information.

14.2 FVR Stabilization Period

When the Fixed Voltage Reference module is enabled, it requires time for the reference and amplifier circuits to stabilize. Once the circuits stabilize and are ready for use, the FVRRDY bit of the FVRCON register will be set. See Section 30.0 "Electrical Specifications" for the minimum delay requirement.

FIGURE 14-1: VOLTAGE REFERENCE BLOCK DIAGRAM



14.3 FVR Control Registers

REGISTER 14-1: FVRCON: FIXED VOLTAGE REFERENCE CONTROL REGISTER

R/W-0/0	R-q/q	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
FVREN	FVRRDY ⁽¹⁾	TSEN	TSRNG	CDAFVR<1:0>		ADFV	R<1:0>
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	q = Value depends on condition

bit 7	FVREN: Fixed Voltage Reference Enable bit 0 = Fixed Voltage Reference is disabled 1 = Fixed Voltage Reference is enabled
bit 6	FVRRDY: Fixed Voltage Reference Ready Flag bit ⁽¹⁾ 0 = Fixed Voltage Reference output is not ready or not enabled 1 = Fixed Voltage Reference output is ready for use
bit 5	TSEN: Temperature Indicator Enable bit 0 = Temperature indicator is disabled 1 = Temperature indicator is enabled
bit 4	TSRNG: Temperature Indicator Range Selection bit ⁽³⁾ 0 = VOUT = VDD - 2VT (Low Range) 1 = VOUT = VDD - 4VT (High Range)
bit 3-2	CDAFVR<1:0>: Comparator and DAC Fixed Voltage Reference Selection bit 00 = Comparator and DAC Fixed Voltage Reference Peripheral output is off 01 = Comparator and DAC Fixed Voltage Reference Peripheral output is 1x (1.024V) 10 = Comparator and DAC Fixed Voltage Reference Peripheral output is 2x (2.048V) 11 = Comparator and DAC Fixed Voltage Reference Peripheral output is 4x (4.096V) 12
bit 1-0	ADFVR<1:0>: ADC Fixed Voltage Reference Selection bit 00 = ADC Fixed Voltage Reference Peripheral output is off 01 = ADC Fixed Voltage Reference Peripheral output is 1x (1.024V) 10 = ADC Fixed Voltage Reference Peripheral output is 2x (2.048V) ⁽²⁾ 11 = ADC Fixed Voltage Reference Peripheral output is 4x (4.096V) ⁽²⁾

Note 1: FVRRDY is always '1' on devices with the LDO (PIC16F1825/1829).

2: Fixed Voltage Reference output cannot exceed VDD.

3: See Section 15.0 "Temperature Indicator Module" for additional information.

TABLE 14-1: SUMMARY OF REGISTERS ASSOCIATED WITH THE FIXED VOLTAGE REFERENCE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDAFVR<1:0>		ADFVF	R<1:0>	146

Legend: Shaded cells are unused by the Fixed Voltage Reference module.

15.0 TEMPERATURE INDICATOR MODULE

This family of devices is equipped with a temperature circuit designed to measure the operating temperature of the silicon die. The circuit's range of operating temperature falls between -40°C and +85°C. The output is a voltage that is proportional to the device temperature. The output of the temperature indicator is internally connected to the device ADC.

The circuit may be used as a temperature threshold detector or a more accurate temperature indicator, depending on the level of calibration performed. A one-point calibration allows the circuit to indicate a temperature closely surrounding that point. A two-point calibration allows the circuit to sense the entire range of temperature more accurately. Reference Application Note AN1333, "Use and Calibration of the Internal Temperature Indicator" (DS01333) for more details regarding the calibration process.

15.1 Circuit Operation

Figure 15-1 shows a simplified block diagram of the temperature circuit. The proportional voltage output is achieved by measuring the forward voltage drop across multiple silicon junctions.

Equation 15-1 describes the output characteristics of the temperature indicator.

EQUATION 15-1: VOUT RANGES

High Range: Vout = VDD - 4VT

Low Range: Vout = VDD - 2VT

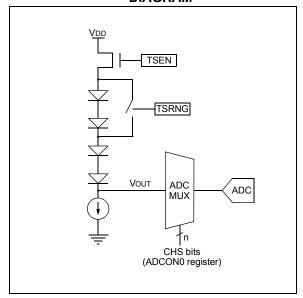
The temperature sense circuit is integrated with the Fixed Voltage Reference (FVR) module. See Section 14.0 "Fixed Voltage Reference (FVR)" for more information.

The circuit is enabled by setting the TSEN bit of the FVRCON register. When disabled, the circuit draws no current.

The circuit operates in either high or low range. The high range, selected by setting the TSRNG bit of the FVRCON register, provides a wider output voltage. This provides more resolution over the temperature range, but may be less consistent from part to part. This range requires a higher bias voltage to operate and thus, a higher VDD is needed.

The low range is selected by clearing the TSRNG bit of the FVRCON register. The low range generates a lower voltage drop and thus, a lower bias voltage is needed to operate the circuit. The low range is provided for low voltage operation.

FIGURE 15-1: TEMPERATURE CIRCUIT DIAGRAM



15.2 Minimum Operating VDD vs. Minimum Sensing Temperature

When the temperature circuit is operated in low range, the device may be operated at any operating voltage that is within specifications.

When the temperature circuit is operated in high range, the device operating voltage, VDD, must be high enough to ensure that the temperature circuit is correctly biased.

Table 15-1 shows the recommended minimum VDD vs. range setting.

TABLE 15-1: RECOMMENDED VDD VS. RANGE

Min. VDD, TSRNG = 1	Min. VDD, TSRNG = 0
3.6V	1.8V

15.3 Temperature Output

The output of the circuit is measured using the internal Analog-to-Digital converter. A channel is reserved for the temperature circuit output. Refer to Section 16.0 "Analog-to-Digital Converter (ADC) Module" for detailed information.

15.4 ADC Acquisition Time

To ensure accurate temperature measurements, the user must wait at least 200 μs after the ADC input multiplexer is connected to the temperature indicator output before the conversion is performed. In addition, the user must wait 200 μs between sequential conversions of the temperature indicator output.

NOTES:

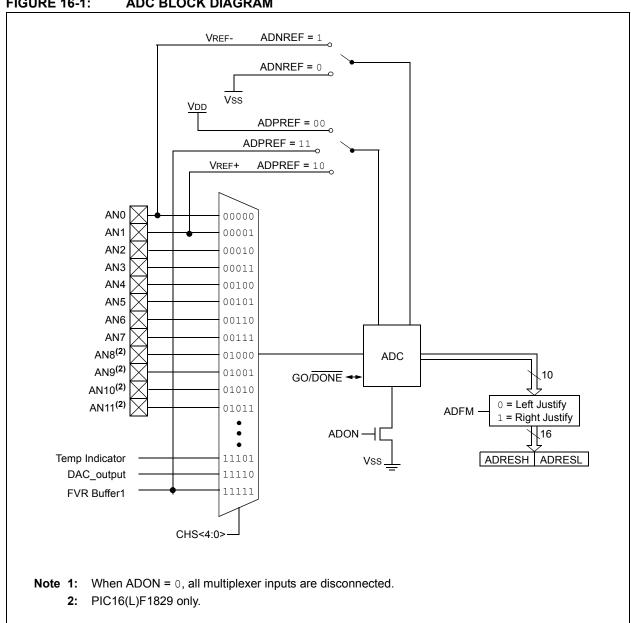
16.0 ANALOG-TO-DIGITAL **CONVERTER (ADC) MODULE**

The Analog-to-Digital Converter (ADC) allows conversion of an analog input signal to a 10-bit binary representation of that signal. This device uses analog inputs, which are multiplexed into a single sample and hold circuit. The output of the sample and hold is connected to the input of the converter. The converter generates a 10-bit binary result via successive approximation and stores the conversion result into the ADC result registers (ADRESH:ADRESL register pair). Figure 16-1 shows the block diagram of the ADC.

The ADC voltage reference is software selectable to be either internally generated or externally supplied.

The ADC can generate an interrupt upon completion of a conversion. This interrupt can be used to wake-up the device from Sleep.

FIGURE 16-1: ADC BLOCK DIAGRAM



16.1 ADC Configuration

When configuring and using the ADC the following functions must be considered:

- · Port configuration
- · Channel selection
- · ADC voltage reference selection
- · ADC conversion clock source
- · Interrupt control
- · Result formatting

16.1.1 PORT CONFIGURATION

The ADC can be used to convert both analog and digital signals. When converting analog signals, the I/O pin should be configured for analog by setting the associated TRIS and ANSEL bits. Refer to Section 12.0 "I/O Ports" for more information.

Note: Analog voltages on any pin that is defined as a digital input may cause the input buffer to conduct excess current.

16.1.2 CHANNEL SELECTION

There are up to 15 channel selections available:

- AN<7:0> pins (PIC16(L)F1825 only)
- AN<11:0> pins (PIC16(L)F1829 only)
- · Temperature Indicator
- · DAC_output
- FVR Buffer1

Refer to Section 17.0 "Digital-to-Analog Converter (DAC) Module", Section 14.0 "Fixed Voltage Reference (FVR)" and Section 15.0 "Temperature Indicator Module" for more information on these channel selections.

The CHS bits of the ADCON0 register determine which channel is connected to the sample and hold circuit.

When changing channels, a delay is required before starting the next conversion. Refer to **Section 16.2** "**ADC Operation**" for more information.

16.1.3 ADC VOLTAGE REFERENCE

The ADPREF bits of the ADCON1 register provides control of the positive voltage reference. The positive voltage reference can be:

- VREF+ pin
- VDD
- FVR 2.048V
- FVR 4.096V (Not available on LF devices)

The ADNREF bits of the ADCON1 register provides control of the negative voltage reference. The negative voltage reference can be:

- · VREF- pin
- Vss

See Section 14.0 "Fixed Voltage Reference (FVR)" for more details on the Fixed Voltage Reference.

16.1.4 CONVERSION CLOCK

The source of the conversion clock is software selectable via the ADCS bits of the ADCON1 register. There are seven possible clock options:

- Fosc/2
- Fosc/4
- Fosc/8
- Fosc/16
- Fosc/32
- Fosc/64
- · FRC (dedicated internal oscillator)

The time to complete one bit conversion is defined as TAD. One full 10-bit conversion requires 11.5 TAD periods as shown in Figure 16-2.

For correct conversion, the appropriate TAD specification must be met. Refer to the A/D conversion requirements in **Section 30.0 "Electrical Specifications"** for more information. Table 16-1 gives examples of appropriate ADC clock selections.

Note: Unless using the FRC, any changes in the system clock frequency will change the ADC clock frequency, which may adversely affect the ADC result.

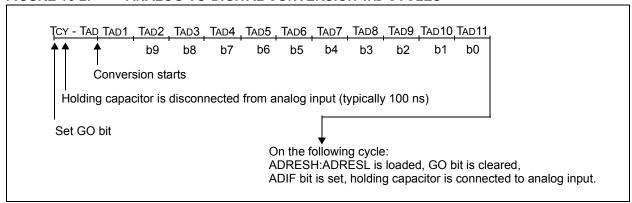
TABLE 16-1: ADC CLOCK PERIOD (TAD) Vs. DEVICE OPERATING FREQUENCIES

ADC Clock P	eriod (TAD)	Device Frequency (Fosc)						
ADC Clock Source	ADCS<2:0>	32 MHz	20 MHz	16 MHz	8 MHz	4 MHz	1 MHz	
Fosc/2	000	62.5ns ⁽²⁾	100 ns ⁽²⁾	125 ns ⁽²⁾	250 ns ⁽²⁾	500 ns ⁽²⁾	2.0 μs	
Fosc/4	100	125 ns ⁽²⁾	200 ns ⁽²⁾	250 ns ⁽²⁾	500 ns ⁽²⁾	1.0 μs	4.0 μs	
Fosc/8	001	0.5 μs ⁽²⁾	400 ns ⁽²⁾	0.5 μs ⁽²⁾	1.0 μs	2.0 μs	8.0 μs ⁽³⁾	
Fosc/16	101	800 ns	800 ns	1.0 μs	2.0 μs	4.0 μs	16.0 μs ⁽³⁾	
Fosc/32	010	1.0 μs	1.6 μs	2.0 μs	4.0 μs	8.0 μs ⁽³⁾	32.0 μs ⁽³⁾	
Fosc/64	110	2.0 μs	3.2 μs	4.0 μs	8.0 μs ⁽³⁾	16.0 μs ⁽³⁾	64.0 μs ⁽³⁾	
FRC	x11	1.0-6.0 μs ^(1,4)						

Legend: Shaded cells are outside of recommended range.

- Note 1: The FRC source has a typical TAD time of 1.6 μs for VDD.
 - 2: These values violate the minimum required TAD time.
 - **3:** For faster conversion times, the selection of another clock source is recommended.
 - 4: The ADC clock period (TAD) and total ADC conversion time can be minimized when the ADC clock is derived from the system clock Fosc. However, the FRC clock source must be used when conversions are to be performed with the device in Sleep mode.

FIGURE 16-2: ANALOG-TO-DIGITAL CONVERSION TAD CYCLES



16.1.5 INTERRUPTS

The ADC module allows for the ability to generate an interrupt upon completion of an Analog-to-Digital conversion. The ADC Interrupt Flag is the ADIF bit in the PIR1 register. The ADC Interrupt Enable is the ADIE bit in the PIE1 register. The ADIF bit must be cleared in software.

- **Note 1:** The ADIF bit is set at the completion of every conversion, regardless of whether or not the ADC interrupt is enabled.
 - **2:** The ADC operates during Sleep only when the FRC oscillator is selected.

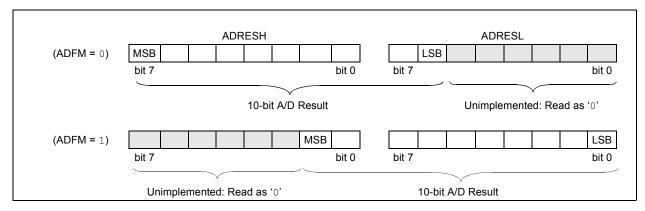
This interrupt can be generated while the device is operating or while in Sleep. If the device is in Sleep, the interrupt will wake-up the device. Upon waking from Sleep, the next instruction following the SLEEP instruction is always executed. If the user is attempting to wake-up from Sleep and resume in-line code execution, the GIE and PEIE bits of the INTCON register must be disabled. If the GIE and PEIE bits of the INTCON register are enabled, execution will switch to the Interrupt Service Routine.

16.1.6 RESULT FORMATTING

The 10-bit A/D conversion result can be supplied in two formats, left justified or right justified. The ADFM bit of the ADCON1 register controls the output format.

Figure 16-3 shows the two output formats.

FIGURE 16-3: 10-BIT A/D CONVERSION RESULT FORMAT



16.2 ADC Operation

16.2.1 STARTING A CONVERSION

To enable the ADC module, the ADON bit of the ADCON0 register must be set to a '1'. Setting the GO/DONE bit of the ADCON0 register to a '1' will start the Analog-to-Digital conversion.

Note: The GO/DONE bit should not be set in the same instruction that turns on the ADC. Refer to Section 16.2.6 "A/D Conversion Procedure".

16.2.2 COMPLETION OF A CONVERSION

When the conversion is complete, the ADC module will:

- Clear the GO/DONE bit
- · Set the ADIF Interrupt Flag bit
- Update the ADRESH and ADRESL registers with new conversion result

16.2.3 TERMINATING A CONVERSION

If a conversion must be terminated before completion, the GO/DONE bit can be cleared in software. The ADRESH and ADRESL registers will be updated with the partially complete Analog-to-Digital conversion sample. Incomplete bits will match the last bit converted.

Note: A device Reset forces all registers to their Reset state. Thus, the ADC module is turned off and any pending conversion is terminated.

16.2.4 ADC OPERATION DURING SLEEP

The ADC module can operate during Sleep. This requires the ADC clock source to be set to the FRC option. When the FRC clock source is selected, the ADC waits one additional instruction before starting the conversion. This allows the SLEEP instruction to be executed, which can reduce system noise during the conversion. If the ADC interrupt is enabled, the device will wake-up from Sleep when the conversion completes. If the ADC interrupt is disabled, the ADC module is turned off after the conversion completes, although the ADON bit remains set.

When the ADC clock source is something other than FRC, a SLEEP instruction causes the present conversion to be aborted and the ADC module is turned off, although the ADON bit remains set.

16.2.5 SPECIAL EVENT TRIGGER

The Special Event Trigger of the CCPx/ECCPX module allows periodic ADC measurements without software intervention. When this trigger occurs, the GO/DONE bit is set by hardware and the Timer1 counter resets to zero.

TABLE 16-2: SPECIAL EVENT TRIGGER

Device	CCPx/ECCPx
PIC16(L)F1825/1829	CCP4

Using the Special Event Trigger does not assure proper ADC timing. It is the user's responsibility to ensure that the ADC timing requirements are met.

Refer to Section 24.0 "Capture/Compare/PWM Modules" for more information.

16.2.6 A/D CONVERSION PROCEDURE

This is an example procedure for using the ADC to perform an Analog-to-Digital conversion:

- 1. Configure Port:
 - Disable pin output driver (Refer to the TRIS register)
 - Configure pin as analog (Refer to the ANSEL register)
- 2. Configure the ADC module:
 - · Select ADC conversion clock
 - · Configure voltage reference
 - · Select ADC input channel
 - · Turn on ADC module
- 3. Configure ADC interrupt (optional):
 - · Clear ADC interrupt flag
 - · Enable ADC interrupt
 - · Enable peripheral interrupt
 - Enable global interrupt⁽¹⁾
- 4. Wait the required acquisition time⁽²⁾.
- Start conversion by setting the GO/DONE bit.
- 6. Wait for ADC conversion to complete by one of the following:
 - Polling the GO/DONE bit
 - Waiting for the ADC interrupt (interrupts enabled)
- 7. Read ADC Result.
- 8. Clear the ADC interrupt flag (required if interrupt is enabled).
 - **Note 1:** The global interrupt can be disabled if the user is attempting to wake-up from Sleep and resume in-line code execution.
 - 2: Refer to Section 16.3 "A/D Acquisition Requirements".

EXAMPLE 16-1: A/D CONVERSION

```
; This code block configures the ADC
; for polling, Vdd and Vss references, Frc
; clock and ANO input.
;Conversion start & polling for completion
; are included.
BANKSEL ADCON1
MOVLW B'11110000' ; Right justify, Frc
                   ;clock
MOVWF
       ADCON1
                   ; Vdd and Vss Vref
BANKSEL TRISA
BSF
       TRISA, 0 ; Set RAO to input
BANKSEL ANSEL
        ANSEL, 0 ;Set RAO to analog
BSF
BANKSEL
        ADCON0
        B'00000001' ;Select channel ANO
MOVLW
MOVWF
        ADCON0
                    ;Turn ADC On
        SampleTime ; Acquisiton delay
CALL
       ADCONO, ADGO ; Start conversion
BSF
BTFSC ADCON0, ADGO ; Is conversion done?
GOTO
        $-1
                 ;No, test again
BANKSEL ADRESH
MOVF
        ADRESH,W ;Read upper 2 bits
        RESULTHI ;store in GPR space
MOVWF
BANKSEL
        ADRESL
        ADRESL, W
MOVF
                   ;Read lower 8 bits
        RESULTLO ;Store in GPR space
MOVWF
```

16.2.7 ADC REGISTER DEFINITIONS

The following registers are used to control the operation of the ADC.

REGISTER 16-1: ADCON0: A/D CONTROL REGISTER 0

U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
_			CHS<4:0>			GO/DONE	ADON
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

```
bit 7
               Unimplemented: Read as '0'
bit 6-2
               CHS<4:0>: Analog Channel Select bits
               00000 = ANO
               00001 = AN1
               00010 = AN2
               00011 = AN3
               00100 = AN4
               00101 = AN5
               00110 = AN6
               00111 = AN7
               01000 = AN8<sup>(1)</sup>
               01001 = AN9^{(1)}
               01010 = AN10^{(1)}
               01011 = AN11^{(1)}
               01100 = Reserved. No channel connected.
               11100 = Reserved. No channel connected.
               11101 = Temperature Indicator<sup>(4)</sup>
               11110 = DAC output<sup>(2)</sup>
               11111 = FVR (Fixed Voltage Reference) Output (3)
               GO/DONE: A/D Conversion Status bit
bit 1
               1 = A/D conversion cycle in progress. Setting this bit starts an A/D conversion cycle.
                   This bit is automatically cleared by hardware when the A/D conversion has completed.
               0 = A/D conversion completed/not in progress
bit 0
               ADON: ADC Enable bit
               1 = ADC is enabled
               0 = ADC is disabled and consumes no operating current
Note 1: PIC16(L)F1829 only.
```

2: See Section 17.0 "Digital-to-Analog Converter (DAC) Module"for more information.

3: See Section 14.0 "Fixed Voltage Reference (FVR)" for more information.

4: See Section 15.0 "Temperature Indicator Module" for more information.

REGISTER 16-2: ADCON1: A/D CONTROL REGISTER 1

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	R/W-0/0	R/W-0/0	R/W-0/0
ADFM	ADCS<2:0>			_	ADNREF	ADPRE	F<1:0>
bit 7							bit 0

Legend:R = Readable bitW = Writable bitU = Unimplemented bit, read as '0'u = Bit is unchangedx = Bit is unknown-n/n = Value at POR and BOR/Value at all other Resets'1' = Bit is set'0' = Bit is cleared

bit 7 ADFM: A/D Result Format Select bit

1 = Right justified. Six Most Significant bits of ADRESH are set to '0' when the conversion result is loaded.

0 = Left justified. Six Least Significant bits of ADRESL are set to '0' when the conversion result is loaded.

bit 6-4 ADCS<2:0>: A/D Conversion Clock Select bits

000 = Fosc/2

001 = Fosc/8

010 = Fosc/32

011 = FRC (clock supplied from a dedicated RC oscillator)

100 = Fosc/4

101 = Fosc/16

110 = Fosc/64

111 = FRC (clock supplied from a dedicated RC oscillator)

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

bit 2 ADNREF: A/D Negative Voltage Reference Configuration bit

0 = VREF- is connected to Vss

1 = VREF- is connected to external VREF- pin⁽¹⁾

bit 1-0 ADPREF<1:0>: A/D Positive Voltage Reference Configuration bits

00 = VREF+ is connected to VDD

01 = Reserved

10 = VREF+ is connected to external VREF+ pin⁽¹⁾

11 = VREF+ is connected to internal Fixed Voltage Reference (FVR) module⁽¹⁾

Note 1: When selecting the FVR or the VREF+ pin as the source of the positive reference, be aware that a minimum voltage specification exists. See **Section 30.0 "Electrical Specifications"** for details.

REGISTER 16-3: ADRESH: ADC RESULT REGISTER HIGH (ADRESH) ADFM = 0

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u		
ADRES<9:2>									
bit 7							bit 0		

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

bit 7-0 **ADRES<9:2>**: ADC Result Register bits Upper eight bits of 10-bit conversion result

REGISTER 16-4: ADRESL: ADC RESULT REGISTER LOW (ADRESL) ADFM = 0

| R/W-x/u |
|---------|---------|---------|---------|---------|---------|---------|---------|
| ADRES | S<1:0> | _ | _ | _ | _ | _ | _ |
| bit 7 | | | | | | | bit 0 |

Legend:R = Readable bitW = Writable bitU = Unimplemented bit, read as '0'u = Bit is unchangedx = Bit is unknown-n/n = Value at POR and BOR/Value at all other Resets'1' = Bit is set'0' = Bit is cleared

bit 7-6 **ADRES<1:0>**: ADC Result Register bits

Lower two bits of 10-bit conversion result

bit 5-0 **Reserved**: Do not use.

REGISTER 16-5: ADRESH: ADC RESULT REGISTER HIGH (ADRESH) ADFM = 1

| R/W-x/u |
|---------|---------|---------|---------|---------|---------|---------|---------|
| _ | _ | _ | _ | _ | _ | ADRES | S<9:8> |
| bit 7 | | | | | | | bit 0 |

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets '1' = Bit is set '0' = Bit is cleared

bit 7-2 **Reserved**: Do not use.

bit 1-0 ADRES<9:8>: ADC Result Register bits

Upper two bits of 10-bit conversion result

REGISTER 16-6: ADRESL: ADC RESULT REGISTER LOW (ADRESL) ADFM = 1

| R/W-x/u |
|---------|---------|---------|---------|---------|---------|---------|---------|
| | | | ADRES | S<7:0> | | | |
| bit 7 | | | | | | | bit 0 |

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

bit 7-0 ADRES<7:0>: ADC Result Register bits

Lower eight bits of 10-bit conversion result

16.3 A/D Acquisition Requirements

For the ADC to meet its specified accuracy, the charge holding capacitor (Chold) must be allowed to fully charge to the input channel voltage level. The Analog Input model is shown in Figure 16-4. The source impedance (Rs) and the internal sampling switch (Rss) impedance directly affect the time required to charge the capacitor Chold. The sampling switch (Rss) impedance varies over the device voltage (VDD), refer to Figure 16-4. The maximum recommended impedance for analog sources is 10 k Ω . As the

source impedance is decreased, the acquisition time may be decreased. After the analog input channel is selected (or changed), an A/D acquisition must be done before the conversion can be started. To calculate the minimum acquisition time, Equation 16-1 may be used. This equation assumes that 1/2 LSb error is used (1,024 steps for the ADC). The 1/2 LSb error is the maximum error allowed for the ADC to meet its specified resolution.

EQUATION 16-1: ACQUISITION TIME EXAMPLE

Assumptions: Temperature = 50° C and external impedance of $10k\Omega 5.0V VDD$

$$TACQ = Amplifier Settling Time + Hold Capacitor Charging Time + Temperature Coefficient$$

= $TAMP + TC + TCOFF$
= $2\mu s + TC + [(Temperature - 25°C)(0.05\mu s/°C)]$

The value for TC can be approximated with the following equations:

$$V_{APPLIED}\left(1 - \frac{1}{(2^{n+1}) - 1}\right) = V_{CHOLD}$$
 ;[1] V_{CHOLD} charged to within 1/2 lsb

$$V_{APPLIED}\left(1-e^{\frac{-Tc}{RC}}\right) = V_{CHOLD}$$
 ;[2] V_{CHOLD} charge response to $V_{APPLIED}$

$$V_{APPLIED}\left(1-e^{\frac{-Tc}{RC}}\right) = V_{APPLIED}\left(1-\frac{1}{(2^{n+1})-1}\right)$$
 ; combining [1] and [2]

Note: Where n = number of bits of the ADC.

Solving for TC:

$$TC = -C_{HOLD}(RIC + RSS + RS) \ln(1/2047)$$

= $-12.5pF(1k\Omega + 7k\Omega + 10k\Omega) \ln(0.0004885)$
= $1.12\mu s$

Therefore:

$$TACQ = 5\mu s + 1.12\mu s + [(50^{\circ}C - 25^{\circ}C)(0.05\mu s/^{\circ}C)]$$

= 7.37\mu s

- Note 1: The reference voltage (VREF+) has no effect on the equation, since it cancels itself out.
 - 2: The charge holding capacitor (CHOLD) is not discharged after each conversion.
 - **3:** The maximum recommended impedance for analog sources is 10 kΩ. This is required to meet the pin leakage specification.

FIGURE 16-4: ANALOG INPUT MODEL

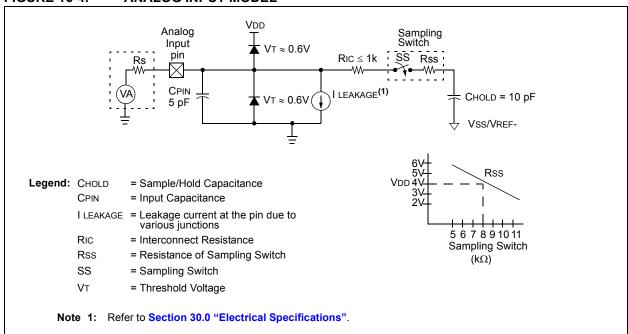


FIGURE 16-5: ADC TRANSFER FUNCTION

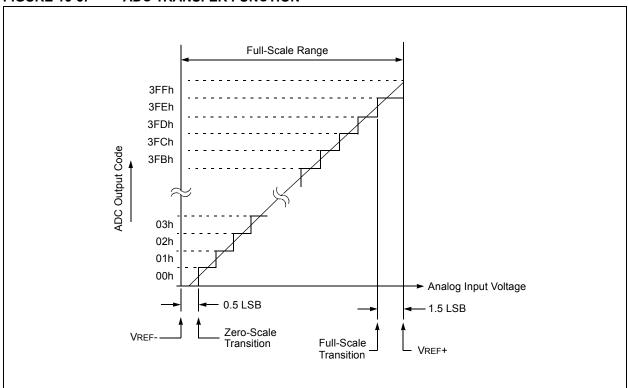


TABLE 16-3: SUMMARY OF REGISTERS ASSOCIATED WITH ADC

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ADCON0	_			CHS<4:0>			GO/DONE	ADON	155
ADCON1	ADFM		ADCS<2:0>		_	ADNREF	ADPRI	EF<1:0	156
ADRESH	A/D Result Re	egister High							157, 158
ADRESL	A/D Result Re	egister Low							157, 158
ANSELA	_	_	_	ANSA4	_	ANSA2	ANSA1	ANSA0	126
ANSELB ⁽¹⁾	ANSB7	ANSB6	ANSB5	ANSB4	_	_	_	_	132
ANSELC	ANSC7 ⁽¹⁾	ANSC6 ⁽¹⁾	_	_	ANSC3	ANSC2	ANSC1	ANSC0	137
CCP4CON	P4M·	<1:0>	1:0> DC4B<1:0>			CCP4I	M<3:0>		236
INLVLA	_	_	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	127
INLVLB ⁽¹⁾	INLVLB7	INLVLB6	INLVLB5	INLVLB4	_	_	_	_	132
INLVLC	INLVLC7 ⁽¹⁾	INLVLC6 ⁽¹⁾	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	138
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	90
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	91
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	95
TRISA	_	_	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	125
TRISB ⁽¹⁾	TRISB7	TRISB6	TRISB5	TRISB4	_	_	_	_	131
TRISC	TRISC7 ⁽¹⁾	TRISC6 ⁽¹⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	136
FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDAFV	′R<1:0>	ADFVF	₹<1:0>	146
DACCON0	DACEN	DACLPS	DACOE	_	DACPS	SS<1:0>	_	DACNSS	166
DACCON1	_	_	_			DACR<4:0>			166

x = unknown, u = unchanged, - = unimplemented read as '0', q = value depends on condition. Shaded cells are not used for ADC module. PIC16(L)F1829 only. Legend:

Note 1:

NOTES:

17.0 DIGITAL-TO-ANALOG CONVERTER (DAC) MODULE

The Digital-to-Analog Converter supplies a variable voltage reference, ratiometric with the input source, with 32 selectable output levels.

The input of the DAC can be connected to:

- · External VREF pins
- VDD supply voltage
- FVR (Fixed Voltage Reference)

The output of the DAC can be configured to supply a reference voltage to the following:

- · Comparator positive input
- · ADC input channel
- · DACOUT pin
- Capacitive Sensing module (CPS)

The Digital-to-Analog Converter (DAC) can be enabled by setting the DACEN bit of the DACCON0 register.

17.1 Output Voltage Selection

The DAC has 32 voltage level ranges. The 32 levels are set with the DACR<4:0> bits of the DACCON1 register.

The DAC output voltage is determined by the following equations:

EQUATION 17-1: DAC OUTPUT VOLTAGE

IF DACEN = 1

$$VOUT = \left((VSOURCE + - VSOURCE -) \times \frac{DACR[4:0]}{2^5} \right) + VSOURCE - VSOUR$$

IF DACEN = 0 and DACLPS = 1 and DACR[4:0] = 11111

VOUT = VSOURCE +

IF DACEN = 0 and DACLPS = 0 and DACR[4:0] = 00000

VOUT = VSOURCE -

VSOURCE+ = VDD, VREF, or FVR BUFFER 2

VSOURCE-=VSS

17.2 Ratiometric Output Level

The DAC output value is derived using a resistor ladder with each end of the ladder tied to a positive and negative voltage reference input source. If the voltage of either input source fluctuates, a similar fluctuation will result in the DAC output value.

The value of the individual resistors within the ladder can be found in Section 30.0 "Electrical Specifications".

17.3 DAC Voltage Reference Output

The DAC can be output to the DACOUT pin by setting the DACOE bit of the DACCON0 register to '1'. Selecting the DAC reference voltage for output on the DACOUT pin automatically overrides the digital output buffer and digital input threshold detector functions of that pin. Reading the DACOUT pin when it has been configured for DAC reference voltage output will always return a '0'.

Due to the limited current drive capability, a buffer must be used on the DAC voltage reference output for external connections to DACOUT. Figure 17.5 shows an example buffering technique.

FIGURE 17-1: DIGITAL-TO-ANALOG CONVERTER BLOCK DIAGRAM

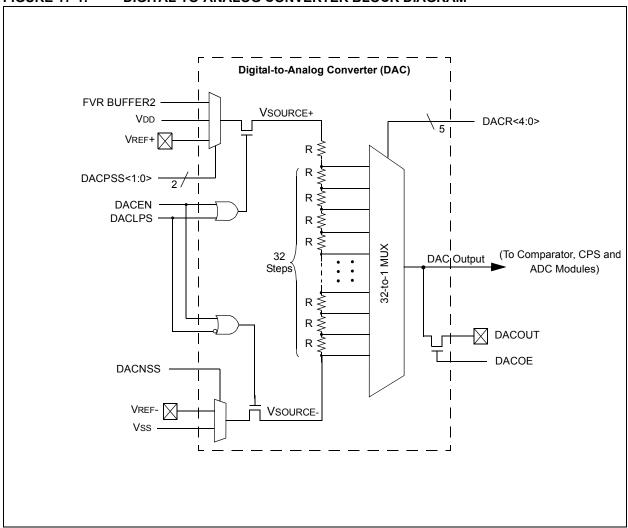
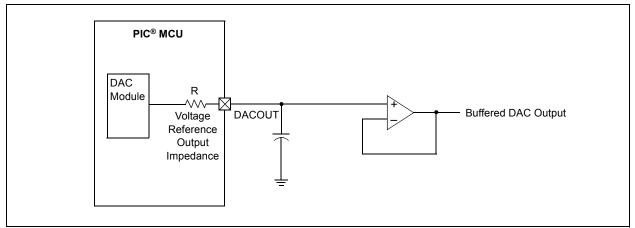


FIGURE 17-2: VOLTAGE REFERENCE OUTPUT BUFFER EXAMPLE



17.4 Low-Power Voltage State

In order for the DAC module to consume the least amount of power, one of the two voltage reference input sources to the resistor ladder must be disconnected. Either the positive voltage source, (VSOURCE+), or the negative voltage source, (VSOURCE-) can be disabled.

The negative voltage source is disabled by setting the DACLPS bit in the DACCON0 register. Clearing the DACLPS bit in the DACCON0 register disables the positive voltage source.

17.4.1 OUTPUT CLAMPED TO POSITIVE VOLTAGE SOURCE

The DAC output voltage can be set to VSOURCE+ with the least amount of power consumption by performing the following:

- · Clearing the DACEN bit in the DACCON0 register.
- Setting the DACLPS bit in the DACCON0 register.
- Configuring the DACPSS bits to the proper positive source.
- Configuring the DACR<4:0> bits to '11111' in the DACCON1 register.

This is also the method used to output the voltage level from the FVR to an output pin. See **Section 17.5** "Operation During Sleep" for more information.

Reference Figure 17-3 for output clamping examples.

17.4.2 OUTPUT CLAMPED TO NEGATIVE VOLTAGE SOURCE

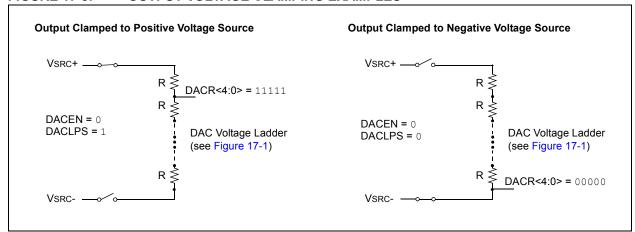
The DAC output voltage can be set to VSOURCE- with the least amount of power consumption by performing the following:

- · Clearing the DACEN bit in the DACCON0 register.
- · Clearing the DACLPS bit in the DACCON0 register.
- Configuring the DACNSS bits to the proper negative source.
- Configuring the DACR<4:0> bits to '00000' in the DACCON1 register.

This allows the comparator to detect a zero-crossing while not consuming additional current through the DAC module.

Reference Figure 17-3 for output clamping examples.

FIGURE 17-3: OUTPUT VOLTAGE CLAMPING EXAMPLES



17.5 Operation During Sleep

When the device wakes up from Sleep through an interrupt or a Watchdog Timer time-out, the contents of the DACCON0 register are not affected. To minimize current consumption in Sleep mode, the voltage reference should be disabled.

17.6 Effects of a Reset

A device Reset affects the following:

- · DAC is disabled.
- DAC output voltage is removed from the DACOUT pin.
- The DACR<4:0> range select bits are cleared.

17.7 DAC Control Registers

REGISTER 17-1: DACCON0: VOLTAGE REFERENCE CONTROL REGISTER 0

R/W-0/0	R/W-0/0	R/W-0/0	U-0	R/W-0/0	R/W-0/0	U-0	R/W-0/0
DACEN	DACLPS	DACOE	_	DACPS	SS<1:0>	_	DACNSS
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

bit 7 DACEN: DAC Enable bit

1 = DAC is enabled

0 = DAC is disabled

bit 6 DACLPS: DAC Low-Power Voltage State Select bit

1 = DAC Positive reference source selected0 = DAC Negative reference source selected

bit 5 DACOE: DAC Voltage Output Enable bit

1 = DAC voltage level is also an output on the DACOUT pin

0 = DAC voltage level is disconnected from the DACOUT pin

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

bit 3-2 DACPSS<1:0>: DAC Positive Source Select bits

00 = VDD

01 = **V**REF+

10 = FVR Buffer2 output 11 = Reserved, do not use

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

bit 0 DACNSS: DAC Negative Source Select bits

1 = **V**REF-

0 = Vss

REGISTER 17-2: DACCON1: VOLTAGE REFERENCE CONTROL REGISTER 1

U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
_	_	_			DACR<4:0>		
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

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

bit 4-0 DACR<4:0>: DAC Voltage Output Select bits

Vout = ((VSRC+) - (VSRC-))*(DACR<4:0>/(2⁵)) + VSRC-

Note 1: The output select bits are always right justified to ensure that any number of bits can be used without affecting the register layout

and coming and register layout

TABLE 17-1: SUMMARY OF REGISTERS ASSOCIATED WITH THE DAC MODULE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDAFVR<1:0>		ADFVR<1:0>		146
DACCON0	DACEN	DACLPS	DACOE	-	DACPS	SS<1:0>	_	DACNSS	166
DACCON1	_	_	_		DACR<4:0>				

Legend: — Unimplemented, read as '0'. Shaded cells are unused by the DAC module.

NOTES:

18.0 SR LATCH

The module consists of a single SR latch with multiple Set and Reset inputs as well as separate latch outputs. The SR latch module includes the following features:

- · Programmable input selection
- SR latch output is available externally
- Separate Q and Q outputs
- · Firmware Set and Reset

The SR latch can be used in a variety of analog applications, including oscillator circuits, one-shot circuit, hysteretic controllers, and analog timing applications.

18.1 Latch Operation

The latch is a Set-Reset Latch that does not depend on a clock source. Each of the Set and Reset inputs are active-high. The latch can be Set or Reset by:

- Software control (SRPS and SRPR bits)
- Comparator C1 output (sync C1OUT)
- Comparator C2 output (sync_C2OUT) (PIC16(L)F1829 only)
- SRI pin
- · Programmable clock (SRCLK)

The SRPS and the SRPR bits of the SRCON0 register may be used to Set or Reset the SR latch, respectively. The latch is Reset-dominant. Therefore, if both Set and Reset inputs are high, the latch will go to the Reset state. Both the SRPS and SRPR bits are self resetting which means that a single write to either of the bits is all that is necessary to complete a latch Set or Reset operation.

The output from Comparator C1 or C2 can be used as the Set or Reset inputs of the SR latch. The output of either Comparator can be synchronized to the Timer1 clock source. See Section 19.0 "Comparator Module" and Section 21.0 "Timer1 Module with Gate Control" for more information.

An external source on the SRI pin can be used as the Set or Reset inputs of the SR latch.

An internal clock source is available that can periodically Set or Reset the SR latch. The SRCLK<2:0> bits in the SRCON0 register are used to select the clock source period. The SRSCKE and SRRCKE bits of the SRCON1 register enable the clock source to Set or Reset the SR latch, respectively.

18.2 Latch Output

The SRQEN and SRNQEN bits of the SRCON0 register control the Q and \overline{Q} latch outputs. Both of the SR latch outputs may be directly output to an I/O pin at the same time.

The applicable TRIS bit of the corresponding port must be cleared to enable the port pin output driver.

18.3 Effects of a Reset

Upon any device Reset, the SR latch output is not initialized to a known state. The user's firmware is responsible for initializing the latch output before enabling the output pins.

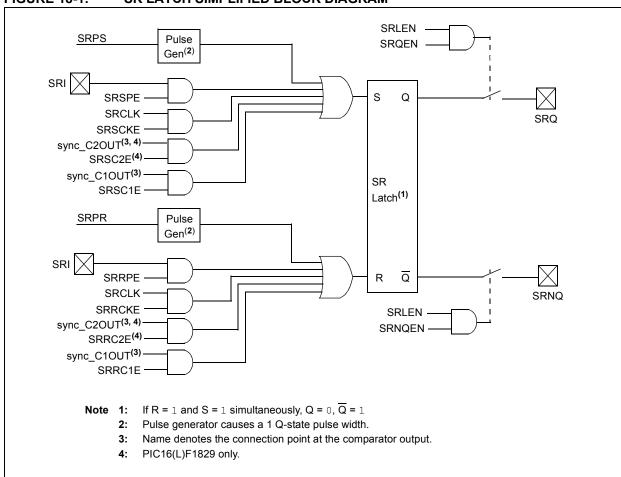


TABLE 18-1: SRCLK FREQUENCY TABLE

SRCLK	Divider	Fosc = 32 MHz	Fosc = 20 MHz	Fosc = 16 MHz	Fosc = 4 MHz	Fosc = 1 MHz
111	512	62.5 kHz	39.0 kHz	31.3 kHz	7.81 kHz	1.95 kHz
110	256	125 kHz	78.1 kHz	62.5 kHz	15.6 kHz	3.90 kHz
101	128	250 kHz	156 kHz	125 kHz	31.25 kHz	7.81 kHz
100	64	500 kHz	313 kHz	250 kHz	62.5 kHz	15.6 kHz
011	32	1 MHz	625 kHz	500 kHz	125 kHz	31.3 kHz
010	16	2 MHz	1.25 MHz	1 MHz	250 kHz	62.5 kHz
001	8	4 MHz	2.5 MHz	2 MHz	500 kHz	125 kHz
000	4	8 MHz	5 MHz	4 MHz	1 MHz	250 kHz

REGISTER 18-1: SRCON0: SR LATCH CONTROL 0 REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/S-0/0	R/S-0/0
SRLEN		SRCLK<2:0>		SRQEN	SRNQEN	SRPS	SRPR
bit 7							bit 0

Legend:

bit 2

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared S = Bit is set only

bit 7 SRLEN: SR Latch Enable bit

1 = SR latch is enabled

0 = SR latch is disabled

bit 6-4 SRCLK<2:0>: SR Latch Clock Divider bits

000 = Generates a 1 Fosc wide pulse every 4th Fosc cycle clock

001 = Generates a 1 Fosc wide pulse every 8th Fosc cycle clock

010 = Generates a 1 Fosc wide pulse every 16th Fosc cycle clock

011 = Generates a 1 Fosc wide pulse every 32nd Fosc cycle clock

100 = Generates a 1 Fosc wide pulse every 64th Fosc cycle clock

101 = Generates a 1 Fosc wide pulse every 128th Fosc cycle clock

110 = Generates a 1 Fosc wide pulse every 256th Fosc cycle clock

111 = Generates a 1 Fosc wide pulse every 512th Fosc cycle clock

bit 3 SRQEN: SR Latch Q Output Enable bit

If SRLEN = 1:

1 = Q is present on the SRQ pin

0 = External Q output is disabled

If SRLEN = 0:

SR latch is disabled

SRNQEN: SR Latch Q Output Enable bit

If SRLEN = 1:

 $1 = \overline{Q}$ is present on the SRnQ pin

 $0 = \text{External } \overline{Q} \text{ output is disabled}$

If SRLEN = 0:

SR latch is disabled

bit 1 SRPS: Pulse Set Input of the SR Latch bit⁽¹⁾

1 = Pulse set input for 1 Q-clock period

0 = No effect on set input.

bit 0 SRPR: Pulse Reset Input of the SR Latch bit (1)

1 = Pulse Reset input for 1 Q-clock period

0 = No effect on Reset input.

Note 1: Set only, always reads back '0'.

REGISTER 18-2: SRCON1: SR LATCH CONTROL 1 REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
SRSPE	SRSCKE	SRSC2E ⁽¹⁾	SRSC1E	SRRPE	SRRCKE	SRRC2E ⁽¹⁾	SRRC1E
bit 7							bit 0

Legend:R = Readable bitW = Writable bitU = Unimplemented bit, read as '0'u = Bit is unchangedx = Bit is unknown-n/n = Value at POR and BOR/Value at all other Resets'1' = Bit is set'0' = Bit is cleared

bit 7	CDCDE: CD Latch Deripheral Set Enable hit
DIL /	SRSPE: SR Latch Peripheral Set Enable bit
	1 = SR latch is set when the SRI pin is high0 = SRI pin has no effect on the set input of the SR latch
1.11.0	·
bit 6	SRSCKE: SR Latch Set Clock Enable bit
	1 = Set input of SR latch is pulsed with SRCLK
	0 = SRCLK has no effect on the set input of the SR latch
bit 5	SRSC2E: SR Latch C2 Set Enable bit ⁽¹⁾
	1 = SR latch is set when the C2 Comparator output is high
	0 = C2 Comparator output has no effect on the set input of the SR latch
bit 4	SRSC1E: SR Latch C1 Set Enable bit
	1 = SR latch is set when the C1 Comparator output is high
	0 = C1 Comparator output has no effect on the set input of the SR latch
bit 3	SRRPE: SR Latch Peripheral Reset Enable bit
	1 = SR latch is reset when the SRI pin is high
	0 = SRI pin has no effect on the Reset input of the SR latch
bit 2	SRRCKE: SR Latch Reset Clock Enable bit
	1 = Reset input of SR latch is pulsed with SRCLK
	0 = SRCLK has no effect on the Reset input of the SR latch
bit 1	SRRC2E: SR Latch C2 Reset Enable bit ⁽¹⁾
	1 = SR latch is reset when the C2 Comparator output is high
	0 = C2 Comparator output has no effect on the Reset input of the SR latch
bit 0	SRRC1E: SR Latch C1 Reset Enable bit
	1 = SR latch is reset when the C1 Comparator output is high
	0 = C1 Comparator output has no effect on the Reset input of the SR latch

Note 1: PIC16(L)F1829 only.

TABLE 18-2: SUMMARY OF REGISTERS ASSOCIATED WITH SR LATCH MODULE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ANSELA	_	1	_	ANSA4	1	ANSA2	ANSA1	ANSA0	126
INLVLA	_	1	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	127
INLVLC	INLVLC7 ⁽¹⁾	INLVLC6 ⁽¹⁾	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	138
SRCON0	SRLEN	Ş	SRCLK<2:0>		SRQEN	SRNQEN	SRPS	SRPR	171
SRCON1	SRSPE	SRSCKE	SRSC2E ⁽¹⁾	SRSC1E	SRRPE	SRRCKE	SRRC2E ⁽¹⁾	SRRC1E	172
TRISA	_	_	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	125
TRISC	TRISC7 ⁽¹⁾	TRISC6 ⁽¹⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	136

Legend: — Unimplemented, read as '0'. Shaded cells are unused by the SR latch module.

Note 1: PIC16(L)F1829 only.

NOTES:

19.0 COMPARATOR MODULE

Comparators are used to interface analog circuits to a digital circuit by comparing two analog voltages and providing a digital indication of their relative magnitudes. Comparators are very useful mixed signal building blocks because they provide analog functionality independent of program execution. The analog Comparator module includes the following features:

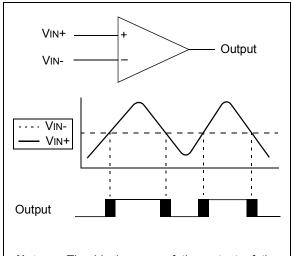
- · Independent comparator control
- · Programmable input selection
- Comparator output is available internally/externally
- · Programmable output polarity
- · Interrupt-on-change
- · Wake-up from Sleep
- Programmable Speed/Power optimization
- · PWM shutdown
- · Programmable and Fixed Voltage Reference

19.1 Comparator Overview

A single comparator is shown in Figure 19-1 along with the relationship between the analog input levels and the digital output. When the analog voltage at VIN+ is less than the analog voltage at VIN-, the output of the

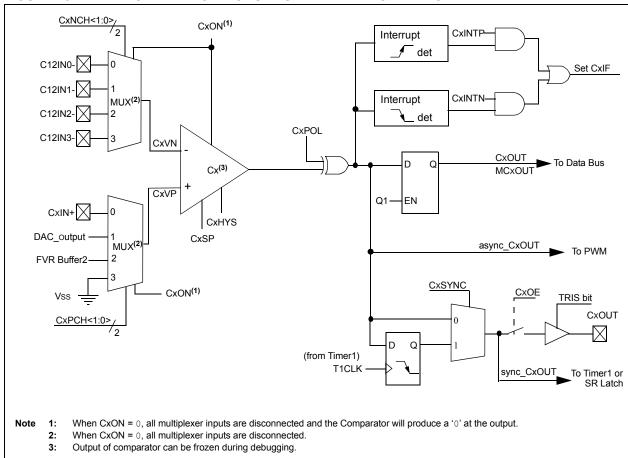
comparator is a digital low level. When the analog voltage at VIN+ is greater than the analog voltage at VIN-, the output of the comparator is a digital high level.

FIGURE 19-1: SINGLE COMPARATOR



Note: The black areas of the output of the comparator represents the uncertainty due to input offsets and response time.

FIGURE 19-2: COMPARATOR MODULE SIMPLIFIED BLOCK DIAGRAM



19.2 Comparator Control

Each comparator has two control registers: CMxCON0 and CMxCON1.

The CMxCON0 registers (see Register 19-1) contain Control and Status bits for the following:

- Fnable
- · Output selection
- · Output polarity
- · Speed/Power selection
- · Hysteresis enable
- · Output synchronization

The CMxCON1 registers (see Register 19-2) contain Control bits for the following:

- · Interrupt enable
- · Interrupt edge polarity
- · Positive input channel selection
- · Negative input channel selection

19.2.1 COMPARATOR ENABLE

Setting the CxON bit of the CMxCON0 register enables the comparator for operation. Clearing the CxON bit disables the comparator resulting in minimum current consumption.

19.2.2 COMPARATOR OUTPUT SELECTION

The output of the comparator can be monitored by reading either the CxOUT bit of the CMxCON0 register or the MCxOUT bit of the CMOUT register. In order to make the output available for an external connection, the following conditions must be true:

- CxOE bit of the CMxCON0 register must be set
- · Corresponding TRIS bit must be cleared
- · CxON bit of the CMxCON0 register must be set
 - Note 1: The CxOE bit of the CMxCON0 register overrides the PORT data latch. Setting the CxON bit of the CMxCON0 register has no impact on the port override.
 - 2: The internal output of the comparator is latched with each instruction cycle. Unless otherwise specified, external outputs are not latched.

19.2.3 COMPARATOR OUTPUT POLARITY

Inverting the output of the comparator is functionally equivalent to swapping the comparator inputs. The polarity of the comparator output can be inverted by setting the CxPOL bit of the CMxCON0 register. Clearing the CxPOL bit results in a non-inverted output.

Table 19-1 shows the output state versus input conditions, including polarity control.

TABLE 19-1: COMPARATOR OUTPUT STATE VS. INPUT CONDITIONS

Input Condition	CxPOL	CxOUT
CxVn > CxVp	0	0
CxVn < CxVp	0	1
CxVn > CxVp	1	1
CxVn < CxVp	1	0

19.2.4 COMPARATOR SPEED/POWER SELECTION

The trade-off between speed or power can be optimized during program execution with the CxSP control bit. The default state for this bit is '1' which selects the Normal Speed mode. Device power consumption can be optimized at the cost of slower comparator propagation delay by clearing the CxSP bit to '0'.

19.3 Comparator Hysteresis

A selectable amount of separation voltage can be added to the input pins of each comparator to provide a hysteresis function to the overall operation. Hysteresis is enabled by setting the CxHYS bit of the CMxCON0 register.

See Section 30.0 "Electrical Specifications" for more information.

19.4 Timer1 Gate Operation

The output resulting from a comparator operation can be used as a source for gate control of Timer1. See **Section 21.6 "Timer1 Gate"** for more information. This feature is useful for timing the duration or interval of an analog event.

It is recommended that the comparator output be synchronized to Timer1. This ensures that Timer1 does not increment while a change in the comparator is occurring.

19.4.1 COMPARATOR OUTPUT SYNCHRONIZATION

The output from either comparator, C1 or C2, can be synchronized with Timer1 by setting the CxSYNC bit of the CMxCON0 register.

Once enabled, the comparator output is latched on the falling edge of the Timer1 source clock. If a prescaler is used with Timer1, the comparator output is latched after the prescaling function. To prevent a race condition, the comparator output is latched on the falling edge of the Timer1 clock source and Timer1 increments on the rising edge of its clock source. See the Comparator Block Diagram (Figure 19-2) and the Timer1 Block Diagram (Figure 21-1) for more information.

19.5 Comparator Interrupt

An interrupt can be generated upon a change in the output value of the comparator for each comparator, a rising edge detector and a Falling edge detector are present.

When either edge detector is triggered and its associated enable bit is set (CxINTP and/or CxINTN bits of the CMxCON1 register), the Corresponding Interrupt Flag bit (CxIF bit of the PIR2 register) will be set.

To enable the interrupt, you must set the following bits:

- CxON, CxPOL and CxSP bits of the CMxCON0 register
- · CxIE bit of the PIE2 register
- CxINTP bit of the CMxCON1 register (for a rising edge detection)
- CxINTN bit of the CMxCON1 register (for a falling edge detection)
- · PEIE and GIE bits of the INTCON register

The associated interrupt flag bit, CxIF bit of the PIR2 register, must be cleared in software. If another edge is detected while this flag is being cleared, the flag will still be set at the end of the sequence.

Note:

Although a comparator is disabled, an interrupt can be generated by changing the output polarity with the CxPOL bit of the CMxCON0 register, or by switching the comparator on or off with the CxON bit of the CMxCON0 register.

19.6 Comparator Positive Input Selection

Configuring the CxPCH<1:0> bits of the CMxCON1 register directs an internal voltage reference or an analog pin to the non-inverting input of the comparator:

- CxIN+ analog pin
- · DAC_output
- FVR (Fixed Voltage Reference)
- · Vss (Ground)

See Section 14.0 "Fixed Voltage Reference (FVR)" for more information on the Fixed Voltage Reference module.

See Section 17.0 "Digital-to-Analog Converter (DAC) Module" for more information on the DAC input signal.

Any time the comparator is disabled (CxON = 0), all comparator inputs are disabled.

19.7 Comparator Negative Input Selection

The CxNCH<1:0> bits of the CMxCON0 register direct one of four analog pins to the comparator inverting input.

Note:

To use CxIN+ and CxINx- pins as analog input, the appropriate bits must be set in the ANSEL register and the corresponding TRIS bits must also be set to disable the output drivers.

19.8 Comparator Response Time

The comparator output is indeterminate for a period of time after the change of an input source or the selection of a new reference voltage. This period is referred to as the response time. The response time of the comparator differs from the settling time of the voltage reference. Therefore, both of these times must be considered when determining the total response time to a comparator input change. See the Comparator and Voltage Reference Specifications in Section 30.0 "Electrical Specifications" for more details.

19.9 Interaction with ECCP Logic

In some devices, a comparator output signal can be used to trigger the auto-shutdown feature found within the ECCP module. When the ECCP auto-shutdown feature is enabled and a comparator output signal is selected as the source, the comparator can be used simultaneously as a general purpose comparator and as the ECCP auto-shutdown source. In addition, the comparator output signal can also be routed to the designated I/O pin. If the ECCP Auto-Restart mode is also enabled, the comparators can be used as a closed loop analog feedback circuit to the ECCP, thereby creating an analog controlled PWM.

Please see section **Section 24.4.3 "Enhanced PWM Auto-shutdown mode**" for more information.

Note: When the comparator module is first initialized the output state is unknown. Upon initialization, the user should verify the output state of the comparator prior to

relying on the result, primarily when using the result in connection with other peripheral features, such as the ECCP Auto-Shutdown mode.

19.10 Analog Input Connection Considerations

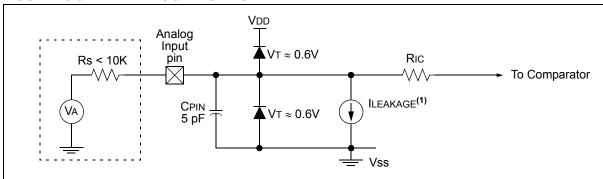
A simplified circuit for an analog input is shown in Figure 19-3. Since the analog input pins share their connection with a digital input, they have reverse biased ESD protection diodes to VDD and Vss. The analog input, therefore, must be between Vss and VDD.

If the input voltage deviates from this range by more than 0.6V in either direction, one of the diodes is forward biased and a latch-up may occur.

A maximum source impedance of 10 k Ω is recommended for the analog sources. Also, any external component connected to an analog input pin, such as a capacitor or a Zener diode, should have very little leakage current to minimize inaccuracies introduced.

- Note 1: When reading a PORT register, all pins configured as analog inputs will read as a '0'. Pins configured as digital inputs will convert as an analog input, according to the input specification.
 - **2:** Analog levels on any pin defined as a digital input, may cause the input buffer to consume more current than is specified.

FIGURE 19-3: ANALOG INPUT MODEL



Legend: CPIN = Input Capacitance

ILEAKAGE = Leakage Current at the pin due to various junctions

RIC = Interconnect Resistance
RS = Source Impedance
VA = Analog Voltage
VT = Threshold Voltage

Note 1: See Section 30.0 "Electrical Specifications".

REGISTER 19-1: CMxCON0: COMPARATOR Cx CONTROL REGISTER 0

R/W-0/0	R-0/0	R/W-0/0	R/W-0/0	U-0	R/W-1/1	R/W-0/0	R/W-0/0	
CxON	CxOUT	CxOE	CxPOL	_	CxSP	CxHYS	CxSYNC	
bit 7 bit 0								

Legend:R = Readable bitW = Writable bitU = Unimplemented bit, read as '0'u = Bit is unchangedx = Bit is unknown-n/n = Value at POR and BOR/Value at all other Resets'1' = Bit is set'0' = Bit is cleared

bit 7 **CxON:** Comparator Enable bit

1 = Comparator is enabled and consumes no active power

0 = Comparator is disabled

bit 6 **CxOUT:** Comparator Output bit

If CxPOL = 1 (inverted polarity):

1 = CxVP < CxVN 0 = CxVP > CxVN

If CxPOL = 0 (non-inverted polarity):

1 = CxVP > CxVN

0 = CxVP < CxVN

bit 5 **CxOE:** Comparator Output Enable bit

1 = CxOUT is present on the CxOUT pin. Requires that the associated TRIS bit be cleared to actually drive the pin. Not affected by CxON.

0 = CxOUT is internal only

1 = Comparator output is inverted0 = Comparator output is not inverted

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

bit 2 CxSP: Comparator Speed/Power Select bit

1 = Comparator operates in normal power, higher speed mode

0 = Comparator operates in low-power, low-speed mode

bit 1 CxHYS: Comparator Hysteresis Enable bit

1 = Comparator hysteresis enabled

0 = Comparator hysteresis disabled

bit 0 **CxSYNC:** Comparator Output Synchronous Mode bit

1 = Comparator output to Timer1 and I/O pin is synchronous to changes on Timer1 clock source. Output updated on the falling edge of Timer1 clock source.

0 = Comparator output to Timer1 and I/O pin is asynchronous.

REGISTER 19-2: CMxCON1: COMPARATOR Cx CONTROL REGISTER 1

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	U-0	R/W-0/0	R/W-0/0
CxINTP	CxINTN	CxPCH<1:0>		_	_	CxNCH<1:0>	
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

bit 7 CxINTP: Comparator Interrupt on Positive Going Edge Enable bits

1 = The CxIF interrupt flag will be set upon a positive going edge of the CxOUT bit

0 = No interrupt flag will be set on a positive going edge of the CxOUT bit

bit 6 CXINTN: Comparator Interrupt on Negative Going Edge Enable bits

1 = The CxIF interrupt flag will be set upon a negative going edge of the CxOUT bit

0 = No interrupt flag will be set on a negative going edge of the CxOUT bit

bit 5-4 **CxPCH<1:0>:** Comparator Positive Input Channel Select bits

00 = CxVP connects to CxIN+ pin

01 = CxVP connects to DAC voltage reference10 = CxVP connects to FVR voltage reference

For C1:

11 = CxVP connects to C12IN+ pin

For C2:

11 = CxVP connects to Vss

bit 3-2 **Unimplemented:** Read as '0'

bit 1-0 CxNCH<1:0>: Comparator Negative Input Channel Select bits

00 = CxVN connects to C12IN0- pin 01 = CxVN connects to C12IN1- pin

10 = CxVN connects to C12IN2- pin

11 = CxVN connects to C12IN3- pin

REGISTER 19-3: CMOUT: COMPARATOR OUTPUT REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	R-0/0	R-0/0
_	_	_	_	_	_	MC2OUT	MC1OUT
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

bit 7-2 **Unimplemented:** Read as '0'

bit 1 MC2OUT: Mirror Copy of C2OUT bit bit 0 MC1OUT: Mirror Copy of C1OUT bit

TABLE 19-2: SUMMARY OF REGISTERS ASSOCIATED WITH COMPARATOR MODULE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
CM1CON0	C10N	C10UT	C10E	C1POL	_	C1SP	C1HYS	C1SYNC	179
CM1CON1	C1NTP	C1INTN	C1PCH1	C1PCH0	_	_	C1NCI	H<1:0>	180
CM2CON0	C2ON	C2OUT	C2OE	C2POL	_	C2SP	C2HYS	C2SYNC	179
CM2CON1	C2NTP	C2INTN	C2PCI	H<1:0>	_	_	C2NCI	H<1:0>	180
CMOUT	_	-	-	-	_	_	MC2OUT	MC10UT	180
DACCON0	DACEN	DACLPS	DACOE	_	DACPS	SS<1:0>	_	DACNSS	166
DACCON1	_	-				DACR<4:0>			166
FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDAFV	'R<1:0>	ADFVF	R<1:0>	146
INLVLA	_	1	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	127
INLVLC	INLVLC7 ⁽¹⁾	INLVLC6 ⁽¹⁾	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	138
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	90
PIE2	OSFIE	C2IE	C1IE	EEIE	BCL1IE	_	_	CCP2IE	92
PIR2	OSFIF	C2IF	C1IF	EEIF	BCL1IF	_	_	CCP2IF	96
PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	136
LATC	LATC7	LATC6	LATC5	LATC4	LATC3	LATC2	LATC1	LATC0	136
TRISA	_	_	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	125
TRISC	TRISC7 ⁽¹⁾	TRISC6 ⁽¹⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	136

Legend: — Unimplemented location, read as '0'. Shaded cells are unused by the comparator module.

Note 1: PIC16(L)F1829 only.

NOTES:

20.0 TIMERO MODULE

The Timer0 module is an 8-bit timer/counter with the following features:

- 8-bit timer/counter register (TMR0)
- 8-bit prescaler (independent of Watchdog Timer)
- · Programmable internal or external clock source
- · Programmable external clock edge selection
- · Interrupt on overflow
- TMR0 can be used to gate Timer1

Figure 20-1 is a block diagram of the Timer0 module.

20.1 Timer0 Operation

The Timer0 module can be used as either an 8-bit timer or an 8-bit counter.

20.1.1 8-BIT TIMER MODE

The Timer0 module will increment every instruction cycle, if used without a prescaler. 8-bit Timer mode is selected by clearing the TMR0CS bit of the OPTION REG register.

When TMR0 is written, the increment is inhibited for two instruction cycles immediately following the write.

Note: The value written to the TMR0 register can be adjusted, in order to account for the two instruction cycle delay when TMR0 is written.

20.1.2 8-BIT COUNTER MODE

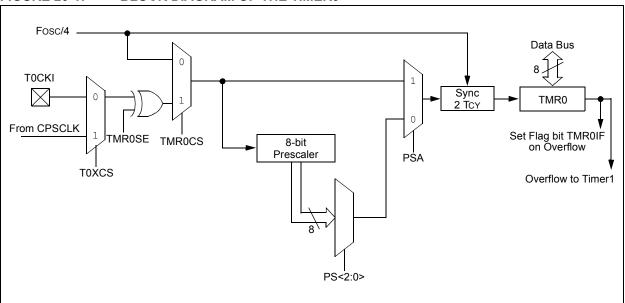
In 8-Bit Counter mode, the Timer0 module will increment on every rising or falling edge of the T0CKI pin or the Capacitive Sensing Oscillator (CPSCLK) signal.

8-Bit Counter mode using the T0CKI pin is selected by setting the TMR0CS bit in the OPTION_REG register to '1' and resetting the T0XCS bit in the CPSCON0 register to '0'.

8-Bit Counter mode using the Capacitive Sensing Oscillator (CPSCLK) signal is selected by setting the TMR0CS bit in the OPTION_REG register to '1' and setting the T0XCS bit in the CPSCON0 register to '1'.

The rising or falling transition of the incrementing edge for either input source is determined by the TMR0SE bit in the OPTION REG register.

FIGURE 20-1: BLOCK DIAGRAM OF THE TIMER0



20.1.3 SOFTWARE PROGRAMMABLE PRESCALER

A software programmable prescaler is available for exclusive use with Timer0. The prescaler is enabled by clearing the PSA bit of the OPTION_REG register.

Note: The Watchdog Timer (WDT) uses its own independent prescaler.

There are eight prescaler options for the Timer0 module ranging from 1:2 to 1:256. The prescale values are selectable via the PS<2:0> bits of the OPTION_REG register. In order to have a 1:1 prescaler value for the Timer0 module, the prescaler must be disabled by setting the PSA bit of the OPTION_REG register.

The prescaler is not readable or writable. All instructions writing to the TMR0 register will clear the prescaler.

20.1.4 TIMERO INTERRUPT

Timer0 will generate an interrupt when the TMR0 register overflows from FFh to 00h. The TMR0IF interrupt flag bit of the INTCON register is set every time the TMR0 register overflows, regardless of whether or not the Timer0 interrupt is enabled. The TMR0IF bit can only be cleared in software. The Timer0 interrupt enable is the TMR0IE bit of the INTCON register.

Note: The Timer0 interrupt cannot wake the processor from Sleep since the timer is frozen during Sleep.

20.1.5 8-BIT COUNTER MODE SYNCHRONIZATION

When in 8-Bit Counter mode, the incrementing edge on the T0CKI pin must be synchronized to the instruction clock. Synchronization can be accomplished by sampling the prescaler output on the Q2 and Q4 cycles of the instruction clock. The high and low periods of the external clocking source must meet the timing requirements as shown in Section 30.0 "Electrical Specifications".

20.1.6 OPERATION DURING SLEEP

Timer0 cannot operate while the processor is in Sleep mode. The contents of the TMR0 register will remain unchanged while the processor is in Sleep mode.

20.2 Option and Timer0 Control Register

REGISTER 20-1: OPTION_REG: OPTION REGISTER

| R/W-1/1 |
|---------|---------|---------|---------|---------|---------|---------|---------|
| WPUEN | INTEDG | TMR0CS | TMR0SE | PSA | | PS<2:0> | |
| bit 7 | | | | | | | bit 0 |

Legend:R = Readable bitW = Writable bitU = Unimplemented bit, read as '0'u = Bit is unchangedx = Bit is unknown-n/n = Value at POR and BOR/Value at all other Resets'1' = Bit is set'0' = Bit is cleared

bit 7 WPUEN: Weak Pull-up Enable bit 1 = All weak pull-ups are disabled (except MCLR, if it is enabled) 0 = Weak pull-ups are enabled by individual WPUx latch values bit 6 INTEDG: Interrupt Edge Select bit 1 = Interrupt on rising edge of INT pin 0 = Interrupt on falling edge of INT pin bit 5 TMR0CS: Timer0 Clock Source Select bit 1 = Transition on T0CKI pin 0 = Internal instruction cycle clock (Fosc/4) bit 4 TMR0SE: Timer0 Source Edge Select bit 1 = Increment on high-to-low transition on T0CKI pin 0 = Increment on low-to-high transition on T0CKI pin bit 3 PSA: Prescaler Assignment bit 1 = Prescaler is not assigned to the Timer0 module 0 = Prescaler is assigned to the Timer0 module

> Bit Value Timer0 Rate 1:2 000 001 1:4 010 1:8 1:16 011 100 1:32 101 1:64 1:128 110 1:256 111

PS<2:0>: Prescaler Rate Select bits

bit 2-0

TABLE 20-1: SUMMARY OF REGISTERS ASSOCIATED WITH TIMERO

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
CPSCON0	CPSON	CPSRM	_	_	CPSRN	IG<1:0>	CPSOUT	T0xCS	327
FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDAFV	′R<1:0>	ADFVI	R<1:0>	146
INLVLA	_	_	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	127
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	90
OPTION_REG	WPUEN	INTEDG	TMR0CS	TMR0SE	PSA		PS<2:0>		185
TMR0	Timer0 Mod	dule Register							183*
TRISA	_	_	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	125

Legend: — Unimplemented location, read as '0'. Shaded cells are not used by the Timer0 module.

^{*} Page provides register information.

NOTES:

21.0 TIMER1 MODULE WITH GATE CONTROL

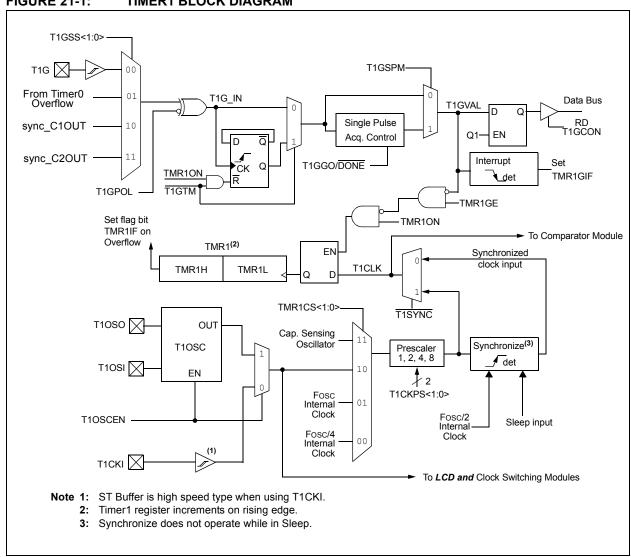
The Timer1 module is a 16-bit timer/counter with the following features:

- 16-bit timer/counter register pair (TMR1H:TMR1L)
- · Programmable internal or external clock source
- · 2-bit prescaler
- · Dedicated 32 kHz oscillator circuit
- · Optionally synchronized comparator out
- Multiple Timer1 gate (count enable) sources
- · Interrupt on overflow
- Wake-up on overflow (external clock, Asynchronous mode only)
- Time base for the Capture/Compare function
- Special Event Trigger (with CCP/ECCP)
- · Selectable Gate Source Polarity

- · Gate Toggle Mode
- · Gate Single-pulse Mode
- · Gate Value Status
- · Gate Event Interrupt

Figure 21-1 is a block diagram of the Timer1 module.

FIGURE 21-1: TIMER1 BLOCK DIAGRAM



21.1 Timer1 Operation

The Timer1 module is a 16-bit incrementing counter which is accessed through the TMR1H:TMR1L register pair. Writes to TMR1H or TMR1L directly update the counter.

When used with an internal clock source, the module is a timer and increments on every instruction cycle. When used with an external clock source, the module can be used as either a timer or counter and increments on every selected edge of the external source.

Timer1 is enabled by configuring the TMR1ON and TMR1GE bits in the T1CON and T1GCON registers, respectively. Table 21-1 displays the Timer1 enable selections.

TABLE 21-1: TIMER1 ENABLE SELECTIONS

TMR10N	TMR1GE	Timer1 Operation
0	0	Off
0	1	Off
1	0	Always On
1	1	Count Enabled

21.2 Clock Source Selection

The TMR1CS<1:0> and T1OSCEN bits of the T1CON register are used to select the clock source for Timer1. Table 21-2 displays the clock source selections.

21.2.1 INTERNAL CLOCK SOURCE

When the internal clock source is selected the TMR1H:TMR1L register pair will increment on multiples of Fosc as determined by the Timer1 prescaler.

When the Fosc internal clock source is selected, the Timer1 register value will increment by four counts every instruction clock cycle. Due to this condition, a 2 LSB error in resolution will occur when reading the Timer1 value. To utilize the full resolution of Timer1, an asynchronous input signal must be used to gate the Timer1 clock input.

The following asynchronous sources may be used:

- Asynchronous event on the T1G pin to Timer1 gate
- · C1 or C2 comparator input to Timer1 gate

21.2.2 EXTERNAL CLOCK SOURCE

When the external clock source is selected, the Timer1 module may work as a timer or a counter.

When enabled to count, Timer1 is incremented on the rising edge of the external clock input T1CKI or the capacitive sensing oscillator signal. Either of these external clock sources can be synchronized to the microcontroller system clock or they can run asynchronously.

When used as a timer with a clock oscillator, an external 32.768 kHz crystal can be used in conjunction with the dedicated internal oscillator circuit.

Note:

In Counter mode, a falling edge must be registered by the counter prior to the first incrementing rising edge after any one or more of the following conditions:

- · Timer1 enabled after POR
- Write to TMR1H or TMR1L
- · Timer1 is disabled
- Timer1 is disabled (TMR1ON = 0) when T1CKI is high then Timer1 is enabled (TMR1ON=1) when T1CKI is low

TABLE 21-2: CLOCK SOURCE SELECTIONS

TMR1CS1	TMR1CS0	T10SCEN	Clock Source
0	0	х	Instruction Clock (Fosc/4)
0	1	X	System Clock (Fosc)
1	0	0	External Clocking on T1CKI Pin
1	0	0	External Clocking on T1CKI Pin
1	1	Х	Capacitive Sensing Oscillator

21.3 Timer1 Prescaler

Timer1 has four prescaler options allowing 1, 2, 4 or 8 divisions of the clock input. The T1CKPS bits of the T1CON register control the prescale counter. The prescale counter is not directly readable or writable; however, the prescaler counter is cleared upon a write to TMR1H or TMR1L.

21.4 Timer1 Oscillator

A dedicated low-power 32.768 kHz oscillator circuit is built-in between pins T1OSI (input) and T1OSO (amplifier output). This internal circuit is to be used in conjunction with an external 32.768 kHz crystal.

The oscillator circuit is enabled by setting the T1OSCEN bit of the T1CON register. The oscillator will continue to run during Sleep.

Note:

The oscillator requires a start-up and stabilization time before use. Thus, T10SCEN should be set and a suitable delay observed prior to using Timer1. A suitable delay similar to the OST delay can be implemented in software by clearing the TMR1IF bit then presetting the TMR1H:TMR1L register pair to FC00h. The TMR1IF flag will be set when 1024 clock cycles have elapsed, thereby indicating that the oscillator is running and reasonably stable.

21.5 Timer1 Operation in Asynchronous Counter Mode

If control bit T1SYNC of the T1CON register is set, the external clock input is not synchronized. The timer increments asynchronously to the internal phase clocks. If the external clock source is selected then the timer will continue to run during Sleep and can generate an interrupt on overflow, which will wake-up the processor. However, special precautions in software are needed to read/write the timer (see Section 21.5.1 "Reading and Writing Timer1 in Asynchronous Counter Mode").

Note:

When switching from synchronous to asynchronous operation, it is possible to skip an increment. When switching from asynchronous to synchronous operation, it is possible to produce an additional increment.

21.5.1 READING AND WRITING TIMER1 IN ASYNCHRONOUS COUNTER MODE

Reading TMR1H or TMR1L while the timer is running from an external asynchronous clock will ensure a valid read (taken care of in hardware). However, the user should keep in mind that reading the 16-bit timer in two 8-bit values itself, poses certain problems, since the timer may overflow between the reads.

For writes, it is recommended that the user simply stop the timer and write the desired values. A write contention may occur by writing to the timer registers, while the register is incrementing. This may produce an unpredictable value in the TMR1H:TMR1L register pair.

21.6 Timer1 Gate

Timer1 can be configured to count freely or the count can be enabled and disabled using Timer1 gate circuitry. This is also referred to as Timer1 Gate Enable.

Timer1 gate can also be driven by multiple selectable sources.

21.6.1 TIMER1 GATE ENABLE

The Timer1 Gate Enable mode is enabled by setting the TMR1GE bit of the T1GCON register. The polarity of the Timer1 Gate Enable mode is configured using the T1GPOL bit of the T1GCON register.

When Timer1 Gate Enable mode is enabled, Timer1 will increment on the rising edge of the Timer1 clock source. When Timer1 Gate Enable mode is disabled, no incrementing will occur and Timer1 will hold the current count. See Figure 21-3 for timing details.

TABLE 21-3: TIMER1 GATE ENABLE SELECTIONS

T1CLK	T1GPOL	T1G	Timer1 Operation
↑	0	0	Counts
1	0	1	Holds Count
1	1	0	Holds Count
↑	1	1	Counts

21.6.2 TIMER1 GATE SOURCE SELECTION

The Timer1 gate source can be selected from one of four different sources. Source selection is controlled by the T1GSS bits of the T1GCON register. The polarity for each available source is also selectable. Polarity selection is controlled by the T1GPOL bit of the T1GCON register.

TABLE 21-4: TIMER1 GATE SOURCES

T1GSS	Timer1 Gate Source
00	Timer1 Gate Pin
01	Overflow of Timer0 (TMR0 increments from FFh to 00h)
10	Comparator 1 Output sync_C1OUT (optionally Timer1 synchronized output)
11	Comparator 2 Output sync_C2OUT (optionally Timer1 synchronized output)

21.6.2.1 T1G Pin Gate Operation

The T1G pin is one source for Timer1 gate control. It can be used to supply an external source to the Timer1 gate circuitry.

21.6.2.2 Timer0 Overflow Gate Operation

When Timer0 increments from FFh to 00h, a low-to-high pulse will automatically be generated and internally supplied to the Timer1 gate circuitry.

21.6.2.3 Comparator C1 Gate Operation

The output resulting from a Comparator 1 operation can be selected as a source for Timer1 gate control. The Comparator 1 output (sync_C1OUT) can be synchronized to the Timer1 clock or left asynchronous. For more information see Section 19.4.1 "Comparator Output Synchronization".

21.6.2.4 Comparator C2 Gate Operation

The output resulting from a Comparator 2 operation can be selected as a source for Timer1 Gate Control. The Comparator 2 output (sync_C2OUT) can be synchronized to the Timer1 clock or left asynchronous. For more information see Section 19.4.1 "Comparator Output Synchronization".

21.6.3 TIMER1 GATE TOGGLE MODE

When Timer1 Gate Toggle mode is enabled, it is possible to measure the full-cycle length of a Timer1 gate signal, as opposed to the duration of a single level pulse.

The Timer1 gate source is routed through a flip-flop that changes state on every incrementing edge of the signal. See Figure 21-4 for timing details.

Timer1 Gate Toggle mode is enabled by setting the T1GTM bit of the T1GCON register. When the T1GTM bit is cleared, the flip-flop is cleared and held clear. This is necessary in order to control which edge is measured.

Note:	Enabling Toggle mode at the same time
	as changing the gate polarity may result in
	indeterminate operation.

21.6.4 TIMER1 GATE SINGLE-PULSE MODE

When Timer1 Gate Single-Pulse mode is enabled, it is possible to capture a single pulse gate event. Timer1 Gate Single-Pulse mode is first enabled by setting the T1GSPM bit in the T1GCON register. Next, the T1GGO/DONE bit in the T1GCON register must be set. The Timer1 will be fully enabled on the next incrementing edge. On the next trailing edge of the pulse, the T1GGO/DONE bit will automatically be cleared. No other gate events will be allowed to increment Timer1 until the T1GGO/DONE bit is once again set in software. See Figure 21-5 for timing details.

If the Single Pulse Gate mode is disabled by clearing the T1GSPM bit in the T1GCON register, the T1GGO/DONE bit should also be cleared.

Enabling the Toggle mode and the Single-Pulse mode simultaneously will permit both sections to work together. This allows the cycle times on the Timer1 Gate source to be measured. See Figure 21-6 for timing details.

21.6.5 TIMER1 GATE VALUE STATUS

When Timer1 gate value status is utilized, it is possible to read the most current level of the gate control value. The value is stored in the T1GVAL bit in the T1GCON register. The T1GVAL bit is valid even when the Timer1 gate is not enabled (TMR1GE bit is cleared).

21.6.6 TIMER1 GATE EVENT INTERRUPT

When Timer1 gate event interrupt is enabled, it is possible to generate an interrupt upon the completion of a gate event. When the falling edge of T1GVAL occurs, the TMR1GIF flag bit in the PIR1 register will be set. If the TMR1GIE bit in the PIE1 register is set, then an interrupt will be recognized.

The TMR1GIF flag bit operates even when the Timer1 gate is not enabled (TMR1GE bit is cleared).

21.7 Timer1 Interrupt

The Timer1 register pair (TMR1H:TMR1L) increments to FFFFh and rolls over to 0000h. When Timer1 rolls over, the Timer1 interrupt flag bit of the PIR1 register is set. To enable the interrupt on rollover, you must set these bits:

- · TMR1ON bit of the T1CON register
- · TMR1IE bit of the PIE1 register
- · PEIE bit of the INTCON register
- · GIE bit of the INTCON register

The interrupt is cleared by clearing the TMR1IF bit in the Interrupt Service Routine.

Note: The TMR1H:TMR1L register pair and the TMR1IF bit should be cleared before enabling interrupts.

21.8 Timer1 Operation During Sleep

Timer1 can only operate during Sleep when setup in Asynchronous Counter mode. In this mode, an external crystal or clock source can be used to increment the counter. To set up the timer to wake the device:

- TMR1ON bit of the T1CON register must be set
- TMR1IE bit of the PIE1 register must be set
- · PEIE bit of the INTCON register must be set
- T1SYNC bit of the T1CON register must be set
- TMR1CS bits of the T1CON register must be configured
- T10SCEN bit of the T1CON register must be configured

The device will wake-up on an overflow and execute the next instructions. If the GIE bit of the INTCON register is set, the device will call the Interrupt Service Routine.

Timer1 oscillator will continue to operate in Sleep regardless of the $\overline{\text{T1SYNC}}$ bit setting.

21.9 ECCP/CCP Capture/Compare Time Base

The CCP modules use the TMR1H:TMR1L register pair as the time base when operating in Capture or Compare mode.

In Capture mode, the value in the TMR1H:TMR1L register pair is copied into the CCPR1H:CCPR1L register pair on a configured event.

In Compare mode, an event is triggered when the value CCPR1H:CCPR1L register pair matches the value in the TMR1H:TMR1L register pair. This event can be a Special Event Trigger.

For more information, see Section 24.0 "Capture/Compare/PWM Modules".

21.10 ECCP/CCP Special Event Trigger

When any of the CCP's are configured to trigger a special event, the trigger will clear the TMR1H:TMR1L register pair. This special event does not cause a Timer1 interrupt. The CCP module may still be configured to generate a CCP interrupt.

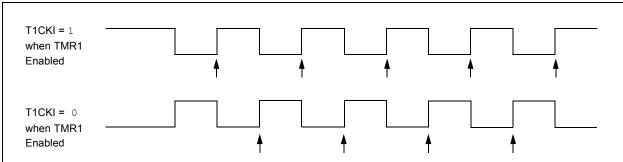
In this mode of operation, the CCPR1H:CCPR1L register pair becomes the period register for Timer1.

Timer1 should be synchronized and FOSC/4 should be selected as the clock source in order to utilize the Special Event Trigger. Asynchronous operation of Timer1 can cause a Special Event Trigger to be missed.

In the event that a write to TMR1H or TMR1L coincides with a Special Event Trigger from the CCP, the write will take precedence.

For more information, see **Section 16.2.5 "Special Event Trigger"**.

FIGURE 21-2: TIMER1 INCREMENTING EDGE



Note 1: Arrows indicate counter increments.

2: In Counter mode, a falling edge must be registered by the counter prior to the first incrementing rising edge of the clock.

FIGURE 21-3: TIMER1 GATE ENABLE MODE

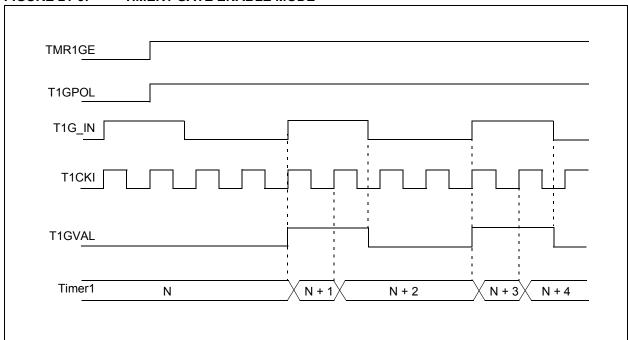
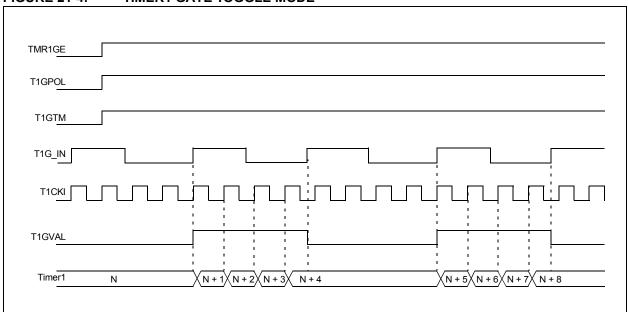
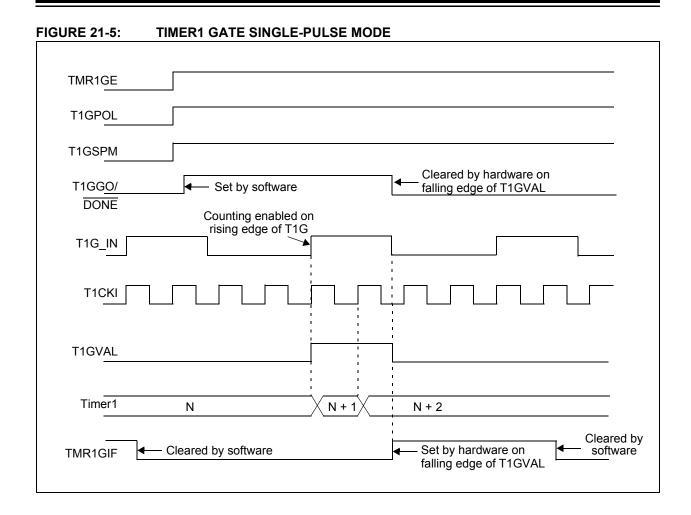
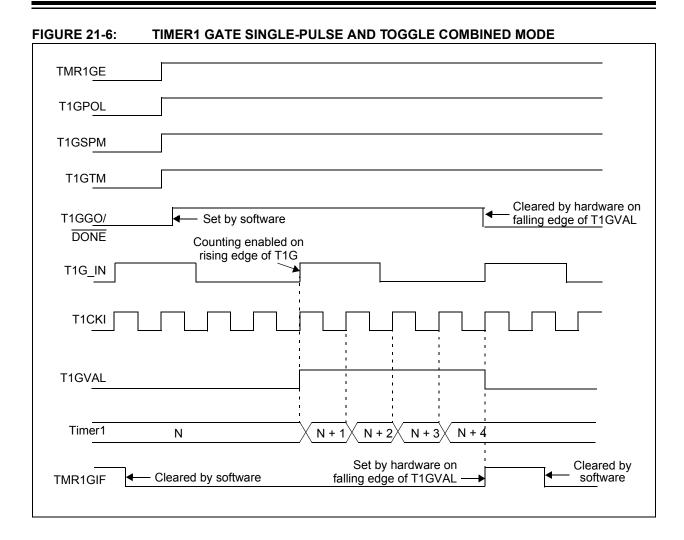


FIGURE 21-4: TIMER1 GATE TOGGLE MODE







21.11 Timer1 Control Register

The Timer1 Control register (T1CON), shown in Register 21-1, is used to control Timer1 and select the various features of the Timer1 module.

REGISTER 21-1: T1CON: TIMER1 CONTROL REGISTER

R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	U-0	R/W-0/u
TMR1C	S<1:0>	T1CKP	S<1:0>	T10SCEN	T1SYNC	_	TMR10N
bit 7							bit 0

Legend:R = Readable bitW = Writable bitU = Unimplemented bit, read as '0'u = Bit is unchangedx = Bit is unknown-n/n = Value at POR and BOR/Value at all other Resets'1' = Bit is set'0' = Bit is cleared

bit 7-6 TMR1CS<1:0>: Timer1 Clock Source Select bits

11 = Timer1 clock source is Capacitive Sensing Oscillator (CAPOSC)

10 = Timer1 clock source is pin or oscillator:

If T1OSCEN = 0:

External clock from T1CKI pin (on the rising edge)

If T10SCEN = 1:

Crystal oscillator on T1OSI/T1OSO pins

01 = Timer1 clock source is system clock (Fosc)

00 = Timer1 clock source is instruction clock (Fosc/4)

bit 5-4 T1CKPS<1:0>: Timer1 Input Clock Prescale Select bits

11 = 1:8 Prescale value

10 = 1:4 Prescale value

01 = 1:2 Prescale value

00 = 1:1 Prescale value

bit 3 T10SCEN: LP Oscillator Enable Control bit

1 = Dedicated Timer1 oscillator circuit enabled

0 = Dedicated Timer1 oscillator circuit disabled

bit 2 T1SYNC: Timer1 External Clock Input Synchronization Control bit

TMR1CS<1:0> = 1X:

1 = Do not synchronize external clock input

0 = Synchronize external clock input with system clock (Fosc)

TMR1CS<1:0> = 0X:

This bit is ignored.

Unimplemented: Read as '0'

bit 0 TMR10N: Timer1 On bit

bit 1

1 = Enables Timer1

0 = Stops Timer1

Clears Timer1 gate flip-flop

21.12 Timer1 Gate Control Register

The Timer1 Gate Control register (T1GCON), shown in Register 21-2, is used to control Timer1 Gate.

REGISTER 21-2: T1GCON: TIMER1 GATE CONTROL REGISTER

R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W/HC-0/u	R-x/x	R/W-0/u	R/W-0/u
TMR1GE	T1GPOL	T1GTM	T1GSPM	T1GGO/ DONE	T1GVAL	T1GS	S<1:0>
bit 7							bit 0

Legend:R = Readable bitW = Writable bitU = Unimplemented bit, read as '0'u = Bit is unchangedx = Bit is unknown-n/n = Value at POR and BOR/Value at all other Resets'1' = Bit is set'0' = Bit is clearedHC = Bit is cleared by hardware

bit 7 TMR1GE: Timer1 Gate Enable bit

 $\frac{\text{If TMR1ON} = 0}{\text{This bit is ignored}}$ $\frac{\text{If TMR1ON} = 1}{\text{If TMR1ON}}$

1 = Timer1 counting is controlled by the Timer1 gate function

0 = Timer1 counts regardless of Timer1 gate function

bit 6 T1GPOL: Timer1 Gate Polarity bit

1 = Timer1 gate is active-high (Timer1 counts when gate is high)0 = Timer1 gate is active-low (Timer1 counts when gate is low)

bit 5 T1GTM: Timer1 Gate Toggle Mode bit

1 = Timer1 Gate Toggle mode is enabled

0 = Timer1 Gate Toggle mode is disabled and toggle flip-flop is cleared

Timer1 gate flip-flop toggles on every rising edge.

bit 4 T1GSPM: Timer1 Gate Single-Pulse Mode bit

1 = Timer1 gate Single-Pulse mode is enabled and is controlling Timer1 gate

0 = Timer1 gate Single-Pulse mode is disabled

bit 3 T1GGO/DONE: Timer1 Gate Single-Pulse Acquisition Status bit

1 = Timer1 gate single-pulse acquisition is ready, waiting for an edge

0 = Timer1 gate single-pulse acquisition has completed or has not been started

bit 2 T1GVAL: Timer1 Gate Current State bit

Indicates the current state of the Timer1 gate that could be provided to TMR1H:TMR1L.

Unaffected by Timer1 Gate Enable (TMR1GE).

bit 1-0 T1GSS<1:0>: Timer1 Gate Source Select bits

00 = Timer1 Gate pin

01 = Timer0 overflow output

10 = Comparator 1 optionally synchronized output (sync_C1OUT)

11 = Comparator 2 optionally synchronized output (sync_C2OUT)

TABLE 21-5: SUMMARY OF REGISTERS ASSOCIATED WITH TIMER1

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ANSELA	_	_	_	ANSA4	_	ANSA2	ANSA1	ANSA0	126
CCP1CON	P1M	<1:0>	DC1B	<1:0>		CCP1N	1<3:0>		236
CCP2CON	P2M⁴	<1:0>	DC2B	<1:0>		CCP2N	1<3:0>		236
INLVLA	_	1	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	127
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	90
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	91
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	95
TMR1H	Holding Regist	er for the Most	Significant Byte	of the 16-bit T	MR1 Register				191*
TMR1L	Holding Regist	er for the Least	Significant Byte	e of the 16-bit 7	MR1 Register				191*
TRISA	_	_	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	125
T1CON	TMR1C	S<1:0>	T1CKP	S<1:0>	T10SCEN	T1SYNC	_	TMR10N	195
T1GCON	TMR1GE	T1GPOL	T1GTM	T1GSPM	T <u>1GGO</u> / DONE	T1GVAL	T1GS	S<1:0>	196

Legend:

Unimplemented location, read as '0'. Shaded cells are not used by the Timer1 module.
 * Page provides register information.

NOTES:

22.0 TIMER2/4/6 MODULES

There are up to three identical Timer2-type modules available. To maintain pre-existing naming conventions, the Timers are called Timer2, Timer4 and Timer6 (also Timer2/4/6).

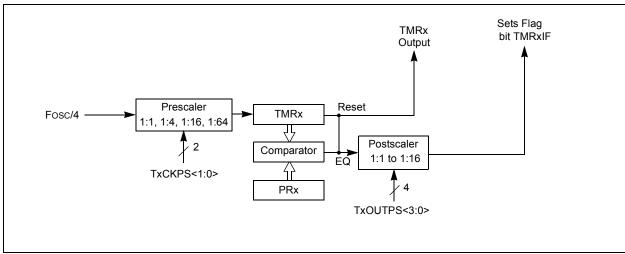
Note: The 'x' variable used in this section is used to designate Timer2, Timer4 or Timer6. For example, TxCON references T2CON, T4CON or T6CON. PRx references PR2, PR4 or PR6.

The Timer2/4/6 modules incorporate the following features:

- 8-bit Timer and Period registers (TMRx and PRx, respectively)
- Readable and writable (both registers)
- Software programmable prescaler (1:1, 1:4, 1:16, and 1:64)
- Software programmable postscaler (1:1 to 1:16)
- Interrupt on TMRx match with PRx, respectively
- Optional use as the shift clock for the MSSPx modules (Timer2 only)

See Figure 22-1 for a block diagram of Timer2/4/6.

FIGURE 22-1: TIMER2/4/6 BLOCK DIAGRAM



22.1 Timer2/4/6 Operation

The clock input to the Timer2/4/6 modules is the system instruction clock (Fosc/4).

TMRx increments from 00h on each clock edge.

A 4-bit counter/prescaler on the clock input allows direct input, divide-by-4 and divide-by-16 prescale options. These options are selected by the prescaler control bits, TxCKPS<1:0> of the TxCON register. The value of TMRx is compared to that of the Period register, PRx, on each clock cycle. When the two values match, the comparator generates a match signal as the timer output. This signal also resets the value of TMRx to 00h on the next cycle and drives the output counter/postscaler (see Section 22.2 "Timer2/4/6 Interrupt").

The TMRx and PRx registers are both directly readable and writable. The TMRx register is cleared on any device Reset, whereas the PRx register initializes to FFh. Both the prescaler and postscaler counters are cleared on the following events:

- · a write to the TMRx register
- · a write to the TxCON register
- · Power-on Reset (POR)
- · Brown-out Reset (BOR)
- MCLR Reset
- · Watchdog Timer (WDT) Reset
- · Stack Overflow Reset
- · Stack Underflow Reset
- RESET Instruction

Note: TMRx is not cleared when TxCON is written.

22.2 Timer2/4/6 Interrupt

Timer2/4/6 can also generate an optional device interrupt. The Timer2/4/6 output signal (TMRx-to-PRx match) provides the input for the 4-bit counter/postscaler. This counter generates the TMRx match interrupt flag which is latched in TMRxIF of the PIRx register. The interrupt is enabled by setting the TMRx Match Interrupt Enable bit, TMRxIE of the PIEx register.

A range of 16 postscale options (from 1:1 through 1:16 inclusive) can be selected with the postscaler control bits, TxOUTPS<3:0>, of the TxCON register.

22.3 Timer2/4/6 Output

The unscaled output of TMRx is available primarily to the CCP modules, where it is used as a time base for operations in PWM mode.

Timer2 can be optionally used as the shift clock source for the MSSPx modules operating in SPI mode. Additional information is provided in Section 25.0 "Master Synchronous Serial Port (MSSP1 and MSSP2) Module".

22.4 Timer2/4/6 Operation During Sleep

The Timer2/4/6 timers cannot be operated while the processor is in Sleep mode. The contents of the TMRx and PRx registers will remain unchanged while the processor is in Sleep mode.

22.5 Timer2 Control Register

REGISTER 22-1: TXCON: TIMER2/TIMER4/TIMER6 CONTROL REGISTER

U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
_	TxOUTPS<3:0>				TMRxON	TxCKP	S<1:0>
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

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

bit 6-3 **TxOUTPS<3:0>:** Timerx Output Postscaler Select bits

0000 = 1:1 Postscaler 0001 = 1:2 Postscaler 0010 = 1:3 Postscaler 0011 = 1:4 Postscaler 0100 = 1:5 Postscaler 0101 = 1:6 Postscaler

0101 = 1:6 Postscaler 0110 = 1:7 Postscaler 0111 = 1:8 Postscaler 1000 = 1:9 Postscaler 1001 = 1:10 Postscaler 1010 = 1:11 Postscaler

1011 = 1:12 Postscaler 1100 = 1:13 Postscaler 1101 = 1:14 Postscaler 1110 = 1:15 Postscaler 1111 = 1:16 Postscaler

bit 2 TMRxON: Timerx On bit

1 = Timerx is on 0 = Timerx is off

bit 1-0 TxCKPS<1:0>: Timer2-type Clock Prescale Select bits

00 = Prescaler is 1 01 = Prescaler is 4 10 = Prescaler is 16 11 = Prescaler is 64

TABLE 22-1: SUMMARY OF REGISTERS ASSOCIATED WITH TIMER2/4/6

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
CCP2CON	P2M	<1:0>	DC2B	<1:0>		CCP2	VI<3:0>		236
CCP4CON	1	ı	DC4B	<1:0>		CCP4	V<3:0>		236
CCP6CON	_	_	DC6B	<1:0>		CCP6	VI<3:0>		236
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	90
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	91
PIE3	_	_	CCP4IE	CCP3IE	TMR6IE	_	TMR4IE	_	93
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	95
PIR3	_	_	CCP4IF	CCP3IF	TMR6IF	_	TMR4IF	_	97
PR2	Timer2 Mo	dule Period	Register						199*
PR4	Timer4 Mo	dule Period	Register						199*
PR6	Timer6 Mod	dule Period	Register						199*
T2CON	_		T2OUTF	PS<3:0>		TMR2ON	T2CKP	S<1:0>	201
T4CON	_		T4OUTF	PS<3:0>		TMR4ON	T4CKP	S<1:0>	201
T6CON	_	T6OUTPS<3:0> TMR6ON T6CKPS<1:0>						201	
TMR2	Holding Register for the 8-bit TMR2 Register								199*
TMR4	Holding Re	gister for the	e 8-bit TMR4	1 Register					199*
TMR6	Holding Re	gister for the	e 8-bit TMR6	6 Register					199*

Legend: — Unimplemented location, read as '0'. Shaded cells are not used for Timer2/4/6 module.

^{*} Page provides register information.

23.0 DATA SIGNAL MODULATOR

The Data Signal Modulator (DSM) is a peripheral which allows the user to mix a data stream, also known as a modulator signal, with a carrier signal to produce a modulated output.

Both the carrier and the modulator signals are supplied to the DSM module either internally, from the output of a peripheral, or externally through an input pin.

The modulated output signal is generated by performing a logical "AND" operation of both the carrier and modulator signals and then provided to the MDOUT pin.

The carrier signal is comprised of two distinct and separate signals. A carrier high (CARH) signal and a carrier low (CARL) signal. During the time in which the modulator (MOD) signal is in a logic high state, the DSM mixes the carrier high signal with the modulator signal. When the modulator signal is in a logic low state, the DSM mixes the carrier low signal with the modulator signal.

Using this method, the DSM can generate the following types of Key Modulation schemes:

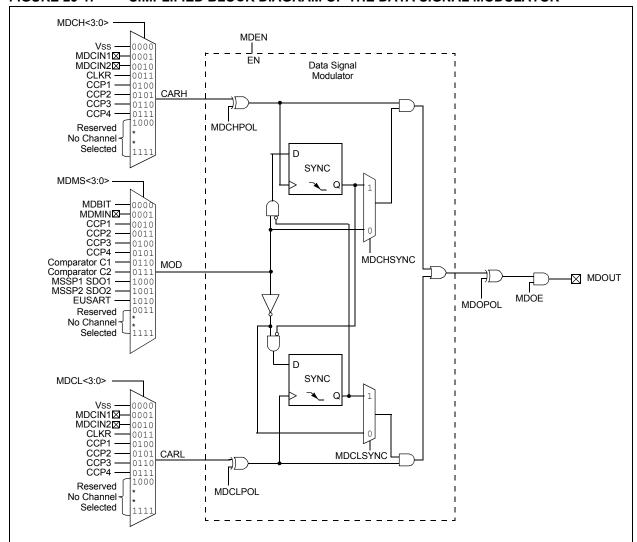
- Frequency-Shift Keying (FSK)
- Phase-Shift Keying (PSK)
- · On-Off Keying (OOK)

Additionally, the following features are provided within the DSM module:

- Carrier Synchronization
- · Carrier Source Polarity Select
- · Carrier Source Pin Disable
- · Programmable Modulator Data
- · Modulator Source Pin Disable
- · Modulated Output Polarity Select
- · Slew Rate Control

Figure 23-1 shows a Simplified Block Diagram of the Data Signal Modulator peripheral.

FIGURE 23-1: SIMPLIFIED BLOCK DIAGRAM OF THE DATA SIGNAL MODULATOR



23.1 DSM Operation

The DSM module can be enabled by setting the MDEN bit in the MDCON register. Clearing the MDEN bit in the MDCON register, disables the DSM module by automatically switching the carrier high and carrier low signals to the Vss signal source. The modulator signal source is also switched to the MDBIT in the MDCON register. This not only assures that the DSM module is inactive, but that it is also consuming the least amount of current

The values used to select the carrier high, carrier low, and modulator sources held by the Modulation Source, Modulation High Carrier, and Modulation Low Carrier control registers are not affected when the MDEN bit is cleared and the DSM module is disabled. The values inside these registers remain unchanged while the DSM is inactive. The sources for the carrier high, carrier low and modulator signals will once again be selected when the MDEN bit is set and the DSM module is again enabled and active.

The modulated output signal can be disabled without shutting down the DSM module. The DSM module will remain active and continue to mix signals, but the output value will not be sent to the MDOUT pin. During the time that the output is disabled, the MDOUT pin will remain low. The modulated output can be disabled by clearing the MDOE bit in the MDCON register.

23.2 Modulator Signal Sources

The modulator signal can be supplied from the following sources:

- CCP1 Signal
- · CCP2 Signal
- · CCP3 Signal
- CCP4 Signal
- MSSP1 SDO1 Signal (SPI Mode Only)
- MSSP2 SDO2 Signal (SPI Mode Only)
- · Comparator C1 Signal
- · Comparator C2 Signal
- EUSART TX Signal
- · External Signal on MDMIN pin
- MDBIT bit in the MDCON register

The modulator signal is selected by configuring the MDMS <3:0> bits in the MDSRC register.

23.3 Carrier Signal Sources

The carrier high signal and carrier low signal can be supplied from the following sources:

- · CCP1 Signal
- · CCP2 Signal
- · CCP3 Signal
- · CCP4 Signal
- · Reference Clock Module Signal
- External Signal on MDCIN1 pin
- · External Signal on MDCIN2 pin
- Vs:

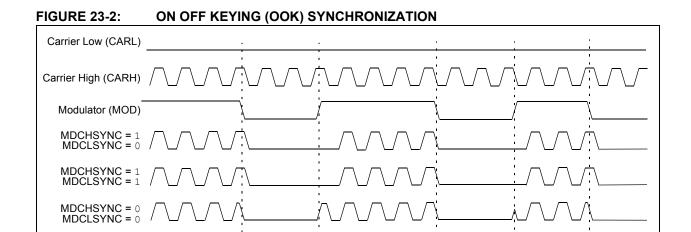
The carrier high signal is selected by configuring the MDCH <3:0> bits in the MDCARH register. The carrier low signal is selected by configuring the MDCL <3:0> bits in the MDCARL register.

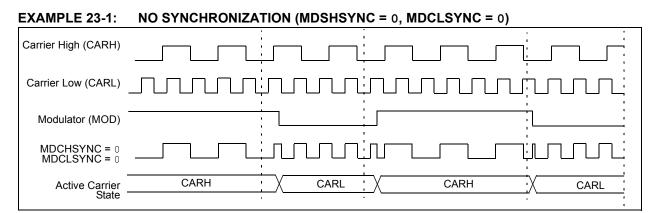
23.4 Carrier Synchronization

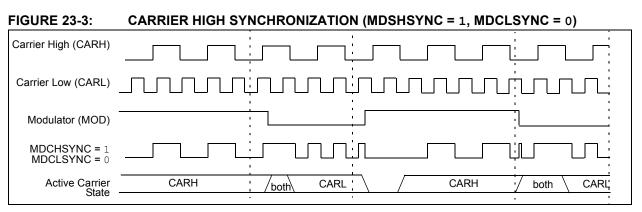
During the time when the DSM switches between carrier high and carrier low signal sources, the carrier data in the modulated output signal can become truncated. To prevent this, the carrier signal can be synchronized to the modulator signal. When synchronization is enabled, the carrier pulse that is being mixed at the time of the transition is allowed to transition low before the DSM switches over to the next carrier source.

Synchronization is enabled separately for the carrier high and carrier low signal sources. Synchronization for the carrier high signal can be enabled by setting the MDCHSYNC bit in the MDCARH register. Synchronization for the carrier low signal can be enabled by setting the MDCLSYNC bit in the MDCARL register.

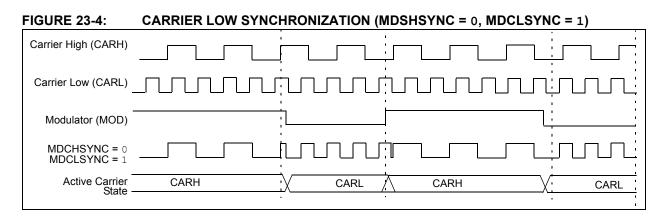
Figure 23-1 through Figure 23-5 show timing diagrams of using various synchronization methods.

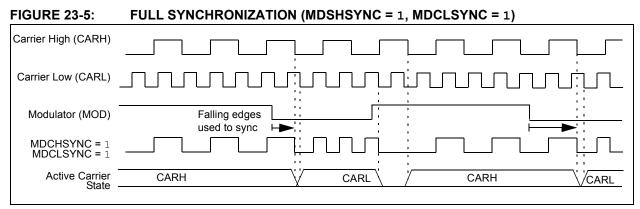






MDCHSYNC = 0 MDCLSYNC = 1





23.5 Carrier Source Polarity Select

The signal provided from any selected input source for the carrier high and carrier low signals can be inverted. Inverting the signal for the carrier high source is enabled by setting the MDCHPOL bit of the MDCARH register. Inverting the signal for the carrier low source is enabled by setting the MDCLPOL bit of the MDCARL register.

23.6 Carrier Source Pin Disable

Some peripherals assert control over their corresponding output pin when they are enabled. For example, when the CCP1 module is enabled, the output of CCP1 is connected to the CCP1 pin.

This default connection to a pin can be disabled by setting the MDCHODIS bit in the MDCARH register for the carrier high source and the MDCLODIS bit in the MDCARL register for the carrier low source.

23.7 Programmable Modulator Data

The MDBIT of the MDCON register can be selected as the source for the modulator signal. This gives the user the ability to program the value used for modulation.

23.8 Modulator Source Pin Disable

The modulator source default connection to a pin can be disabled by setting the MDMSODIS bit in the MDSRC register.

23.9 Modulated Output Polarity

The modulated output signal provided on the MDOUT pin can also be inverted. Inverting the modulated output signal is enabled by setting the MDOPOL bit of the MDCON register.

23.10 Slew Rate Control

The slew rate limitation on the output port pin can be disabled. The slew rate limitation can be removed by clearing the MDSLR bit in the MDCON register.

23.11 Operation in Sleep Mode

The DSM module is not affected by Sleep mode. The DSM can still operate during Sleep, if the Carrier and Modulator input sources are also still operable during Sleep.

23.12 Effects of a Reset

Upon any device Reset, the DSM module is disabled. The user's firmware is responsible for initializing the module before enabling the output. The registers are reset to their default values.

REGISTER 23-1: MDCON: MODULATION CONTROL REGISTER

R/W-0/0	R/W-0/0	R/W-1/1	R/W-0/0	R-0/0	U-0	U-0	R/W-0/0
MDEN	MDOE	MDSLR	MDOPOL MDOUT —		_	_	MDBIT
bit 7							bit 0

Legend:R = Readable bitW = Writable bitU = Unimplemented bit, read as '0'u = Bit is unchangedx = Bit is unknown-n/n = Value at POR and BOR/Value at all other Resets'1' = Bit is set'0' = Bit is cleared

bit 7 MDEN: Modulator Module Enable bit

1 = Modulator module is enabled and mixing input signals 0 = Modulator module is disabled and has no output

bit 6 MDOE: Modulator Module Pin Output Enable bit

1 = Modulator pin output enabled0 = Modulator pin output disabled

bit 5 MDSLR: MDOUT Pin Slew Rate Limiting bit

1 = MDOUT pin slew rate limiting enabled0 = MDOUT pin slew rate limiting disabled

bit 4 MDOPOL: Modulator Output Polarity Select bit

1 = Modulator output signal is inverted0 = Modulator output signal is not inverted

bit 3 MDOUT: Modulator Output bit

Displays the current output value of the modulator module. (1)

bit 2-1 Unimplemented: Read as '0'

bit 0 MDBIT: Allows software to manually set modulation source input to module⁽²⁾

1 = Modulator uses High Carrier source0 = Modulator uses Low Carrier source

Note 1: The modulated output frequency can be greater and asynchronous from the clock that updates this register bit, the bit value may not be valid for higher speed modulator or carrier signals.

2: MDBIT must be selected as the modulation source in the MDSRC register for this operation.

REGISTER 23-2: MDSRC: MODULATION SOURCE CONTROL REGISTER

R/W-x/u	U-0	U-0	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	
MDMSODIS	_	_	_	MDMS<3:0>				
bit 7							bit 0	

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

=	NEW CORP. M. L. C.
bit 7	MDMSODIS: Modulation Source Output Disable bit
	1 = Output signal driving the peripheral output pin (selected by MDMS<3:0>) is disabled
	0 = Output signal driving the peripheral output pin (selected by MDMS<3:0>) is enabled
bit 6-4	Unimplemented: Read as '0'
bit 3-0	MDMS<3:0> Modulation Source Selection bits
	1111 = Reserved. No channel connected.
	1110 = Reserved. No channel connected.
	1101 = Reserved. No channel connected.
	1100 = Reserved. No channel connected.
	1011 = Reserved. No channel connected.
	1010 = EUSART TX output
	1001 = MSSP2 SDO1 output
	1000 = MSSP1 SDO2 output
	0111 = Comparator2 output
	0110 = Comparator1 output
	0101 = CCP4 output (PWM Output mode only)
	0100 = CCP3 output (PWM Output mode only)
	0011 = CCP2 output (PWM Output mode only)
	0010 = CCP1 output (PWM Output mode only)
	0001 = MDMIN port pin
	0000 = MDBIT bit of MDCON register is modulation source

Note 1: Narrowed carrier pulse widths or spurs may occur in the signal stream if the carrier is not synchronized.

REGISTER 23-3: MDCARH: MODULATION HIGH CARRIER CONTROL REGISTER

R/W-x/u	R/W-x/u	R/W-x/u	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
MDCHODIS	MDCHPOL	MDCHSYNC	-	MDCH<3:0>			
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

'1' = Bit is set	'U' = Bit is cleared
bit 7	MDCHODIS: Modulator High Carrier Output Disable bit
	1 = Output signal driving the peripheral output pin (selected by MDCH<3:0>) is disabled0 = Output signal driving the peripheral output pin (selected by MDCH<3:0>) is enabled
bit 6	MDCHPOL: Modulator High Carrier Polarity Select bit
	1 = Selected high carrier signal is inverted
	0 = Selected high carrier signal is not inverted
bit 5	MDCHSYNC: Modulator High Carrier Synchronization Enable bit
	1 = Modulator waits for a falling edge on the high time carrier signal before allowing a switch to the low time carrier
	0 = Modulator Output is not synchronized to the high time carrier signal ⁽¹⁾
bit 4	Unimplemented: Read as '0'
bit 3-0	MDCH<3:0> Modulator Data High Carrier Selection bits (1)
	1111 = Reserved. No channel connected.
	•
	•
	1000 = Reserved. No channel connected.
	0111 = CCP4 output (PWM Output mode only)
	0110 = CCP3 output (PWM Output mode only)
	0101 = CCP2 output (PWM Output mode only)
	0100 = CCP1 output (PWM Output mode only)
	0011 = Reference Clock module signal (CLKR)
	0010 = MDCIN2 port pin
	0001 = MDCIN1 port pin 0000 = Vss
	0000 - V30

Note 1: Narrowed carrier pulse widths or spurs may occur in the signal stream if the carrier is not synchronized.

REGISTER 23-4: MDCARL: MODULATION LOW CARRIER CONTROL REGISTER

R/W-x/u	R/W-x/u	R/W-x/u	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
MDCLODIS	MDCLPOL	MDCLSYNC	-	MDCL<3:0>			
bit 7				bit 0			

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7 MDCLODIS: Modulator Low Carrier Output Disable bit

- 1 = Output signal driving the peripheral output pin (selected by MDCL<3:0> of the MDCARL register) is disabled
- 0 = Output signal driving the peripheral output pin (selected by MDCL<3:0> of the MDCARL register) is enabled
- bit 6 MDCLPOL: Modulator Low Carrier Polarity Select bit
 - 1 = Selected low carrier signal is inverted
 - 0 = Selected low carrier signal is not inverted
- bit 5 MDCLSYNC: Modulator Low Carrier Synchronization Enable bit
 - 1 = Modulator waits for a falling edge on the low time carrier signal before allowing a switch to the high time carrier
 - 0 = Modulator Output is not synchronized to the low time carrier signal⁽¹⁾
- bit 4 Unimplemented: Read as '0'
- bit 3-0 MDCL<3:0> Modulator Data High Carrier Selection bits (1)
 - 1111 = Reserved. No channel connected.

•

_

- 1000 = Reserved. No channel connected.
- 0111 = CCP4 output (PWM Output mode only)
- 0110 = CCP3 output (PWM Output mode only)
- 0101 = CCP2 output (PWM Output mode only)
- 0100 = CCP1 output (PWM Output mode only)
- 0011 = Reference Clock module signal
- 0010 = MDCIN2 port pin
- 0001 = MDCIN1 port pin
- 0000 = Vss

Note 1: Narrowed carrier pulse widths or spurs may occur in the signal stream if the carrier is not synchronized.

TABLE 23-1: SUMMARY OF REGISTERS ASSOCIATED WITH DATA SIGNAL MODULATOR MODE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page	
MDCARH	MDCHODIS	MDCHPOL	MDCHSYNC	_		MDCH<3:0>				
MDCARL	MDCLODIS	MDCLPOL	MDCLSYNC	_		MDCL<3:0>				
MDCON	MDEN	MDOE	MDSLR	MDOPOL	MDOUT MDBIT		208			
MDSRC	MDMSODIS	_	_	_	MDMS<3:0>				209	

Legend: — Unimplemented, read as '0'. Shaded cells are not used in the Data Signal Modulator mode.

NOTES:

24.0 CAPTURE/COMPARE/PWM MODULES

The Capture/Compare/PWM module is a peripheral which allows the user to time and control different events, and to generate Pulse-Width Modulation (PWM) signals. In Capture mode, the peripheral allows the timing of the duration of an event. The Compare mode allows the user to trigger an external event when a predetermined amount of time has expired. The PWM mode can generate Pulse-Width Modulated signals of varying frequency and duty cycle.

This family of devices contains two Enhanced Capture/Compare/PWM modules (ECCP1 and ECCP2) and two standard Capture/Compare/PWM modules (CCP3 and CCP4).

The Capture and Compare functions are identical for all four CCP modules (ECCP1, ECCP2, CCP3 and CCP4). The only differences between CCP modules are in the Pulse-Width Modulation (PWM) function. The standard PWM function is identical in modules, CCP3 and CCP4. In CCP modules ECCP1 and ECCP2, the Enhanced PWM function has slight variations from one another. Full-Bridge ECCP modules have four available I/O pins while Half-Bridge ECCP modules only have two available I/O pins. See Table 24-1 for more information.

- Note 1: In devices with more than one CCP module, it is very important to pay close attention to the register names used. A number placed after the module acronym is used to distinguish between separate modules. For example, the CCP1CON and CCP2CON control the same operational aspects of two completely different CCP modules.
 - 2: Throughout this section, generic references to a CCP module in any of its operating modes may be interpreted as being equally applicable to ECCP1, ECCP2, CCP3 and CCP4. Register names, module signals, I/O pins, and bit names may use the generic designator 'x' to indicate the use of a numeral to distinguish a particular module, when required.

TABLE 24-1: PWM RESOURCES

Device Name	evice Name ECCP1		CCP3	CCP4	
PIC16(L)F1825/1829	Enhanced PWM Full-Bridge	Enhanced PWM Half-Bridge	Standard PWM	Standard PWM	

24.1 Capture Mode

The Capture mode function described in this section is available and identical for CCP modules ECCP1, ECCP2, CCP3 and CCP4.

Capture mode makes use of the 16-bit Timer1 resource. When an event occurs on the CCPx pin, the 16-bit CCPRxH:CCPRxL register pair captures and stores the 16-bit value of the TMR1H:TMR1L register pair, respectively. An event is defined as one of the following and is configured by the CCPxM<3:0> bits of the CCPxCON register:

- · Every falling edge
- · Every rising edge
- · Every 4th rising edge
- · Every 16th rising edge

When a capture is made, the Interrupt Request Flag bit CCPxIF of the PIRx register is set. The interrupt flag must be cleared in software. If another capture occurs before the value in the CCPRxH, CCPRxL register pair is read, the old captured value is overwritten by the new captured value.

Figure 24-1 shows a simplified diagram of the Capture operation.

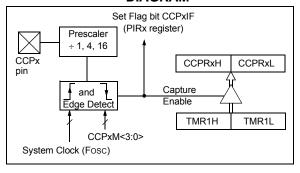
24.1.1 CCP PIN CONFIGURATION

In Capture mode, the CCPx pin should be configured as an input by setting the associated TRIS control bit.

Also, the CCPx pin function can be moved to alternative pins using the APFCON0 or APFCON1 register. Refer to **Section 12.1 "Alternate Pin Function"** for more details.

Note: If the CCPx pin is configured as an output, a write to the port can cause a capture condition.

FIGURE 24-1: CAPTURE MODE OPERATION BLOCK DIAGRAM



24.1.2 TIMER1 MODE RESOURCE

Timer1 must be running in Timer mode or Synchronized Counter mode for the CCP module to use the capture feature. In Asynchronous Counter mode, the capture operation may not work.

See Section 21.0 "Timer1 Module with Gate Control" for more information on configuring Timer1.

24.1.3 SOFTWARE INTERRUPT MODE

When the Capture mode is changed, a false capture interrupt may be generated. The user should keep the CCPxIE interrupt enable bit of the PIEx register clear to avoid false interrupts. Additionally, the user should clear the CCPxIF interrupt flag bit of the PIRx register following any change in Operating mode.

Note: Clocking Timer1 from the system clock (Fosc) should not be used in Capture mode. In order for Capture mode to recognize the trigger event on the CCPx pin, Timer1 must be clocked from the instruction clock (Fosc/4) or from an external clock source.

24.1.4 CCP PRESCALER

There are four prescaler settings specified by the CCPxM<3:0> bits of the CCPxCON register. Whenever the CCP module is turned off, or the CCP module is not in Capture mode, the prescaler counter is cleared. Any Reset will clear the prescaler counter.

Switching from one capture prescaler to another does not clear the prescaler and may generate a false interrupt. To avoid this unexpected operation, turn the module off by clearing the CCPxCON register before changing the prescaler. Equation 24-1 demonstrates the code to perform this function.

EXAMPLE 24-1: CHANGING BETWEEN CAPTURE PRESCALERS

BANKSEL	CCPxCON	;Set Bank bits to point ;to CCPxCON
CLRF	CCPxCON	;Turn CCP module off
MOVLW	NEW CAPT PS	;Load the W reg with
		;the new prescaler
		; move value and CCP ON
MOVWF	CCPxCON	;Load CCPxCON with this
		;value
1		

24.1.5 CAPTURE DURING SLEEP

Capture mode depends upon the Timer1 module for proper operation. There are two options for driving the Timer1 module in Capture mode. It can be driven by the instruction clock (Fosc/4), or by an external clock source.

When Timer1 is clocked by Fosc/4, Timer1 will not increment during Sleep. When the device wakes from Sleep, Timer1 will continue from its previous state.

Capture mode will operate during Sleep when Timer1 is clocked by an external clock source.

24.1.6 ALTERNATE PIN LOCATIONS

This module incorporates I/O pins that can be moved to other locations with the use of the alternate pin function registers, APFCON0 and APFCON1. To determine which pins can be moved and what their default locations are upon a Reset, see Section 12.1 "Alternate Pin Function" for more information.

TABLE 24-2: SUMMARY OF REGISTERS ASSOCIATED WITH CAPTURE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
APFCON1	_	_	SDO2SEL ⁽²⁾	SS2SEL ⁽²⁾	P1DSEL	P1CSEL	P2BSEL	CCP2SEL	122
CCP1CON	P1M<1:0> ⁽¹⁾		DC1B<1:0>		CCP1M<3:0>				236
CCP2CON	P2M<1:0> ⁽¹⁾		DC2B<1:0>		CCP2M<3:0>				236
CCP3CON	_	_	- DC3B<1:0>		CCP3M<3:0>				236
CCP4CON	_	_	DC4B<1:0>		CCP4M<3:0>				236
CCPRxL	Capture/Compare/PWM Register x Low Byte (LSB)							214*	
CCPRxH	Capture/Compare/PWM Register x High Byte (MSB)							214*	
CMxCON0	CxON	CxOUT	CxOE	CxPOL	_	CxSP	CxHYS	CxSYNC	179
CMxCON1	CxINTP	CxINTN	CxPCl	H<1:0>	_	_	CxNC	H<1:0>	180
INLVLA	_	_	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	127
INLVLC	INLVLC7 ⁽²⁾	INLVLC6 ⁽²⁾	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	138
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	90
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	91
PIE2	OSFIE	C2IE	C1IE	EEIE	BCL1IE	_	_	CCP2IE	92
PIE3	_	_	CCP4IE	CCP3IE	TMR6IE	_	TMR4IE	_	93
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	95
PIR2	OSFIF	C2IF	C1IF	EEIF	BCL1IF	_	_	CCP2IF	96
PIR3	_	_	CCP4IF	CCP3IF	TMR6IF	_	TMR4IF	_	97
T1CON	TMR10	S<1:0>	T1CKP	S<1:0>	T1OSCEN	T1SYNC	_	TMR10N	195
T1GCON	TMR1GE	T1GPOL	T1GTM	T1GSPM	T1GGO/DONE	T1GVAL	T1GSS<1:0>		196
TMR1L	Holding Register for the Least Significant Byte of the 16-bit TMR1 Register					191*			
TMR1H	Holding Register for the Most Significant Byte of the 16-bit TMR1 Register					191*			
TRISA	_	_	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	125
TRISC	TRISC7 ⁽²⁾	TRISC6 ⁽²⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	136

Legend: — Unimplemented location, read as '0'. Shaded cells are not used by Capture mode.

* Page provides register information.

Note 1: Applies to ECCP modules only.

2: PIC16(L)F1829 only.

24.2 Compare Mode

The Compare mode function described in this section is available and identical for CCP modules ECCP1, ECCP2, CCP3 and CCP4.

Compare mode makes use of the 16-bit Timer1 resource. The 16-bit value of the CCPRxH:CCPRxL register pair is constantly compared against the 16-bit value of the TMR1H:TMR1L register pair. When a match occurs, one of the following events can occur:

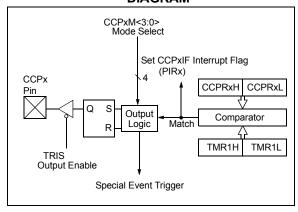
- · Toggle the CCPx output
- · Set the CCPx output
- · Clear the CCPx output
- · Generate a Special Event Trigger
- · Generate a Software Interrupt

The action on the pin is based on the value of the CCPxM<3:0> control bits of the CCPxCON register. At the same time, the interrupt flag CCPxIF bit is set.

All Compare modes can generate an interrupt.

Figure 24-2 shows a simplified diagram of the Compare operation.

FIGURE 24-2: COMPARE MODE OPERATION BLOCK DIAGRAM



24.2.1 CCP PIN CONFIGURATION

The user must configure the CCPx pin as an output by clearing the associated TRIS bit.

Also, the CCPx pin function can be moved to alternative pins using the APFCON0 or APFCON1 register. Refer to **Section 12.1 "Alternate Pin Function"** for more details.

Note: Clearing the CCPxCON register will force the CCPx compare output latch to the default low level. This is not the PORT I/O data latch.

24.2.2 TIMER1 MODE RESOURCE

In Compare mode, Timer1 must be running in either Timer mode or Synchronized Counter mode. The compare operation may not work in Asynchronous Counter mode.

See Section 21.0 "Timer1 Module with Gate Control" for more information on configuring Timer1.

: Clocking Timer1 from the system clock (Fosc) should not be used in Compare mode. In order for Compare mode to recognize the trigger event on the CCPx pin, Tlmer1 must be clocked from the instruction clock (Fosc/4) or from an external clock source.

24.2.3 SOFTWARE INTERRUPT MODE

When Generate Software Interrupt mode is chosen (CCPxM<3:0> = 1010), the CCPx module does not assert control of the CCPx pin (see the CCPxCON register).

24.2.4 SPECIAL EVENT TRIGGER

When Special Event Trigger mode is chosen (CCPxM<3:0> = 1011), the CCPx module does the following:

- · Resets Timer1
- · Starts an ADC conversion if ADC is enabled

The CCPx module does not assert control of the CCPx pin in this mode.

The Special Event Trigger output of the CCP occurs immediately upon a match between the TMR1H, TMR1L register pair and the CCPRxH, CCPRxL register pair. The TMR1H, TMR1L register pair is not reset until the next rising edge of the Timer1 clock. The Special Event Trigger output starts an A/D conversion (if the A/D module is enabled). This allows the CCPRxH, CCPRxL register pair to effectively provide a 16-bit programmable period register for Timer1.

TABLE 24-3: SPECIAL EVENT TRIGGER

Device	CCPx/ECCPx
PIC16(L)F1825/1829	CCP4

Refer to Section 16.2.5 "Special Event Trigger" for more information.

- Note 1: The Special Event Trigger from the CCP module does not set interrupt flag bit TMR1IF of the PIR1 register.
 - 2: Removing the match condition by changing the contents of the CCPRxH and CCPRxL register pair, between the clock edge that generates the Special Event Trigger and the clock edge that generates the Timer1 Reset, will preclude the Reset from occurring.

24.2.5 COMPARE DURING SLEEP

The Compare mode is dependent upon the system clock (Fosc) for proper operation. Since Fosc is shut down during Sleep mode, the Compare mode will not function properly during Sleep.

24.2.6 ALTERNATE PIN LOCATIONS

This module incorporates I/O pins that can be moved to other locations with the use of the alternate pin function registers, APFCON0 and APFCON1. To determine which pins can be moved and what their default locations are upon a Reset, see Section 12.1 "Alternate Pin Function" for more information.

TABLE 24-4: SUMMARY OF REGISTERS ASSOCIATED WITH COMPARE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page	
APFCON1	_	-	SDO2SEL ⁽²⁾	SS2SEL ⁽²⁾	P1DSEL	P1CSEL	P2BSEL	CCP2SEL	122	
CCP1CON	P1M<	1:0> ⁽¹⁾	DC1B	<1:0>		CCP1M<3:0>				
CCP2CON	P2M<	1:0> ⁽¹⁾	DC2B	<1:0>		CCP2M<	3:0>		236	
CCP3CON	_	_	DC3B	<1:0>		CCP3M<3:0>				
CCP4CON	_	_	DC4B	<1:0>		CCP4M<	3:0>		236	
CCPRxL	Capture/Com	pare/PWM Reg	gister x Low By	te (LSB)					214*	
CCPRxH	Capture/Com	pare/PWM Reg	gister x High By	yte (MSB)					214*	
INLVLA	_	_	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	127	
INLVLC	INLVLC7 ⁽²⁾	INLVLC6 ⁽²⁾	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	138	
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	90	
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	91	
PIE2	OSFIE	C2IE	C1IE	EEIE	BCL1IE	ı	1	CCP2IE	92	
PIE3	_	ı	CCP4IE	CCP3IE	TMR6IE	1	TMR4IE	_	93	
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	95	
PIR2	OSFIF	C2IF	C1IF	EEIF	BCL1IF	ı	ı	CCP2IF	96	
PIR3	_	1	CCP4IF	CCP3IF	TMR6IF	-	TMR4IF	_	97	
T1CON	TMR10	S<1:0>	T1CKP	S<1:0>	T1OSCEN	T1SYNC	_	TMR10N	195	
T1GCON	TMR1GE	T1GPOL	T1GTM	T1GSPM	T1GGO/DONE	T1GVAL	T1GS	S<1:0>	196	
TMR1L	Holding Regis	ter for the Lea	st Significant B	syte of the 16-b	oit TMR1 Register				191*	
TMR1H	Holding Register for the Most Significant Byte of the 16-bit TMR1 Register									
TRISA	_	_	_ TRISA5 TRISA4 TRISA3 TRISA2 TRISA1 TRISA0						125	
TRISC	TRISC7 ⁽²⁾	TRISC6 ⁽²⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	136	

Legend: — Unimplemented location, read as '0'. Shaded cells are not used by Compare mode.

Applies to ECCP modules only. PIC16(L)F1829 only. PIC16(L)F1825 only. Note 1:

2:

3:

^{*} Page provides register information.

24.3 PWM Overview

Pulse-Width Modulation (PWM) is a scheme that provides power to a load by switching quickly between fully on and fully off states. The PWM signal resembles a square wave where the high portion of the signal is considered the on state and the low portion of the signal is considered the off state. The high portion, also known as the pulse width, can vary in time and is defined in steps. A larger number of steps applied, which lengthens the pulse width, also supplies more power to the load. Lowering the number of steps applied, which shortens the pulse width, supplies less power. The PWM period is defined as the duration of one complete cycle or the total amount of on and off time combined.

PWM resolution defines the maximum number of steps that can be present in a single PWM period. A higher resolution allows for more precise control of the pulse width time and in turn the power that is applied to the load.

The term duty cycle describes the proportion of the on time to the off time and is expressed in percentages, where 0% is fully off and 100% is fully on. A lower duty cycle corresponds to less power applied and a higher duty cycle corresponds to more power applied.

Figure 24-3 shows a typical waveform of the PWM signal.

24.3.1 STANDARD PWM OPERATION

The standard PWM function described in this section is available and identical for CCP modules ECCP1, ECCP2, CCP3 and CCP4.

The standard PWM mode generates a Pulse-Width Modulation (PWM) signal on the CCPx pin with up to 10 bits of resolution. The period, duty cycle, and resolution are controlled by the following registers:

- · PRx registers
- TxCON registers
- · CCPRxL registers
- · CCPxCON registers

Figure 24-4 shows a simplified block diagram of PWM operation.

- **Note 1:** The corresponding TRIS bit must be cleared to enable the PWM output on the CCPx pin.
 - **2:** Clearing the CCPxCON register will relinquish control of the CCPx pin.

FIGURE 24-3: CCP PWM OUTPUT SIGNAL

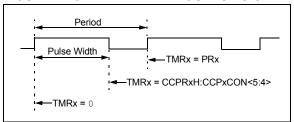
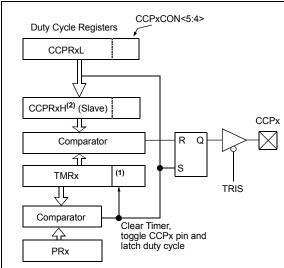


FIGURE 24-4: SIMPLIFIED PWM BLOCK DIAGRAM



- Note 1: The 8-bit timer TMRx register is concatenated with the 2-bit internal system clock (Fosc), or two bits of the prescaler, to create the 10-bit time base
 - 2: In PWM mode, CCPRxH is a read-only register.

24.3.2 SETUP FOR PWM OPERATION

The following steps should be taken when configuring the CCP module for standard PWM operation:

- 1. Disable the CCPx pin output driver by setting the associated TRIS bit.
- Load the PRx register with the PWM period value
- 3. Configure the CCP module for the PWM mode by loading the CCPxCON register with the appropriate values.
- Load the CCPRxL register and the DCxBx bits of the CCPxCON register, with the PWM duty cycle value.
- 5. Configure and start Timer2/4/6:
 - Select the Timer2/4/6 resource to be used for PWM generation by setting the CxTSEL<1:0> bits in the CCPTMRS register.
 - •Clear the TMRxIF interrupt flag bit of the PIRx register. See Note below.
 - •Configure the TxCKPS bits of the TxCON register with the Timer prescale value.
 - •Enable the Timer by setting the TMRxON bit of the TxCON register.
- 6. Enable PWM output pin:
 - Wait until the Timer overflows and the TMRxIF bit of the PIRx register is set. See Note below.
 - •Enable the CCPx pin output driver by clearing the associated TRIS bit.

Note: In order to send a complete duty cycle and period on the first PWM output, the above steps must be included in the setup sequence. If it is not critical to start with a complete PWM signal on the first output, then step 6 may be ignored.

24.3.3 TIMER2/4/6 TIMER RESOURCE

The PWM standard mode makes use of one of the 8-bit Timer2/4/6 timer resources to specify the PWM period.

Configuring the CxTSEL<1:0> bits in the CCPTMRS register selects which Timer2/4/6 timer is used.

24.3.4 PWM PERIOD

The PWM period is specified by the PRx register of Timer2/4/6. The PWM period can be calculated using the formula of Equation 24-1.

EQUATION 24-1: PWM PERIOD

$$PWM \ Period = [(PRx) + 1] \bullet 4 \bullet TOSC \bullet$$

 $(TMRx \ Prescale \ Value)$

Note 1: Tosc = 1/Fosc

When TMRx is equal to PRx, the following three events occur on the next increment cycle:

· TMRx is cleared

Note:

- The CCPx pin is set. (Exception: If the PWM duty cycle = 0%, the pin will not be set.)
- The PWM duty cycle is latched from CCPRxL into CCPRxH.

The Timer postscaler (see Section 22.1 "Timer2/4/6 Operation") is not used in the determination of the PWM frequency.

24.3.5 PWM DUTY CYCLE

The PWM duty cycle is specified by writing a 10-bit value to multiple registers: CCPRxL register and DCxB<1:0> bits of the CCPxCON register. The CCPRxL contains the eight MSbs and the DCxB<1:0> bits of the CCPxCON register contain the two LSbs. CCPRxL and DCxB<1:0> bits of the CCPxCON register can be written to at any time. The duty cycle value is not latched into CCPRxH until after the period completes (i.e., a match between PRx and TMRx registers occurs). While using the PWM, the CCPRxH register is read-only.

Equation 24-2 is used to calculate the PWM pulse width.

Equation 24-3 is used to calculate the PWM duty cycle ratio.

EQUATION 24-2: PULSE WIDTH

 $Pulse\ Width\ =\ (CCPRxL:CCPxCON<5:4>) \bullet$ $TOSC\ \bullet\ (TMRx\ Prescale\ Value)$

EQUATION 24-3: DUTY CYCLE RATIO

 $Duty\ Cycle\ Ratio\ =\ \frac{(CCPRxL:CCPxCON<5:4>)}{4(PRx+I)}$

The CCPRxH register and a 2-bit internal latch are used to double buffer the PWM duty cycle. This double buffering is essential for glitchless PWM operation.

The 8-bit timer TMRx register is concatenated with either the 2-bit internal system clock (Fosc), or two bits of the prescaler, to create the 10-bit time base. The system clock is used if the Timer2/4/6 prescaler is set to 1:1.

When the 10-bit time base matches the CCPRxH and 2-bit latch, then the CCPx pin is cleared (see Figure 24-4).

24.3.6 PWM RESOLUTION

The resolution determines the number of available duty cycles for a given period. For example, a 10-bit resolution will result in 1024 discrete duty cycles, whereas an 8-bit resolution will result in 256 discrete duty cycles.

The maximum PWM resolution is ten bits when PRx is 255. The resolution is a function of the PRx register value as shown by Equation 24-4.

EQUATION 24-4: PWM RESOLUTION

Resolution =
$$\frac{log[4(PRx + 1)]}{log(2)}$$
 bits

Note: If the pulse width value is greater than the period the assigned PWM pin(s) will remain unchanged.

TABLE 24-5: EXAMPLE PWM FREQUENCIES AND RESOLUTIONS (Fosc = 32 MHz)

PWM Frequency	1.95 kHz	7.81 kHz	31.25 kHz	125 kHz	250 kHz	333.3 kHz
Timer Prescale (1, 4, 16)	16	4	1	1	1	1
PRx Value	0xFF	0xFF	0xFF	0x3F	0x1F	0x17
Maximum Resolution (bits)	10	10	10	8	7	6.6

TABLE 24-6: EXAMPLE PWM FREQUENCIES AND RESOLUTIONS (Fosc = 20 MHz)

PWM Frequency	1.22 kHz	4.88 kHz	19.53 kHz	78.12 kHz	156.3 kHz	208.3 kHz
Timer Prescale (1, 4, 16)	16	4	1	1	1	1
PRx Value	0xFF	0xFF	0xFF	0x3F	0x1F	0x17
Maximum Resolution (bits)	10	10	10	8	7	6.6

TABLE 24-7: EXAMPLE PWM FREQUENCIES AND RESOLUTIONS (Fosc = 8 MHz)

PWM Frequency	1.22 kHz	4.90 kHz	19.61 kHz	76.92 kHz	153.85 kHz	200.0 kHz
Timer Prescale (1, 4, 16)	16	4	1	1	1	1
PRx Value	0x65	0x65	0x65	0x19	0x0C	0x09
Maximum Resolution (bits)	8	8	8	6	5	5

24.3.7 OPERATION IN SLEEP MODE

In Sleep mode, the TMRx register will not increment and the state of the module will not change. If the CCPx pin is driving a value, it will continue to drive that value. When the device wakes up, TMRx will continue from its previous state.

24.3.8 CHANGES IN SYSTEM CLOCK **FREQUENCY**

The PWM frequency is derived from the system clock frequency. Any changes in the system clock frequency will result in changes to the PWM frequency. See Section 5.0 "Oscillator Module (With Fail-Safe **Clock Monitor)**" for additional details.

24.3.9 **EFFECTS OF RESET**

Any Reset will force all ports to Input mode and the CCP registers to their Reset states.

24.3.10 ALTERNATE PIN LOCATIONS

This module incorporates I/O pins that can be moved to other locations with the use of the alternate pin function registers, APFCON0 and APFCON1. To determine which pins can be moved and what their default locations are upon a Reset, see Section 12.1 "Alternate Pin Function" for more information.

TABLE 24-8: SUMMARY OF REGISTERS ASSOCIATED WITH STANDARD PWM

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
APFCON1	_	_	SDO2SEL ⁽²⁾	SS2SEL ⁽²⁾	P1DSEL	P1CSEL	P2BSEL	CCP2SEL	122
CCP1CON	P1M<	1:0> ⁽¹⁾	DC1B	<1:0>		•	236		
CCP2CON	P2M<	P2M<1:0> ⁽¹⁾		<1:0>		CCP2I	236		
CCP3CON	_	_	DC3B<1:0>			CCP3I	M<3:0>		236
CCP4CON	_	_	DC4B<1:0>			CCP4I	M<3:0>		236
CCPTMRS	C4TSE	L<1:0>	<1:0> C3TSEL<1:0>			L<1:0>	C1TSE	L<1:0>	237
CCPR1L	Capture/Comp	oare/PWM Regi	ster x Low Byte	(LSB)					214*
INLVLA	_	_	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	127
INLVLC	INLVLC7 ⁽²⁾	INLVLC6 ⁽²⁾	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	138
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	90
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	91
PIE2	OSFIE	C2IE	C1IE EEIE		BCL1IE	_	_	CCP2IE	92
PIE3	_	_	CCP4IE	CCP3IE	TMR6IE	_	TMR4IE	_	93
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	95
PIR2	OSFIF	C2IF	C1IF	EEIF	BCL1IF	_	_	CCP2IF	96
PIR3	_	_	CCP4IF	CCP3IF	TMR6IF	_	TMR4IF	_	97
PRx	Timer2/4/6 Pe	riod Register							199*
T2CON	_		T2OUTF	PS<3:0>		TMR2ON	T2CKP	PS<1:0>	201
T4CON	_		T4OUTF	PS<3:0>		TMR4ON	T4CKP	'S<1:0>	201
T6CON	_		T6OUTPS<3:0> TMR6ON T6CK						201
TMRx	Timer2/4/6 Mo	dule Register							199*
TRISA		_	TRISA5 TRISA4 TRISA3 TRISA2 TRISA1 TRISA0						125
TRISC	TRISC7 ⁽²⁾	TRISC6 ⁽²⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	136

Legend: — Unimplemented location, read as '0'. Shaded cells are not used by the PWM.

Applies to ECCP modules only. PIC16(L)F1829 only.

Page provides register information.

24.4 PWM (Enhanced Mode)

The enhanced PWM function described in this section is available for CCP modules ECCP1 and ECCP2, with any differences between modules noted.

The enhanced PWM mode generates a Pulse-Width Modulation (PWM) signal on up to four different output pins with up to 10 bits of resolution. The period, duty cycle, and resolution are controlled by the following registers:

- · PRx registers
- · TxCON registers
- · CCPRxL registers
- CCPxCON registers

The ECCP modules have the following additional PWM registers which control Auto-shutdown, Auto-restart, Dead-band Delay and PWM Steering modes:

- · CCPxAS registers
- · PSTRxCON registers
- · PWMxCON registers

The enhanced PWM module can generate the following five PWM Output modes:

- · Single PWM
- · Half-Bridge PWM
- · Full-Bridge PWM, Forward mode
- · Full-Bridge PWM, Reverse mode
- · Single PWM with PWM Steering mode

To select an Enhanced PWM Output mode, the PxM bits of the CCPxCON register must be configured appropriately.

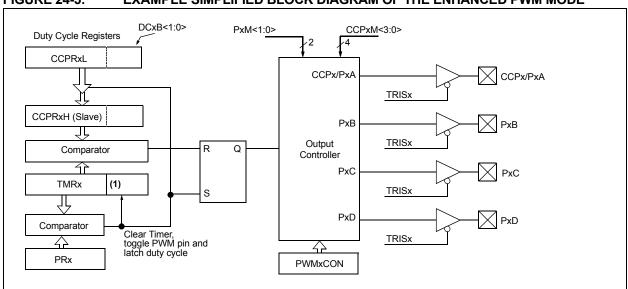
The PWM outputs are multiplexed with I/O pins and are designated PxA, PxB, PxC and PxD. The polarity of the PWM pins is configurable and is selected by setting the CCPxM bits in the CCPxCON register appropriately.

Figure 24-5 shows an example of a simplified block diagram of the Enhanced PWM module.

Figure 24-8 shows the pin assignments for various Enhanced PWM modes.

- Note 1: The corresponding TRIS bit must be cleared to enable the PWM output on the CCPx pin.
 - **2:** Clearing the CCPxCON register will relinquish control of the CCPx pin.
 - **3:** Any pin not used in the enhanced PWM mode is available for alternate pin functions, if applicable.
 - **4:** To prevent the generation of an incomplete waveform when the PWM is first enabled, the ECCP module waits until the start of a new PWM period before generating a PWM signal.

FIGURE 24-5: EXAMPLE SIMPLIFIED BLOCK DIAGRAM OF THE ENHANCED PWM MODE



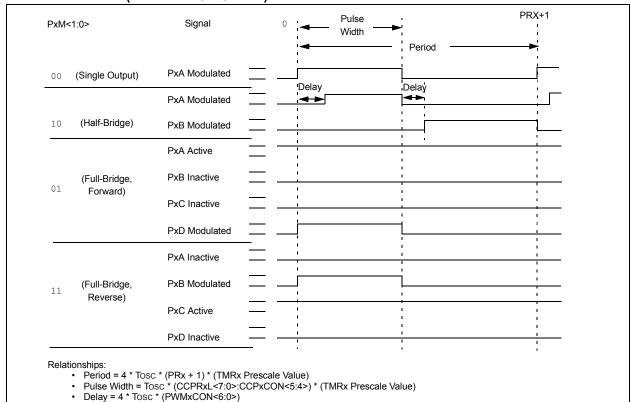
Note 1: The 8-bit timer TMRx register is concatenated with the 2-bit internal Q clock, or two bits of the prescaler to create the 10-bit time base.

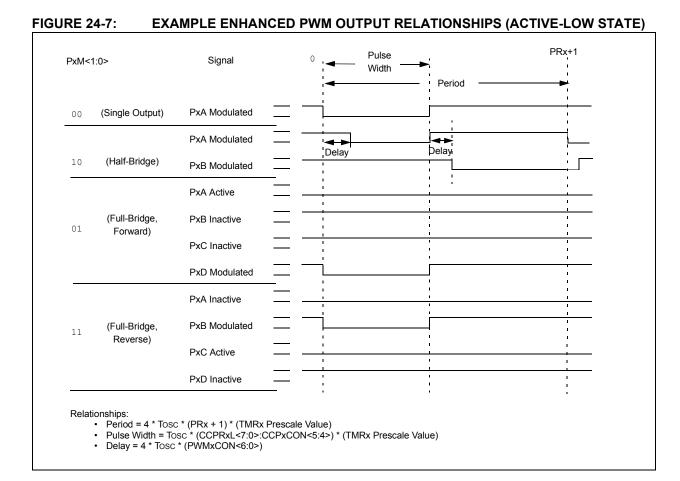
TABLE 24-9: EXAMPLE PIN ASSIGNMENTS FOR VARIOUS PWM ENHANCED MODES

ECCP Mode	PxM<1:0>	CCPx/PxA	PxB	PxC	PxD
Single	00	Yes ⁽¹⁾	Yes ⁽¹⁾	Yes ⁽¹⁾	Yes ⁽¹⁾
Half-Bridge	10	Yes	Yes	No	No
Full-Bridge, Forward	01	Yes	Yes	Yes	Yes
Full-Bridge, Reverse	11	Yes	Yes	Yes	Yes

Note 1: PWM Steering enables outputs in Single mode.

FIGURE 24-6: EXAMPLE PWM (ENHANCED MODE) OUTPUT RELATIONSHIPS (ACTIVE-HIGH STATE)





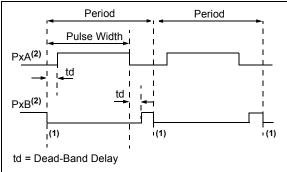
24.4.1 HALF-BRIDGE MODE

In Half-Bridge mode, two pins are used as outputs to drive push-pull loads. The PWM output signal is output on the CCPx/PxA pin, while the complementary PWM output signal is output on the PxB pin (see Figure 24-9). This mode can be used for half-bridge applications, as shown in Figure 24-9, or for full-bridge applications, where four power switches are being modulated with two PWM signals.

In Half-Bridge mode, the programmable dead-band delay can be used to prevent shoot-through current in half-bridge power devices. The value of the PDC<6:0> bits of the PWMxCON register sets the number of instruction cycles before the output is driven active. If the value is greater than the duty cycle, the corresponding output remains inactive during the entire cycle. See Section 24.4.5 "Programmable Dead-Band Delay Mode" for more details of the dead-band delay operations.

Since the PxA and PxB outputs are multiplexed with the PORT data latches, the associated TRIS bits must be cleared to configure PxA and PxB as outputs.

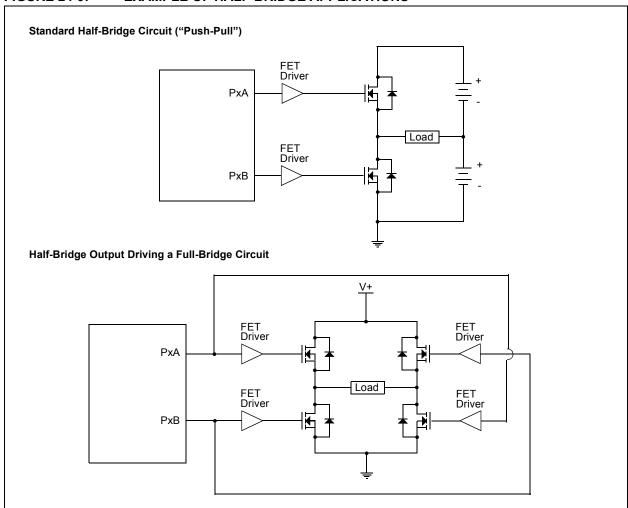
FIGURE 24-8: EXAMPLE OF HALF-BRIDGE PWM OUTPUT



Note 1: At this time, the TMRx register is equal to the PRx register.

2: Output signals are shown as active-high.

FIGURE 24-9: EXAMPLE OF HALF-BRIDGE APPLICATIONS



24.4.2 FULL-BRIDGE MODE

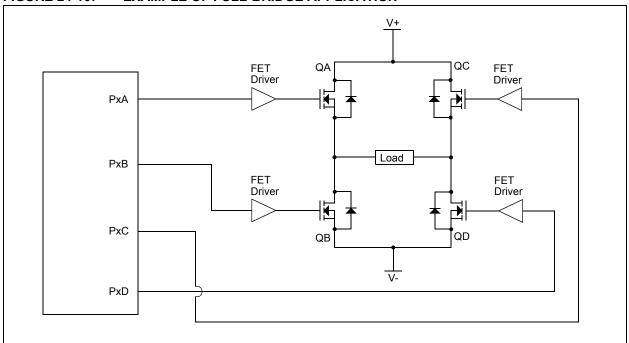
In Full-Bridge mode, all four pins are used as outputs. An example of full-bridge application is shown in Figure 24-10.

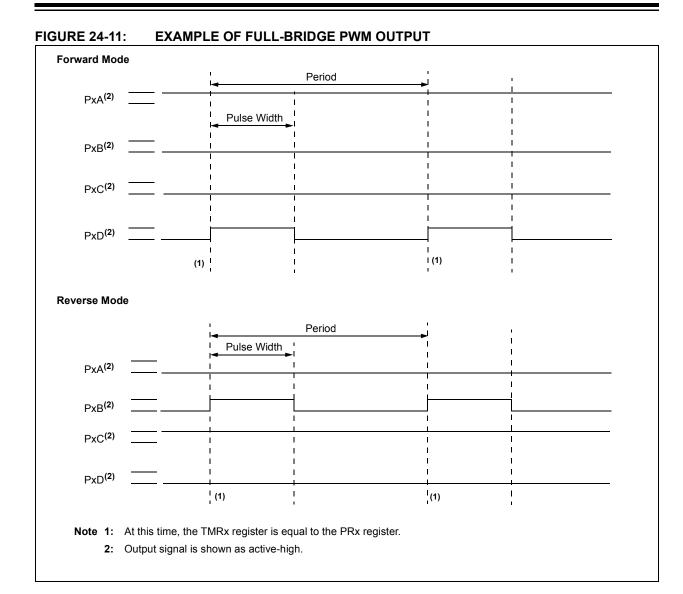
In the Forward mode, pin CCPx/PxA is driven to its active state, pin PxD is modulated, while PxB and PxC will be driven to their inactive state as shown in Figure 24-11.

In the Reverse mode, PxC is driven to its active state, pin PxB is modulated, while PxA and PxD will be driven to their inactive state as shown Figure 24-11.

PxA, PxB, PxC and PxD outputs are multiplexed with the PORT data latches. The associated TRIS bits must be cleared to configure the PxA, PxB, PxC and PxD pins as outputs.

FIGURE 24-10: EXAMPLE OF FULL-BRIDGE APPLICATION





24.4.2.1 Direction Change in Full-Bridge Mode

In the Full-Bridge mode, the PxM1 bit in the CCPxCON register allows users to control the forward/reverse direction. When the application firmware changes this direction control bit, the module will change to the new direction on the next PWM cycle.

A direction change is initiated in software by changing the PxM1 bit of the CCPxCON register. The following sequence occurs four Timer cycles prior to the end of the current PWM period:

- The modulated outputs (PxB and PxD) are placed in their inactive state.
- The associated unmodulated outputs (PxA and PxC) are switched to drive in the opposite direction.
- PWM modulation resumes at the beginning of the next period.

See Figure 24-12 for an illustration of this sequence.

The Full-Bridge mode does not provide dead-band delay. As one output is modulated at a time, dead-band delay is generally not required. There is a situation where dead-band delay is required. This situation occurs when both of the following conditions are true:

- The direction of the PWM output changes when the duty cycle of the output is at or near 100%.
- The turn-off time of the power switch, including the power device and driver circuit, is greater than the turn-on time.

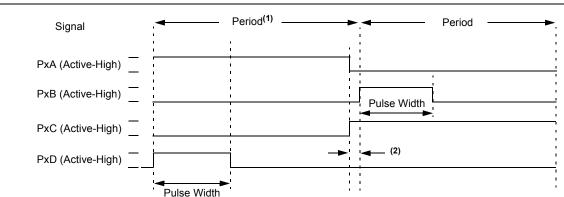
Figure 24-13 shows an example of the PWM direction changing from forward to reverse, at a near 100% duty cycle. In this example, at time t1, the output PxA and PxD become inactive, while output PxC becomes active. Since the turn-off time of the power devices is longer than the turn-on time, a shoot-through current will flow through power devices QC and QD (see Figure 24-10) for the duration of 't'. The same phenomenon will occur to power devices QA and QB for PWM direction change from reverse to forward.

If changing PWM direction at high duty cycle is required for an application, two possible solutions for eliminating the shoot-through current are:

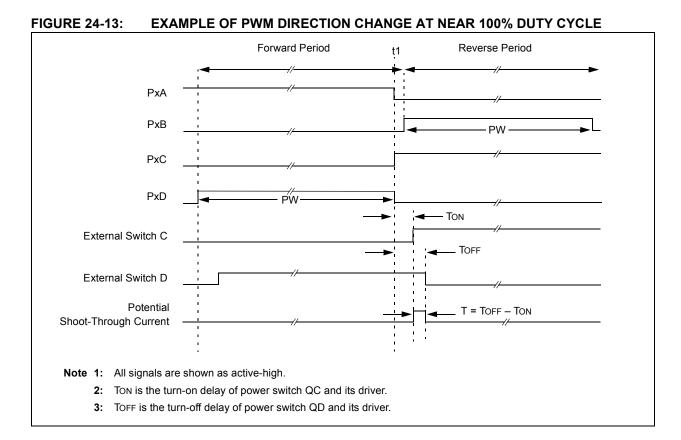
- Reduce PWM duty cycle for one PWM period before changing directions.
- 2. Use switch drivers that can drive the switches off faster than they can drive them on.

Other options to prevent shoot-through current may exist.





- Note 1: The direction bit PxM1 of the CCPxCON register is written any time during the PWM cycle.
 - 2: When changing directions, the PxA and PxC signals switch before the end of the current PWM cycle. The modulated PxB and PxD signals are inactive at this time. The length of this time is four Timer counts.



© 2010-2012 Microchip Technology Inc.

24.4.3 ENHANCED PWM AUTO-SHUTDOWN MODE

The PWM mode supports an Auto-Shutdown mode that will disable the PWM outputs when an external shutdown event occurs. Auto-Shutdown mode places the PWM output pins into a predetermined state. This mode is used to help prevent the PWM from damaging the application.

The auto-shutdown sources are selected using the CCPxAS<2:0> bits of the CCPxAS register. A shutdown event may be generated by:

- A logic '0' on the FLT0 pin
- A logic '1' on a Comparator (async_CxOUT) output

A shutdown condition is indicated by the CCPxASE (Auto-Shutdown Event Status) bit of the CCPxAS register. If the bit is a '0', the PWM pins are operating normally. If the bit is a '1', the PWM outputs are in the shutdown state.

When a shutdown event occurs, two things happen:

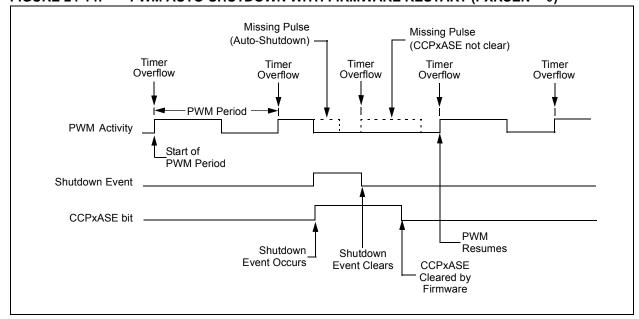
The CCPxASE bit is set to '1'. The CCPxASE will remain set until cleared in firmware or an auto-restart occurs (see Section 24.4.4 "Auto-restart Mode").

The enabled PWM pins are asynchronously placed in their shutdown states. The PWM output pins are grouped into pairs [PxA/PxC] and [PxB/PxD]. The state of each pin pair is determined by the PSSxAC and PSSxBD bits of the CCPxAS register. Each pin pair may be placed into one of three states:

- · Drive logic '1'
- · Drive logic '0'
- Tri-state (high-impedance)

- Note 1: The auto-shutdown condition is a levelbased signal, not an edge-based signal. As long as the level is present, the autoshutdown will persist.
 - **2:** Writing to the CCPxASE bit of the CCPxAS register is disabled while an auto-shutdown condition persists.
 - **3:** Once the auto-shutdown condition has been removed and the PWM restarted (either through firmware or auto-restart) the PWM signal will always restart at the beginning of the next PWM period.
 - 4: Prior to an auto-shutdown event caused by a comparator output or FLT0 pin event, a software shutdown can be triggered in firmware by setting the CCPxASE bit of the CCPxAS register to '1'. The autorestart feature tracks the active status of a shutdown caused by a comparator output or FLT0 pin event only. If it is enabled at this time, it will immediately clear this bit and restart the ECCP module at the beginning of the next PWM period.

FIGURE 24-14: PWM AUTO-SHUTDOWN WITH FIRMWARE RESTART (PXRSEN = 0)

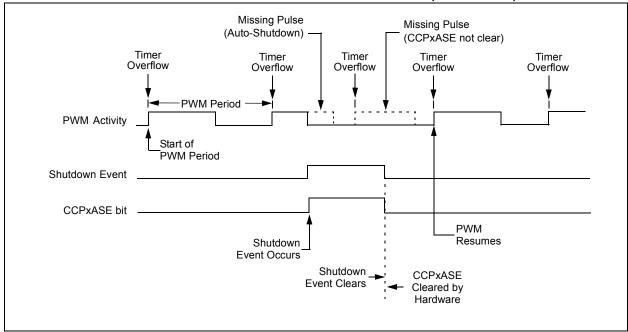


24.4.4 AUTO-RESTART MODE

The Enhanced PWM can be configured to automatically restart the PWM signal once the auto-shutdown condition has been removed. Auto-restart is enabled by setting the PxRSEN bit in the PWMxCON register.

If auto-restart is enabled, the CCPxASE bit will remain set as long as the auto-shutdown condition is active. When the auto-shutdown condition is removed, the CCPxASE bit will be cleared via hardware and normal operation will resume.

FIGURE 24-15: PWM AUTO-SHUTDOWN WITH AUTO-RESTART (PXRSEN = 1)

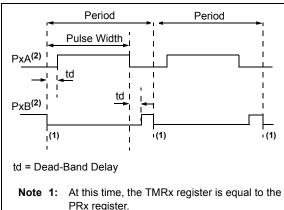


24.4.5 PROGRAMMABLE DEAD-BAND DELAY MODE

In Half-Bridge applications where all power switches are modulated at the PWM frequency, the power switches normally require more time to turn off than to turn on. If both the upper and lower power switches are switched at the same time (one turned on, and the other turned off), both switches may be on for a short period of time until one switch completely turns off. During this brief interval, a very high current (*shootthrough current*) will flow through both power switches, shorting the bridge supply. To avoid this potentially destructive shoot-through current from flowing during switching, turning on either of the power switches is normally delayed to allow the other switch to completely turn off.

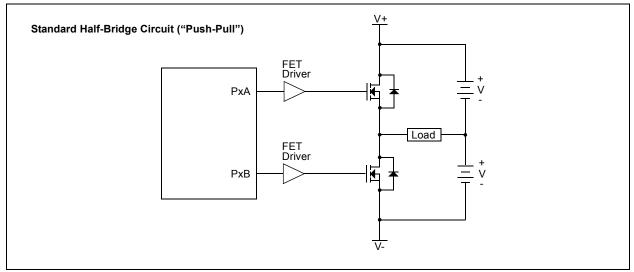
In Half-Bridge mode, a digitally programmable deadband delay is available to avoid shoot-through current from destroying the bridge power switches. The delay occurs at the signal transition from the non-active state to the active state. See Figure 24-16 for illustration. The lower seven bits of the associated PWMxCON register (Register 24-4) sets the delay period in terms of microcontroller instruction cycles (Tcy or 4 Tosc).

FIGURE 24-16: EXAMPLE OF HALF-BRIDGE PWM OUTPUT



2: Output signals are shown as active-high.

FIGURE 24-17: EXAMPLE OF HALF-BRIDGE APPLICATIONS



24.4.6 PWM STEERING MODE

In Single Output mode, PWM steering allows any of the PWM pins to be the modulated signal. Additionally, the same PWM signal can be simultaneously available on multiple pins.

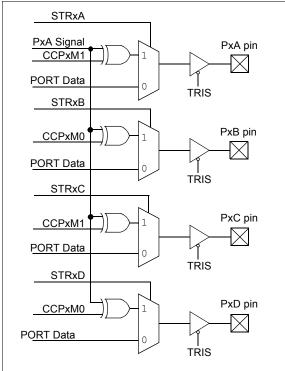
Once the Single Output mode is selected (CCPxM<3:2> = 11 and PxM<1:0> = 00 of the CCPxCON register), the user firmware can bring out the same PWM signal to one, two, three or four output pins by setting the appropriate STRx<D:A> bits of the PSTRxCON register, as shown in Table 24-8.

Note: The associated TRIS bits must be set to output ('0') to enable the pin output driver in order to see the PWM signal on the pin.

While the PWM Steering mode is active, CCPxM<1:0> bits of the CCPxCON register select the PWM output polarity for the Px<D:A> pins.

The PWM auto-shutdown operation also applies to PWM Steering mode as described in **Section 24.4.3** "Enhanced PWM Auto-shutdown mode". An auto-shutdown event will only affect pins that have PWM outputs enabled.

FIGURE 24-18: SIMPLIFIED STEERING BLOCK DIAGRAM



Note 1: Port outputs are configured as shown when the CCPxCON register bits PxM<1:0> = 00 and CCPxM<3:2> = 11.

2: Single PWM output requires setting at least one of the STRx bits.

24.4.6.1 Steering Synchronization

The STRxSYNC bit of the PSTRxCON register gives the user two selections of when the steering event will happen. When the STRxSYNC bit is '0', the steering event will happen at the end of the instruction that writes to the PSTRxCON register. In this case, the output signal at the Px<D:A> pins may be an incomplete PWM waveform. This operation is useful when the user firmware needs to immediately remove a PWM signal from the pin.

When the STRxSYNC bit is '1', the effective steering update will happen at the beginning of the next PWM period. In this case, steering on/off the PWM output will always produce a complete PWM waveform.

Figures 24-19 and 24-20 illustrate the timing diagrams of the PWM steering depending on the STRxSYNC setting.

24.4.7 START-UP CONSIDERATIONS

When any PWM mode is used, the application hardware must use the proper external pull-up and/or pull-down resistors on the PWM output pins.

The CCPxM<1:0> bits of the CCPxCON register allow the user to choose whether the PWM output signals are active-high or active-low for each pair of PWM output pins (PxA/PxC and PxB/PxD). The PWM output polarities must be selected before the PWM pin output drivers are enabled. Changing the polarity

configuration while the PWM pin output drivers are enable is not recommended since it may result in damage to the application circuits.

The PxA, PxB, PxC and PxD output latches may not be in the proper states when the PWM module is initialized. Enabling the PWM pin output drivers at the same time as the Enhanced PWM modes may cause damage to the application circuit. The Enhanced PWM modes must be enabled in the proper Output mode and complete a full PWM cycle before enabling the PWM pin output drivers. The completion of a full PWM cycle is indicated by the TMRxIF bit of the PIRx register being set as the second PWM period begins.

When the microcontroller is released from Reset, all of the I/O pins are in the high-impedance state. The external circuits must keep the power switch devices in the Off state until the microcontroller drives the I/O pins with the proper signal levels or activates the PWM output(s).

24.4.8 ALTERNATE PIN LOCATIONS

This module incorporates I/O pins that can be moved to other locations with the use of the alternate pin function registers, APFCON0 or APFCON1. To determine which pins can be moved and what their default locations are upon a Reset, see Section 12.1 "Alternate Pin Function" for more information.

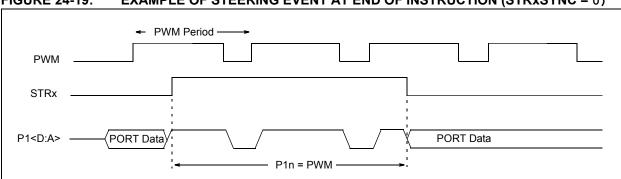


FIGURE 24-19: EXAMPLE OF STEERING EVENT AT END OF INSTRUCTION (STRXSYNC = 0)

Note:

FIGURE 24-20: EXAMPLE OF STEERING EVENT AT BEGINNING OF INSTRUCTION (STRxSYNC = 1)

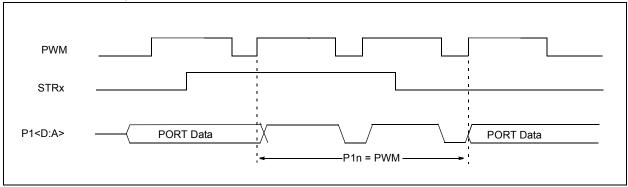


TABLE 24-10: SUMMARY OF REGISTERS ASSOCIATED WITH ENHANCED PWM

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
APFCON1	_	_	SDO2SEL ⁽²⁾	SS2SEL ⁽²⁾	P1DSEL	P1CSEL	P2BSEL	CCP2SEL	122
CCP1CON	P1M<	1:0> ⁽¹⁾	DC1B	3<1:0>		CCP1	VI<3:0>		236
CCP2CON	P2M<	1:0> ⁽¹⁾	DC2B	3<1:0>	CCP2M<3:0>				236
CCP3CON	_	_	DC3B	3<1:0>			236		
CCP4CON	_	_	DC4B	3<1:0>		CCP4	VI<3:0>		236
CCP1AS	CCP1ASE		CCP1AS<2:0>		PSS1A	.C<1:0>	PSS1B	D<1:0>	238
CCP2AS	CCP2ASE		CCP2AS<2:0>			.C<1:0>	PSS2B	D<1:0>	238
CCPTMRS	C4TSE	L<1:0>	C3TSE	L<1:0>	C2TSE	L<1:0>	C1TSE	L<1:0>	237
INLVLA	_	_	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	127
INLVLC	INLVLC7 ⁽¹⁾	INLVLC6 ⁽¹⁾	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	138
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	90
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	91
PIE2	OSFIE	C2IE	C1IE	EEIE	BCL1IE	_	_	CCP2IE	92
PIE3	_	_	CCP4IE	CCP3IE	TMR6IE	_	TMR4IE	_	93
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	95
PIR2	OSFIF	C2IF	C1IF	EEIF	BCL1IF	_	_	CCP2IF	96
PIR3	_	_	CCP4IF	CCP3IF	TMR6IF	_	TMR4IF	_	97
PRx	Timer2/4/6 Per	riod Register							199*
PSTR1CON	_	_	_	STR1SYNC	STR1D	STR1C	STR1B	STR1A	240
PSTR2CON	_	_	_	STR2SYNC	STR2D	STR2C	STR2B	STR2A	240
PWM1CON	P1RSEN		•		P1DC<6:0>	•	•	•	239
PWM2CON	P2RSEN				P2DC<6:0>				239
T2CON	_		T2OUTI	PS<3:0>		TMR2ON	T2CKP	'S<1:0>	201
T4CON	_		T4OUTI	PS<3:0>		TMR4ON	T4CKP	'S<1:0>	201
T6CON			T6OUTI	PS<3:0>		TMR6ON	T6CKP	'S<1:0>	201
TMRx	Timer2/4/6 Mo	dule Register							199*
TRISA	_	_	TRISA5	TRISA5 TRISA4 TRISA3 TRISA2 TRISA1 TRISA0					125
TRISC	TRISC7 ⁽²⁾	TRISC6 ⁽²⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	136

Legend: — Unimplemented location, read as '0'. Shaded cells are not used by the PWM.

1: Applies to ECCP modules only. 2: PIC16(L)F1829 only. Note

^{*} Page provides register information.

24.5 CCP Control Registers

REGISTER 24-1: CCPxCON: CCPx CONTROL REGISTER

R/W-00	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	
PxM<1:0> ⁽¹⁾		DCxB<1:0>		CCPxM<3:0>				
bit 7							bit 0	

 Legend:

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

 u = Bit is unchanged
 x = Bit is unknown
 -n/n = Value at POR and BOR/Value at all other Reset

 '1' = Bit is set
 '0' = Bit is cleared

bit 7-6 PxM<1:0>: Enhanced PWM Output Configuration bits⁽¹⁾

Capture mode:

Unused

Compare mode:

Unused

If CCPxM<3:2> = 00, 01, 10:

xx = PxA assigned as Capture/Compare input; PxB, PxC, PxD assigned as port pins

If CCPxM<3:2> = 11:

00 = Single output; PxA modulated; PxB, PxC, PxD assigned as port pins

01 = Full-Bridge output forward; PxD modulated; PxA active; PxB, PxC inactive

10 = Half-Bridge output; PxA, PxB modulated with dead-band control; PxC, PxD assigned as port pins

11 = Full-Bridge output reverse; PxB modulated; PxC active; PxA, PxD inactive

bit 5-4 DCxB<1:0>: PWM Duty Cycle Least Significant bits

Capture mode:

Unused

Compare mode:

Unused

PWM mode:

These bits are the two LSbs of the PWM duty cycle. The eight MSbs are found in CCPRxL.

bit 3-0 CCPxM<3:0>: ECCPx Mode Select bits

0000 = Capture/Compare/PWM off (resets ECCPx module)

0001 = Reserved

0010 = Compare mode: toggle output on match

0011 = Reserved

0100 = Capture mode: every falling edge

0101 = Capture mode: every rising edge

0110 = Capture mode: every 4th rising edge

0111 = Capture mode: every 16th rising edge

1000 = Compare mode: initialize ECCPx pin low; set output on compare match (set CCPxIF)

1001 = Compare mode: initialize ECCPx pin high; clear output on compare match (set CCPxIF)

1010 = Compare mode: generate software interrupt only; ECCPx pin reverts to I/O state

1011 = Compare mode: Special Event Trigger (ECCPx resets Timer, sets CCPxIF bit, starts A/D conversion if A/D module is enabled)⁽¹⁾

CCP Modules only:

11xx = PWM mode

ECCP Modules only:

1100 = PWM mode: PxA, PxC active-high; PxB, PxD active-high

1101 = PWM mode: PxA, PxC active-high; PxB, PxD active-low

1110 = PWM mode: PxA, PxC active-low; PxB, PxD active-high

1111 = PWM mode: PxA, PxC active-low; PxB, PxD active-low

Note 1: These bits are not implemented on CCP<5:4>.

REGISTER 24-2: CCPTMRS: PWM TIMER SELECTION CONTROL REGISTER 0

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
C4TSEL<1:0>		C3TSEL<1:0>		C2TSEL<1:0>		C1TSEL<1:0>	
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7-6	C4TSEL<1:0>: CCP4 Timer Selection bits 00 = CCP4 is based off Timer2 in PWM mode 01 = CCP4 is based off Timer4 in PWM mode 10 = CCP4 is based off Timer6 in PWM mode 11 = Reserved
bit 5-4	C3TSEL<1:0>: CCP3 Timer Selection bits
	00 = CCP3 is based off Timer2 in PWM mode 01 = CCP3 is based off Timer4 in PWM mode
	10 = CCP3 is based off Timer6 in PWM mode
	11 = Reserved
bit 3-2	C2TSEL<1:0>: CCP2 Timer Selection bits
	00 = CCP2 is based off Timer2 in PWM mode
	01 = CCP2 is based off Timer4 in PWM mode
	10 = CCP2 is based off Timer6 in PWM mode
	11 = Reserved
bit 1-0	C1TSEL<1:0>: CCP1 Timer Selection bits
	00 = CCP1 is based off Timer2 in PWM mode
	01 = CCP1 is based off Timer4 in PWM mode
	10 = CCP1 is based off Timer6 in PWM mode
	11 = Reserved

REGISTER 24-3: CCPxAS: CCPx AUTO-SHUTDOWN CONTROL REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
CCPxASE	CCPxAS<2:0>			PSSxA	C<1:0>	PSSxBD<1:0>	
bit 7							bit 0

Legend:R = Readable bitW = Writable bitU = Unimplemented bit, read as '0'u = Bit is unchangedx = Bit is unknown-n/n = Value at POR and BOR/Value at all other Resets'1' = Bit is set'0' = Bit is cleared

bit 7

CCPxASE: CCPx Auto-Shutdown Event Status bit

1 = A shutdown event has occurred; CCPx outputs are in shutdown state

0 = CCPx outputs are operating

bit 6-4

CCxPAS<2:0>: CCPx Auto-Shutdown Source Select bits

000 = Auto-shutdown is disabled

001 = Comparator C1 output high(1)

010 = Comparator C2 output high(1)

011 = Either Comparator C1 or C2 high(1)

100 = VIL on FLT0 pin 101 = VIL on FLT0 pin or Comparator C1 high⁽¹⁾ 110 = VIL on FLT0 pin or Comparator C2 high⁽¹⁾

111 = VIL on FLT0 pin or Comparator C1 or Comparator C2 high(1)

bit 3-2 **PSSxAC<1:0>:** Pins PxA and PxC Shutdown State Control bits

00 = Drive pins PxA and PxC to '0' 01 = Drive pins PxA and PxC to '1' 1x = Pins PxA and PxC tri-state

bit 1-0 PSSxBD<1:0>: Pins PxB and PxD Shutdown State Control bits

00 = Drive pins PxB and PxD to '0' 01 = Drive pins PxB and PxD to '1' 1x = Pins PxB and PxD tri-state

Note 1: If CxSYNC is enabled, the shutdown will be delayed by Timer1.

REGISTER 24-4: PWMxCON: ENHANCED PWM CONTROL REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
PxRSEN				PxDC<6:0>			
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7 PxRSEN: PWM Restart Enable bit

1 = Upon auto-shutdown, the CCPxASE bit clears automatically once the shutdown event goes away; the PWM restarts automatically

0 = Upon auto-shutdown, CCPxASE must be cleared in software to restart the PWM

bit 6-0 PxDC<6:0>: PWM Delay Count bits

PxDCx = Number of Fosc/4 (4 * Tosc) cycles between the scheduled time when a PWM signal **should** transition active and the **actual** time it transitions active

Note 1: Bit resets to '0' with Two-Speed Start-up and LP, XT or HS selected as the Oscillator mode or Fail-Safe mode is enabled.

REGISTER 24-5: PSTRxCON: PWM STEERING CONTROL REGISTER⁽¹⁾

U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-1/1
_	1		STRxSYNC	STRxD	STRxC	STRxB	STRxA
bit 7							bit 0

Legend:R = Readable bitW = Writable bitU = Unimplemented bit, read as '0'u = Bit is unchangedx = Bit is unknown-n/n = Value at POR and BOR/Value at all other Resets'1' = Bit is set'0' = Bit is cleared

bit 7-5 Unimplemented: Read as '0' bit 4 STRxSYNC: Steering Sync bit 1 = Output steering update occurs on next PWM period 0 = Output steering update occurs at the beginning of the instruction cycle boundary bit 3 STRxD: Steering Enable bit D 1 = PxD pin has the PWM waveform with polarity control from CCPxM<1:0> 0 = PxD pin is assigned to port pin bit 2 STRxC: Steering Enable bit C 1 = PxC pin has the PWM waveform with polarity control from CCPxM<1:0> 0 = PxC pin is assigned to port pin bit 1 STRxB: Steering Enable bit B 1 = PxB pin has the PWM waveform with polarity control from CCPxM<1:0> 0 = PxB pin is assigned to port pin bit 0 STRxA: Steering Enable bit A 1 = PxA pin has the PWM waveform with polarity control from CCPxM<1:0> 0 = PxA pin is assigned to port pin

Note 1: The PWM Steering mode is available only when the CCPxCON register bits CCPxM<3:2> = 11 and PxM<1:0> = 00.

25.0 MASTER SYNCHRONOUS SERIAL PORT (MSSP1 AND MSSP2) MODULE

25.1 Master SSPx (MSSPx) Module Overview

The Master Synchronous Serial Port (MSSPx) module is a serial interface useful for communicating with other peripheral or microcontroller devices. These peripheral devices may be serial EEPROMs, shift registers, display drivers, A/D converters, etc. The MSSPx module can operate in one of two modes:

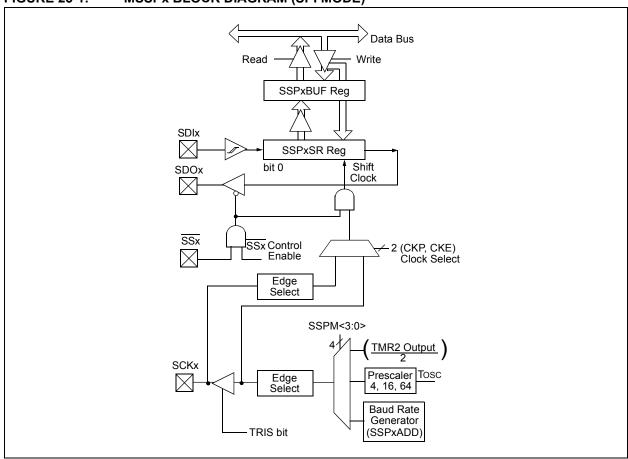
- · Serial Peripheral Interface (SPI)
- Inter-Integrated Circuit (I²C™)

The SPI interface supports the following modes and features:

- · Master mode
- · Slave mode
- · Clock Parity
- Slave Select Synchronization (Slave mode only)
- · Daisy-chain connection of slave devices

Figure 25-1 is a block diagram of the SPI interface module.

FIGURE 25-1: MSSPx BLOCK DIAGRAM (SPI MODE)



The I²C interface supports the following modes and features:

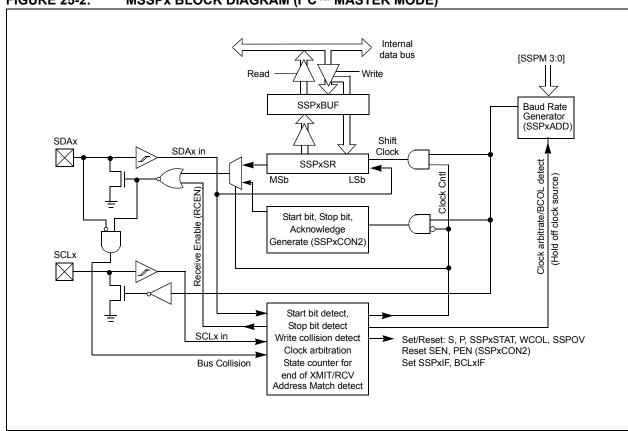
- · Master mode
- · Slave mode
- · Byte NACKing (Slave mode)
- · Limited Multi-master support
- · 7-bit and 10-bit addressing
- · Start and Stop interrupts
- · Interrupt masking
- · Clock stretching
- · Bus collision detection
- · General call address matching
- · Address masking
- · Address Hold and Data Hold modes
- · Selectable SDAx hold times

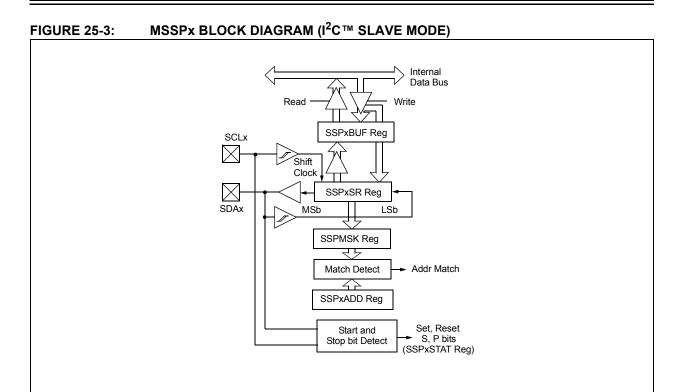
Figure 25-2 is a block diagram of the I²C interface module in Master mode. Figure 25-3 is a diagram of the I²C interface module in Slave mode.

The PIC16F1829 has two MSSP modules, MSSP1 and MSSP2, each module operating independently from the other.

- Note 1: In devices with more than one MSSP module, it is very important to pay close attention to SSPxCONx register names. SSP1CON1 and SSP1CON2 registers control different operational aspects of the same module, while SSP1CON1 and SSP2CON1 control the same features for two different modules.
 - 2: Throughout this section, generic references to an MSSP module in any of its operating modes may be interpreted as being equally applicable to MSSP1 or MSSP2. register names, module I/O signals, and bit names may use the generic designator 'x' to indicate the use of a numeral to distinguish a particular module when required.







25.2 SPI Mode Overview

The Serial Peripheral Interface (SPI) bus is a synchronous serial data communication bus that operates in Full Duplex mode. Devices communicate in a master/slave environment where the master device initiates the communication. A slave device is controlled through a chip select known as Slave Select.

The SPI bus specifies four signal connections:

- Serial Clock (SCKx)
- Serial Data Out (SDOx)
- Serial Data In (SDIx)
- Slave Select (SSx)

Figure 25-1 shows the block diagram of the MSSPx module when operating in SPI Mode.

The SPI bus operates with a single master device and one or more slave devices. When multiple slave devices are used, an independent Slave Select connection is required from the master device to each slave device.

Figure 25-4 shows a typical connection between a master device and multiple slave devices.

The master selects only one slave at a time. Most slave devices have tri-state outputs so their output signal appears disconnected from the bus when they are not selected.

Transmissions involve two shift registers, eight bits in size, one in the master and one in the slave. With either the master or the slave device, data is always shifted out one bit at a time, with the Most Significant bit (MSb) shifted out first. At the same time, a new Least Significant bit (LSb) is shifted into the same register.

Figure 25-5 shows a typical connection between two processors configured as master and slave devices.

Data is shifted out of both shift registers on the programmed clock edge and latched on the opposite edge of the clock.

The master device transmits information out on its SDOx output pin which is connected to, and received by, the slave's SDIx input pin. The slave device transmits information out on its SDOx output pin, which is connected to, and received by, the master's SDIx input pin.

To begin communication, the master device first sends out the clock signal. Both the master and the slave devices should be configured for the same clock polarity.

The master device starts a transmission by sending out the MSb from its shift register. The slave device reads this bit from that same line and saves it into the LSb position of its shift register.

During each SPI clock cycle, a full duplex data transmission occurs. This means that while the master device is sending out the MSb from its shift register (on

its SDOx pin) and the slave device is reading this bit and saving it as the LSb of its shift register, that the slave device is also sending out the MSb from its shift register (on its SDOx pin) and the master device is reading this bit and saving it as the LSb of its shift register.

After eight bits have been shifted out, the master and slave have exchanged register values.

If there is more data to exchange, the shift registers are loaded with new data and the process repeats itself.

Whether the data is meaningful or not (dummy data), depends on the application software. This leads to three scenarios for data transmission:

- Master sends useful data and slave sends dummy data
- Master sends useful data and slave sends useful data
- Master sends dummy data and slave sends useful data

Transmissions may involve any number of clock cycles. When there is no more data to be transmitted, the master stops sending the clock signal and it deselects the slave.

Every slave device connected to the bus that has not been selected through its slave select line must disregard the clock and transmission signals and must not transmit out any data of its own.

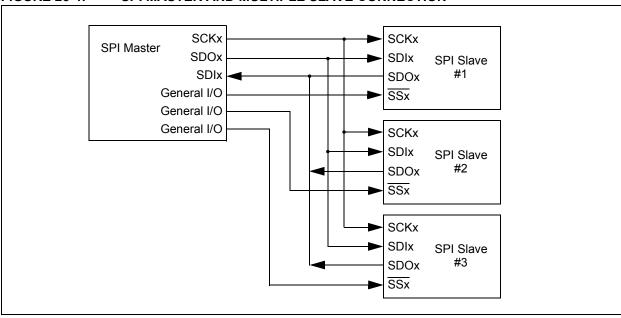


FIGURE 25-4: SPI MASTER AND MULTIPLE SLAVE CONNECTION

25.2.1 SPI MODE REGISTERS

The MSSPx module has five registers for SPI mode operation. These are:

- MSSPx STATUS register (SSPxSTAT)
- MSSPx Control Register 1 (SSPxCON1)
- MSSPx Control Register 3 (SSPxCON3)
- MSSPx Data Buffer register (SSPxBUF)
- MSSPx Address register (SSPxADD)
- MSSPx Shift register (SSPxSR) (Not directly accessible)

SSPxCON1 and SSPxSTAT are the control and STATUS registers in SPI mode operation. The SSPxCON1 register is readable and writable. The lower 6 bits of the SSPxSTAT are read-only. The upper two bits of the SSPxSTAT are read/write.

In one SPI master mode, SSPxADD can be loaded with a value used in the Baud Rate Generator. More information on the Baud Rate Generator is available in **Section 25.7 "Baud Rate Generator"**.

SSPxSR is the shift register used for shifting data in and out. SSPxBUF provides indirect access to the SSPxSR register. SSPxBUF is the buffer register to which data bytes are written, and from which data bytes are read.

In receive operations, SSPxSR and SSPxBUF together create a buffered receiver. When SSPxSR receives a complete byte, it is transferred to SSPxBUF and the SSPxIF interrupt is set.

During transmission, the SSPxBUF is not buffered. A write to SSPxBUF will write to both SSPxBUF and SSPxSR.

25.2.2 SPI MODE OPERATION

When initializing the SPI, several options need to be specified. This is done by programming the appropriate control bits (SSPxCON1<5:0> and SSPxSTAT<7:6>). These control bits allow the following to be specified:

- Master mode (SCKx is the clock output)
- Slave mode (SCKx is the clock input)
- Clock Polarity (Idle state of SCKx)
- Data Input Sample Phase (middle or end of data output time)
- Clock Edge (output data on rising/falling edge of SCKx)
- Clock Rate (Master mode only)
- Slave Select mode (Slave mode only)

To enable the serial port, SSPx Enable bit, SSPEN of the SSPxCON1 register, must be set. To reset or reconfigure SPI mode, clear the SSPEN bit, re-initialize the SSPxCONx registers and then set the SSPEN bit. This configures the SDIx, SDOx, SCKx and SSx pins as serial port pins. For the pins to behave as the serial port function, some must have their data direction bits (in the TRIS register) appropriately programmed as follows:

- · SDIx must have corresponding TRIS bit set
- SDOx must have corresponding TRIS bit cleared
- SCKx (Master mode) must have corresponding TRIS bit cleared
- SCKx (Slave mode) must have corresponding TRIS bit set
- SSx must have corresponding TRIS bit set

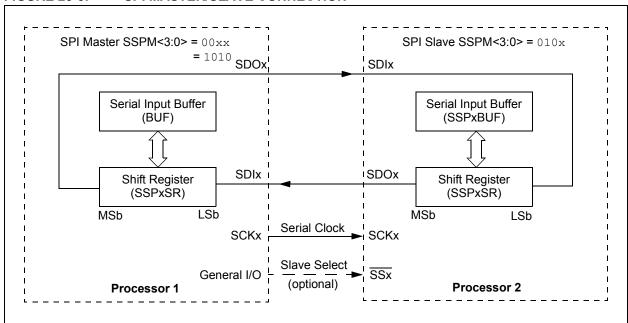
Any serial port function that is not desired may be overridden by programming the corresponding data direction (TRIS) register to the opposite value.

The MSSPx consists of a transmit/receive shift register (SSPxSR) and a buffer register (SSPxBUF). The SSPxSR shifts the data in and out of the device, MSb first. The SSPxBUF holds the data that was written to the SSPxSR until the received data is ready. Once the eight bits of data have been received, that byte is moved to the SSPxBUF register. Then, the Buffer Full Detect bit, BF of the SSPxSTAT register, and the interrupt flag bit, SSPxIF, are set. This double-buffering of the received data (SSPxBUF) allows the next byte to start reception before reading the data that was just received. Any write to the SSPxBUF register during transmission/reception of data will be ignored and the write collision detect bit WCOL of the SSPxCON1 register, will be set. User software must clear the WCOL bit to allow the following write(s) to the SSPxBUF register to complete successfully.

When the application software is expecting to receive valid data, the SSPxBUF should be read before the next byte of data to transfer is written to the SSPxBUF. The Buffer Full bit, BF of the SSPxSTAT register, indicates when SSPxBUF has been loaded with the received data (transmission is complete). When the SSPxBUF is read, the BF bit is cleared. This data may be irrelevant if the SPI is only a transmitter. Generally, the MSSPx interrupt is used to determine when the transmission/reception has completed. If the interrupt method is not going to be used, then software polling can be done to ensure that a write collision does not occur.

The SSPxSR is not directly readable or writable and can only be accessed by addressing the SSPxBUF register. Additionally, the SSPxSTAT register indicates the various Status conditions.

FIGURE 25-5: SPI MASTER/SLAVE CONNECTION



25.2.3 SPI MASTER MODE

The master can initiate the data transfer at any time because it controls the SCKx line. The master determines when the slave (Processor 2, Figure 25-5) is to broadcast data by the software protocol.

In Master mode, the data is transmitted/received as soon as the SSPxBUF register is written to. If the SPI is only going to receive, the SDOx output could be disabled (programmed as an input). The SSPxSR register will continue to shift in the signal present on the SDIx pin at the programmed clock rate. As each byte is received, it will be loaded into the SSPxBUF register as if a normal received byte (interrupts and Status bits appropriately set).

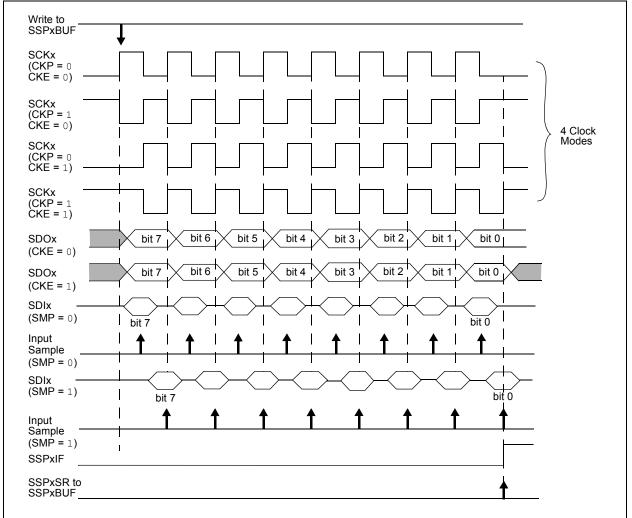
The clock polarity is selected by appropriately programming the CKP bit of the SSPxCON1 register and the CKE bit of the SSPxSTAT register. This then, would give waveforms for SPI communication as shown in Figure 25-6, Figure 25-8, Figure 25-9 and Figure 25-10, where the MSB is transmitted first. In Master mode, the SPI clock rate (bit rate) is user programmable to be one of the following:

- Fosc/4 (or Tcy)
- Fosc/16 (or 4 * Tcy)
- Fosc/64 (or 16 * Tcy)
- · Timer2 output/2
- Fosc/(4 * (SSPxADD + 1))

Figure 25-6 shows the waveforms for Master mode.

When the CKE bit is set, the SDOx data is valid before there is a clock edge on SCKx. The change of the input sample is shown based on the state of the SMP bit. The time when the SSPxBUF is loaded with the received data is shown.

FIGURE 25-6: SPI MODE WAVEFORM (MASTER MODE)



25.2.4 SPI SLAVE MODE

In Slave mode, the data is transmitted and received as external clock pulses appear on SCKx. When the last bit is latched, the SSPxIF interrupt flag bit is set.

Before enabling the module in SPI Slave mode, the clock line must match the proper Idle state. The clock line can be observed by reading the SCKx pin. The Idle state is determined by the CKP bit of the SSPxCON1 register.

While in Slave mode, the external clock is supplied by the external clock source on the SCKx pin. This external clock must meet the minimum high and low times as specified in the electrical specifications.

While in Sleep mode, the slave can transmit/receive data. The shift register is clocked from the SCKx pin input and when a byte is received, the device will generate an interrupt. If enabled, the device will wake-up from Sleep.

25.2.4.1 Daisy-Chain Configuration

The SPI bus can sometimes be connected in a daisy-chain configuration. The first slave output is connected to the second slave input, the second slave output is connected to the third slave input, and so on. The final slave output is connected to the master input. Each slave sends out, during a second group of clock pulses, an exact copy of what was received during the first group of clock pulses. The whole chain acts as one large communication shift register. The daisy-chain feature only requires a single Slave Select line from the master device.

Figure 25-7 shows the block diagram of a typical daisy-chain connection when operating in SPI mode.

In a daisy-chain configuration, only the most recent byte on the bus is required by the slave. Setting the BOEN bit of the SSPxCON3 register will enable writes to the SSPxBUF register, even if the previous byte has not been read. This allows the software to ignore data that may not apply to it.

25.2.5 SLAVE SELECT SYNCHRONIZATION

The Slave Select can also be used to synchronize communication. The Slave Select line is held high until the master device is ready to communicate. When the Slave Select line is pulled low, the slave knows that a new transmission is starting.

If the slave fails to receive the communication properly, it will be reset at the end of the transmission, when the Slave Select line returns to a high state. The slave is then ready to receive a new transmission when the Slave Select line is pulled low again. If the Slave Select line is not used, there is a risk that the slave will eventually become out of sync with the master. If the slave misses a bit, it will always be one bit off in future transmissions. Use of the Slave Select line allows the slave and master to align themselves at the beginning of each transmission.

The \overline{SSx} pin allows a Synchronous Slave mode. The SPI must be in Slave mode with \overline{SSx} pin control enabled (SSPxCON1<3:0> = 0100).

When the \overline{SSx} pin is low, transmission and reception are enabled and the SDOx pin is driven.

When the SSx pin goes high, the SDOx pin is no longer driven, even if in the middle of a transmitted byte and becomes a floating output. External pull-up/pull-down resistors may be desirable depending on the application.

- Note 1: When the SPI is in Slave mode with \$\overline{SSx}\$ pin control enabled (\$SPxCON1<3:0> = 0100), the SPI module will reset if the \$\overline{SSx}\$ pin is set to VDD.
 - 2: When the SPI is used in Slave mode with CKE set; the user must enable SSx pin control.
 - **3:** While operated in SPI Slave mode the SMP bit of the SSPxSTAT register must remain clear.

When the SPI module resets, the bit counter is forced to '0'. This can be done by either forcing the \overline{SSx} pin to a high level or clearing the SSPEN bit.

FIGURE 25-7: SPI DAISY-CHAIN CONNECTION

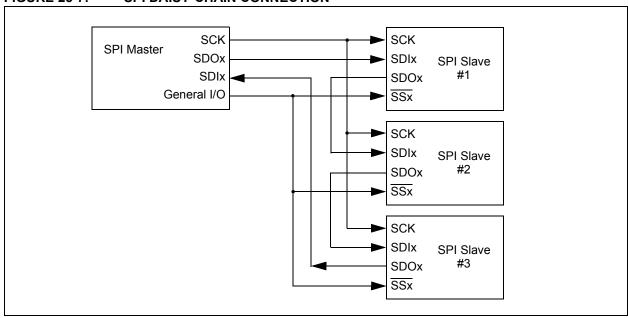
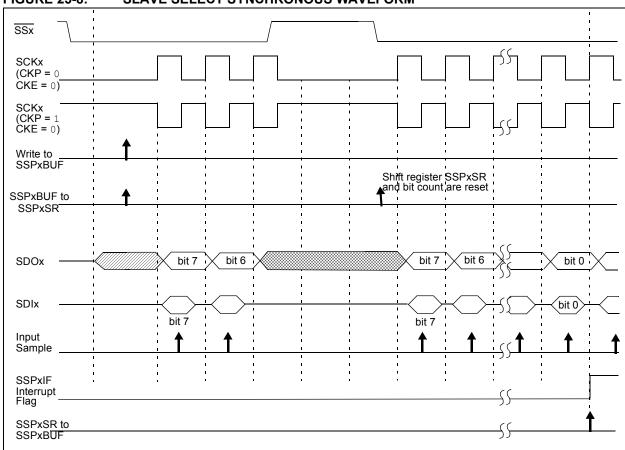


FIGURE 25-8: SLAVE SELECT SYNCHRONOUS WAVEFORM





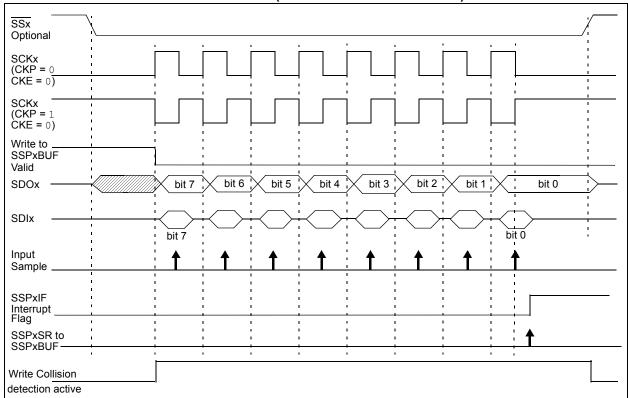
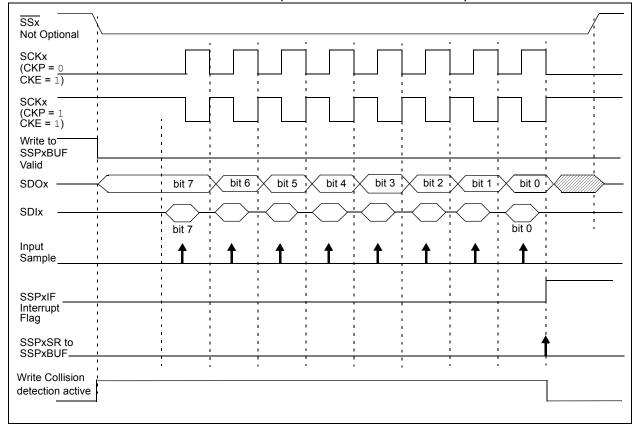


FIGURE 25-10: SPI MODE WAVEFORM (SLAVE MODE WITH CKE = 1)



25.2.6 SPI OPERATION IN SLEEP MODE

In SPI Master mode, module clocks may be operating at a different speed than when in full power mode; in the case of the Sleep mode, all clocks are halted.

Special care must be taken by the user when the MSSPx clock is much faster than the system clock.

In Slave mode, when MSSPx interrupts are enabled, after the master completes sending data, an MSSPx interrupt will wake the controller from Sleep.

If an exit from Sleep mode is not desired, MSSPx interrupts should be disabled.

In SPI Master mode, when the Sleep mode is selected, all module clocks are halted and the transmission/reception will remain in that state until the device wakes. After the device returns to Run mode, the module will resume transmitting and receiving data.

In SPI Slave mode, the SPI Transmit/Receive Shift register operates asynchronously to the device. This allows the device to be placed in Sleep mode and data to be shifted into the SPI Transmit/Receive Shift register. When all eight bits have been received, the MSSPx interrupt flag bit will be set and if enabled, will wake the device.

TABLE 25-1: SUMMARY OF REGISTERS ASSOCIATED WITH SPI OPERATION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ANSELA	_	_	_	ANSA4	_	ANSA2	ANSA1	ANSA0	125
ANSELB ⁽¹⁾	ANSB7	ANSB6	ANSB5	ANSB4	_	_	_	_	132
ANSELC	ANSC7 ⁽¹⁾	ANSC6 ⁽¹⁾	_	_	ANSC3	ANSC2	ANSC1	ANSC0	137
APFCON0	RXDTSEL	SDO1SEL ⁽²⁾	SS1SEL ⁽²⁾	_	T1GSEL	TXCKSEL	_	_	121
APFCON1	_	ı	SDO2SEL ⁽¹⁾	SS2SEL ⁽¹⁾	P1DSEL	P1CSEL	P2BSEL	CCP2SEL	122
INLVLA	_	ı	INLVLA5 ⁽¹⁾	INLVLA4 ⁽¹⁾	INLVLA3	INLVLA2	INLVLA1	INLVLA0	127
INLVLB ⁽¹⁾	INLVLB7	INLVLB6	INLVLB5	INLVLB4	-	-	-	_	132
INLVLC	INLVLC7 ⁽¹⁾	INLVLC6 ⁽¹⁾	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1 ⁽¹⁾	INLVLC0 ⁽¹⁾	138
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	90
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	91
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	95
SSP1BUF	F Synchronous Serial Port Receive Buffer/Transmit Register								
SSP1CON1	WCOL	SSPOV	SSPEN	CKP		289			
SSP1CON3	ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	291
SSP1STAT	SMP	CKE	D/Ā	Р	S	R/W	UA	BF	288
TRISA	_	_	TRISA5 ⁽¹⁾	TRISA4 ⁽¹⁾	TRISA3	TRISA2	TRISA1	TRISA0	125
TRISB ⁽¹⁾	TRISB7	TRISB6	TRISB5	TRISB4	_	_	_	_	131
TRISC	TRISC7 ⁽¹⁾	TRISC6 ⁽¹⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1 ⁽¹⁾	TRISC0 ⁽¹⁾	136

Legend: — Unimplemented location, read as '0'. Shaded cells are not used by the MSSP1 in SPI mode.

Page provides register information.

Note 1: PIC16(L)F1829 only. 2: PIC16(L)F1825 only.

25.3 I²C MODE OVERVIEW

The Inter-Integrated Circuit Bus (I²C) is a multi-master serial data communication bus. Devices communicate in a master/slave environment where the master devices initiate the communication. A Slave device is controlled through addressing.

The I²C bus specifies two signal connections:

- · Serial Clock (SCLx)
- Serial Data (SDAx)

Figure 25-11 shows the block diagram of the MSSPx module when operating in I²C mode.

Both the SCLx and SDAx connections are bidirectional open-drain lines, each requiring pull-up resistors for the supply voltage. Pulling the line to ground is considered a logical zero and letting the line float is considered a logical one.

Figure 25-11 shows a typical connection between two processors configured as master and slave devices.

The I²C bus can operate with one or more master devices and one or more slave devices.

There are four potential modes of operation for a given device:

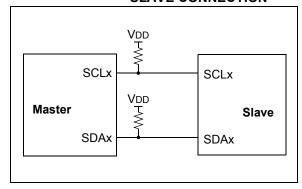
- Master Transmit mode (master is transmitting data to a slave)
- Master Receive mode (master is receiving data from a slave)
- Slave Transmit mode (slave is transmitting data to a master)
- Slave Receive mode (slave is receiving data from the master)

To begin communication, a master device starts out in Master Transmit mode. The master device sends out a Start bit followed by the address byte of the slave it intends to communicate with. This is followed by a single Read/Write bit, which determines whether the master intends to transmit to or receive data from the slave device.

If the requested slave exists on the bus, it will respond with an Acknowledge bit, otherwise known as an ACK. The master then continues in either Transmit mode or Receive mode and the slave continues in the complement, either in Receive mode or Transmit mode, respectively.

A Start bit is indicated by a high-to-low transition of the SDAx line while the SCLx line is held high. Address and data bytes are sent out, Most Significant bit (MSb) first. The Read/Write bit is sent out as a logical one when the master intends to read data from the slave, and is sent out as a logical zero when it intends to write data to the slave.

FIGURE 25-11: I²C MASTER/ SLAVE CONNECTION



The Acknowledge bit (ACK) is an active-low signal, which holds the SDAx line low to indicate to the transmitter that the slave device has received the transmitted data and is ready to receive more.

The transition of a data bit is always performed while the SCLx line is held low. Transitions that occur while the SCLx line is held high are used to indicate Start and Stop bits.

If the master intends to write to the slave, then it repeatedly sends out a byte of data, with the slave responding after each byte with an ACK bit. In this example, the master device is in Master Transmit mode and the slave is in Slave Receive mode.

If the master intends to read from the slave, then it repeatedly receives a byte of data from the slave, and responds after each byte with an \overline{ACK} bit. In this example, the master device is in Master Receive mode and the slave is Slave Transmit mode.

On the last byte of data communicated, the master device may end the transmission by sending a Stop bit. If the master device is in Receive mode, it sends the Stop bit in place of the last \overline{ACK} bit. A Stop bit is indicated by a low-to-high transition of the SDAx line while the SCLx line is held high.

In some cases, the master may want to maintain control of the bus and re-initiate another transmission. If so, the master device may send another Start bit in place of the Stop bit or last ACK bit when it is in receive mode.

The I²C bus specifies three message protocols;

- Single message where a master writes data to a slave.
- Single message where a master reads data from a slave.
- Combined message where a master initiates a minimum of two writes, or two reads, or a combination of writes and reads, to one or more slaves.

When one device is transmitting a logical one, or letting the line float, and a second device is transmitting a logical zero, or holding the line low, the first device can detect that the line is not a logical one. This detection, when used on the SCLx line, is called clock stretching. Clock stretching gives slave devices a mechanism to control the flow of data. When this detection is used on the SDAx line, it is called arbitration. Arbitration ensures that there is only one master device communicating at any single time.

25.3.1 CLOCK STRETCHING

When a slave device has not completed processing data, it can delay the transfer of more data through the process of Clock Stretching. An addressed slave device may hold the SCLx clock line low after receiving or sending a bit, indicating that it is not yet ready to continue. The master that is communicating with the slave will attempt to raise the SCLx line in order to transfer the next bit, but will detect that the clock line has not yet been released. Because the SCLx connection is open-drain, the slave has the ability to hold that line low until it is ready to continue communicating.

Clock stretching allows receivers that cannot keep up with a transmitter to control the flow of incoming data.

25.3.2 ARBITRATION

Each master device must monitor the bus for Start and Stop bits. If the device detects that the bus is busy, it cannot begin a new message until the bus returns to an Idle state.

However, two master devices may try to initiate a transmission on or about the same time. When this occurs, the process of arbitration begins. Each transmitter checks the level of the SDAx data line and compares it to the level that it expects to find. The first transmitter to observe that the two levels don't match, loses arbitration, and must stop transmitting on the SDAx line.

For example, if one transmitter holds the SDAx line to a logical one (lets it float) and a second transmitter holds it to a logical zero (pulls it low), the result is that the SDAx line will be low. The first transmitter then observes that the level of the line is different than expected and concludes that another transmitter is communicating.

The first transmitter to notice this difference is the one that loses arbitration and must stop driving the SDAx line. If this transmitter is also a master device, it also must stop driving the SCLx line. It then can monitor the lines for a Stop condition before trying to reissue its transmission. In the meantime, the other device that has not noticed any difference between the expected and actual levels on the SDAx line continues with its original transmission. It can do so without any complications, because so far, the transmission appears exactly as expected with no other transmitter disturbing the message.

Slave Transmit mode can also be arbitrated, when a master addresses multiple slaves, but this is less common.

If two master devices are sending a message to two different slave devices at the address stage, the master sending the lower slave address always wins arbitration. When two master devices send messages to the same slave address, and addresses can sometimes refer to multiple slaves, the arbitration process must continue into the data stage.

Arbitration usually occurs very rarely, but it is a necessary process for proper multi-master support.

25.4 I²C MODE OPERATION

All MSSPx I²C communication is byte oriented and shifted out MSb first. Six SFR registers and two interrupt flags interface the module with the PIC[®] microcontroller and user software. Two pins, SDAx and SCLx, are exercised by the module to communicate with other external I²C devices.

25.4.1 BYTE FORMAT

All communication in I^2C is done in 9-bit segments. A byte is sent from a master to a slave or vice-versa, followed by an Acknowledge bit sent back. After the 8th falling edge of the SCLx line, the device outputting data on the SDAx changes that pin to an input and reads in an acknowledge value on the next clock pulse.

The clock signal, SCLx, is provided by the master. Data is valid to change while the SCLx signal is low, and sampled on the rising edge of the clock. Changes on the SDAx line while the SCLx line is high define special conditions on the bus, explained below.

25.4.2 DEFINITION OF I²C TERMINOLOGY

There is language and terminology in the description of I^2C communication that have definitions specific to I^2C . That word usage is defined below and may be used in the rest of this document without explanation. This table was adapted from the Philips I^2C specification.

25.4.3 SDAx AND SCLx PINS

Selection of any I²C mode with the SSPEN bit set, forces the SCLx and SDAx pins to be open-drain. These pins should be set by the user to inputs by setting the appropriate TRIS bits.

Note: Data is tied to output zero when an I²C mode is enabled.

25.4.4 SDAx HOLD TIME

The hold time of the SDAx pin is selected by the SDAHT bit of the SSPxCON3 register. Hold time is the time SDAx is held valid after the falling edge of SCLx. Setting the SDAHT bit selects a longer 300 ns minimum hold time and may help on buses with large capacitance.

TABLE 25-2: I²C BUS TERMS

_	1
TERM	Description
Transmitter	The device which shifts data out onto the bus.
Receiver	The device which shifts data in from the bus.
Master	The device that initiates a transfer, generates clock signals and terminates a transfer.
Slave	The device addressed by the master.
Multi-master	A bus with more than one device that can initiate data transfers.
Arbitration	Procedure to ensure that only one master at a time controls the bus. Winning arbitration ensures that the message is not corrupted.
Synchronization	Procedure to synchronize the clocks of two or more devices on the bus.
Idle	No master is controlling the bus, and both SDAx and SCLx lines are high.
Active	Any time one or more master devices are controlling the bus.
Addressed Slave	Slave device that has received a matching address and is actively being clocked by a master.
Matching Address	Address byte that is clocked into a slave that matches the value stored in SSPxADD.
Write Request	Slave receives a matching address with R/W bit clear, and is ready to clock in data.
Read Request	Master sends an address byte with the R/W bit set, indicating that it wishes to clock data out of the Slave. This data is the next and all following bytes until a Restart or Stop.
Clock Stretching	When a device on the bus hold SCLx low to stall communication.
Bus Collision	Any time the SDAx line is sampled low by the module while it is outputting and expected high state.

25.4.5 START CONDITION

The I²C specification defines a Start condition as a transition of SDAx from a high to a low state while SCLx line is high. A Start condition is always generated by the master and signifies the transition of the bus from an Idle to an Active state. Figure 25-12 shows waveforms for Start and Stop conditions.

A bus collision can occur on a Start condition if the module samples the SDAx line low before asserting it low. This does not conform to the I²C Specification that states no bus collision can occur on a Start.

25.4.6 STOP CONDITION

A Stop condition is a transition of the SDAx line from low-to-high state while the SCLx line is high.

Note: At least one SCLx low time must appear before a Stop is valid, therefore, if the SDAx line goes low then high again while the SCLx line stays high, only the Start condition is detected.

25.4.7 RESTART CONDITION

A Restart is valid any time that a Stop would be valid. A master can issue a Restart if it wishes to hold the bus after terminating the current transfer. A Restart has the same effect on the slave that a Start would, resetting all slave logic and preparing it to clock in an address. The master may want to address the same or another slave. Figure 25-13 shows waveforms for a Restart condition.

In 10-bit Addressing Slave mode a Restart is required for the master to clock data out of the addressed slave. Once a slave has been fully addressed, matching both high and low address bytes, the master can issue a Restart and the high address byte with the R/\overline{W} bit set. The slave logic will then hold the clock and prepare to clock out data.

After a full match with R/\overline{W} clear in 10-bit mode, a prior match flag is set and maintained. Until a Stop condition, a high address with R/\overline{W} clear, or high address match fails.

25.4.8 START/STOP CONDITION INTERRUPT MASKING

The SCIE and PCIE bits of the SSPxCON3 register can enable the generation of an interrupt in Slave modes that do not typically support this function. Slave modes where interrupt on Start and Stop detect are already enabled, these bits will have no effect.

FIGURE 25-12: I²C START AND STOP CONDITIONS

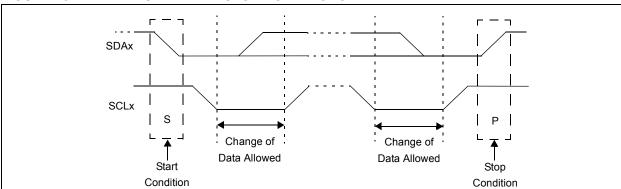
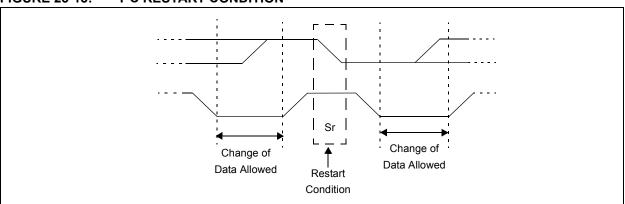


FIGURE 25-13: I²C RESTART CONDITION



25.4.9 ACKNOWLEDGE SEQUENCE

The 9th SCLx pulse for any transferred byte in I^2C is dedicated as an Acknowledge. It allows receiving devices to respond back to the transmitter by pulling the SDAx line low. The transmitter must release control of the line during this time to shift in the response. The Acknowledge (\overline{ACK}) is an active-low signal, pulling the SDAx line low indicated to the transmitter that the device has received the transmitted data and is ready to receive more.

The result of an \overline{ACK} is placed in the ACKSTAT bit of the SSPxCON2 register.

Slave software, when the AHEN and DHEN bits are set, allow the user to set the ACK value sent back to the transmitter. The ACKDT bit of the SSPxCON2 register is set/cleared to determine the response.

Slave hardware will generate an \overline{ACK} response if the AHEN and DHEN bits of the SSPxCON3 register are clear

There are certain conditions where an \overline{ACK} will not be sent by the slave. If the BF bit of the SSPxSTAT register or the SSPOV bit of the SSPxCON1 register are set when a byte is received.

When the module is addressed, after the 8th falling edge of SCLx on the bus, the ACKTIM bit of the SSPxCON3 register is set. The ACKTIM bit indicates the acknowledge time of the active bus. The ACKTIM Status bit is only active when the AHEN bit or DHEN bit is enabled.

25.5 I²C SLAVE MODE OPERATION

The MSSPx Slave mode operates in one of four modes selected in the SSPM bits of SSPxCON1 register. The modes can be divided into 7-bit and 10-bit Addressing mode. 10-bit Addressing modes operate the same as 7-bit with some additional overhead for handling the larger addresses.

Modes with Start and Stop bit interrupts operated the same as the other modes with SSPxIF additionally getting set upon detection of a Start, Restart or Stop condition.

25.5.1 SLAVE MODE ADDRESSES

The SSPxADD register (Register 25-6) contains the Slave mode address. The first byte received after a Start or Restart condition is compared against the value stored in this register. If the byte matches, the value is loaded into the SSPxBUF register and an interrupt is generated. If the value does not match, the module goes Idle and no indication is given to the software that anything happened.

The SSPx Mask register (Register 25-5) affects the address matching process. See **Section 25.5.8** "SSPx Mask Register" for more information.

25.5.1.1 I²C Slave 7-bit Addressing Mode

In 7-bit Addressing mode, the LSb of the received data byte is ignored when determining if there is an address match.

25.5.1.2 I²C Slave 10-bit Addressing Mode

In 10-bit Addressing mode, the first received byte is compared to the binary value of '1 1 1 1 0 A9 A8 0'. A9 and A8 are the two MSb of the 10-bit address and stored in bits 2 and 1 of the SSPxADD register.

After the acknowledge of the high byte the UA bit is set and SCLx is held low until the user updates SSPxADD with the low address. The low address byte is clocked in and all eight bits are compared to the low address value in SSPxADD. Even if there is not an address match; SSPxIF and UA are set, and SCLx is held low until SSPxADD is updated to receive a high byte again. When SSPxADD is updated the UA bit is cleared. This ensures the module is ready to receive the high address byte on the next communication.

A high and low address match as a write request is required at the start of all 10-bit addressing communication. A transmission can be initiated by issuing a Restart once the slave is addressed, and clocking in the high address with the R/\overline{W} bit set. The slave hardware will then acknowledge the read request and prepare to clock out data. This is only valid for a slave after it has received a complete high and low address byte match.

25.5.1.3 Slave Reception

When the R/\overline{W} bit of a matching received address byte is clear, the R/\overline{W} bit of the SSPxSTAT register is cleared. The received address is loaded into the SSPxBUF register and acknowledged.

When the overflow condition exists for a received address, then not Acknowledge is given. An overflow condition is defined as either bit BF of the SSPxSTAT register is set, or bit SSPOV bit of the SSPxCON1 register is set. The BOEN bit of the SSPxCON3 register modifies this operation. For more information see Register 25-4.

An MSSPx interrupt is generated for each transferred data byte. Flag bit, SSPxIF, must be cleared by software.

When the SEN bit of the SSPxCON2 register is set, SCLx will be held low (clock stretch) following each received byte. The clock must be released by setting the CKP bit of the SSPxCON1 register, except sometimes in 10-bit mode. See Section 25.2.3 "SPI Master Mode" for more detail.

25.5.1.4 7-bit Addressing Reception

This section describes a standard sequence of events for the MSSPx module configured as an I²C Slave in 7-bit Addressing mode. All decisions made by hardware or software and their effect on reception. Figure 25-14 and Figure 25-15 is used as a visual reference for this description.

This is a step by step process of what typically must be done to accomplish I²C communication.

- Start bit detected.
- S bit of SSPxSTAT is set; SSPxIF is set if interrupt on Start detect is enabled.
- 3. Matching address with R/W bit clear is received.
- The slave pulls SDAx low sending an ACK to the master, and sets SSPxIF bit.
- Software clears the SSPxIF bit.
- Software reads received address from SSPxBUF clearing the BF flag.
- 7. If SEN = 1; Slave software sets CKP bit to release the SCLx line.
- 8. The master clocks out a data byte.
- Slave drives SDAx low sending an ACK to the master, and sets SSPxIF bit.
- 10. Software clears SSPxIF.
- 11. Software reads the received byte from SSPxBUF clearing BF.
- 12. Steps 8-12 are repeated for all received bytes from the Master.
- Master sends Stop condition, setting P bit of SSPxSTAT, and the bus goes Idle.

25.5.1.5 7-bit Reception with AHEN and DHEN

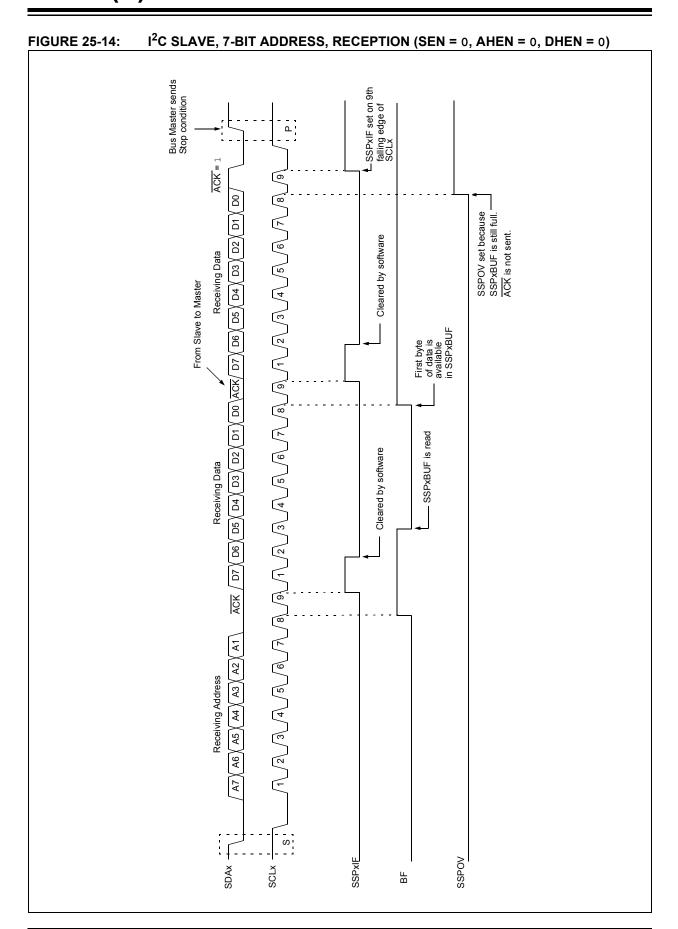
Slave device reception with AHEN and DHEN set operate the same as without these options with extra interrupts and clock stretching added after the 8th falling edge of SCLx. These additional interrupts allow the slave software to decide whether it wants to ACK the receive address or data byte, rather than the hardware. This functionality adds support for PMBus™ that was not present on previous versions of this module.

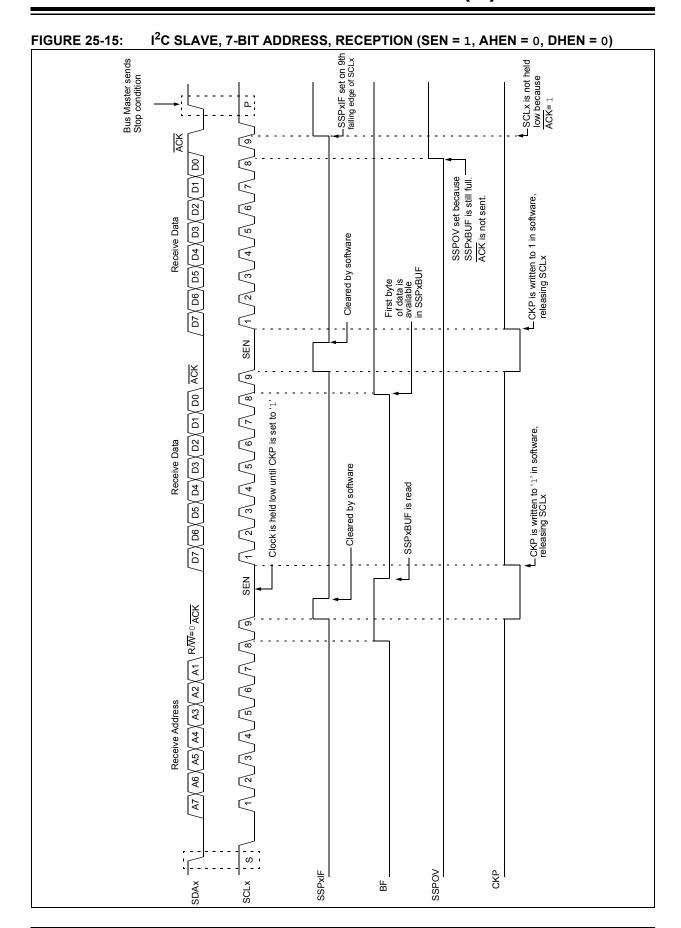
This list describes the steps that need to be taken by slave software to use these options for I²C communcation. Figure 25-16 displays a module using both address and data holding. Figure 25-17 includes the operation with the SEN bit of the SSPxCON2 register set

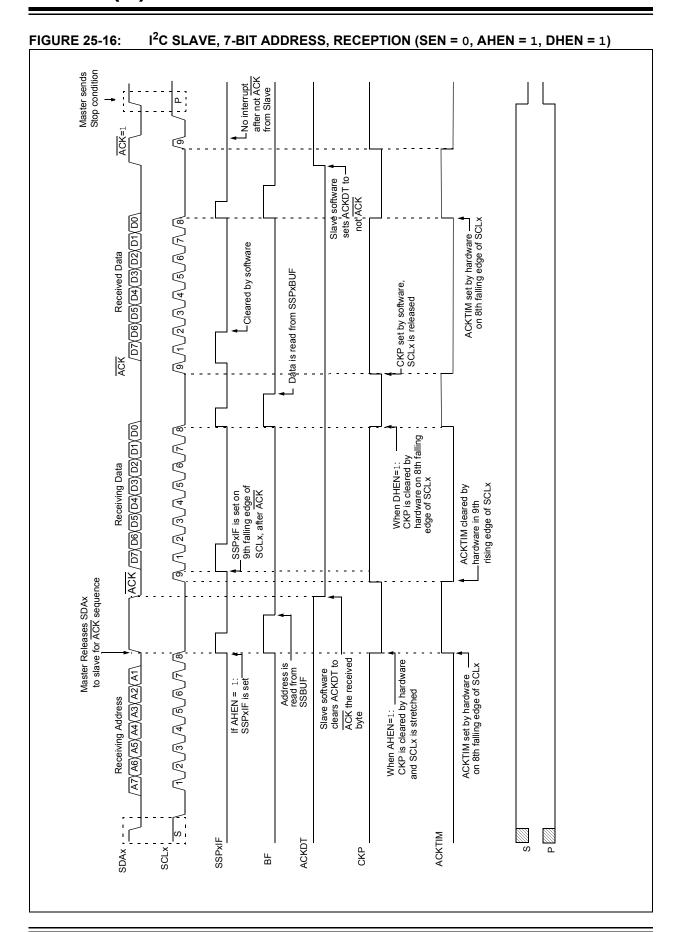
- 1. S bit of SSPxSTAT is set; SSPxIF is set if interrupt on Start detect is enabled.
- Matching address with R/W bit clear is clocked in. SSPxIF is set and CKP cleared after the 8th falling edge of SCLx.
- 3. Slave clears the SSPxIF.
- Slave can look at the ACKTIM bit of the SSPxCON3 register to determine if the SSPxIF was after or before the ACK.
- Slave reads the address value from SSPxBUF, clearing the BF flag.
- 6. Slave sets ACK value clocked out to the master by setting ACKDT.
- 7. Slave releases the clock by setting CKP.
- 8. SSPxIF is set after an ACK, not after a NACK.
- 9. If SEN = 1 the slave hardware will stretch the clock after the ACK.
- 10. Slave clears SSPxIF.

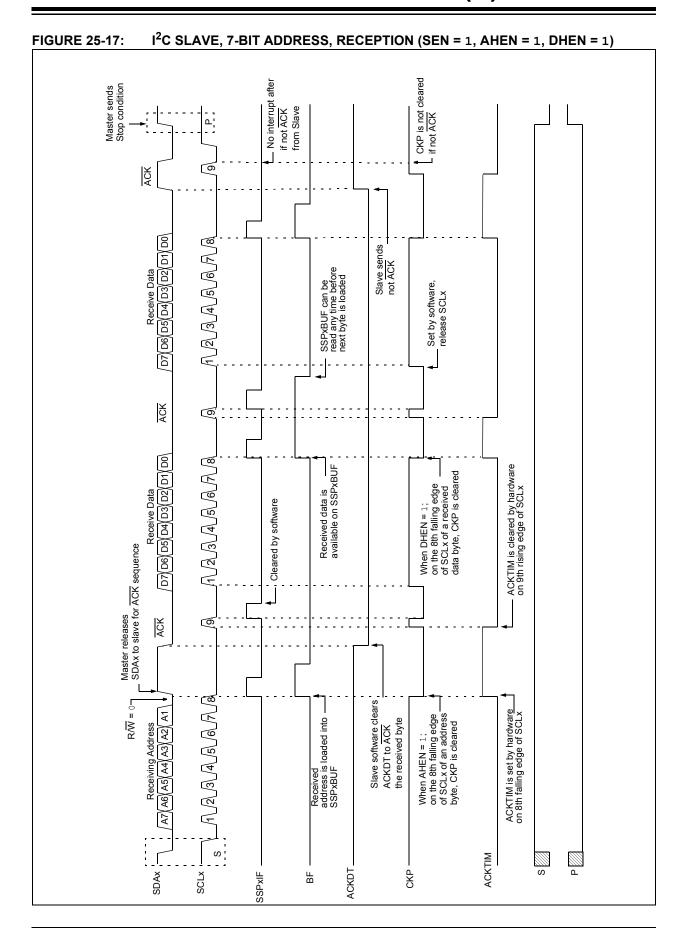
Note: SSPxIF is still set after the 9th falling edge of SCLx even if there is no clock stretching and BF has been cleared. Only if NACK is sent to Master is SSPxIF not set

- 11. SSPxIF set and CKP cleared after 8th falling edge of SCLx for a received data byte.
- 12. Slave looks at ACKTIM bit of SSPxCON3 to determine the source of the interrupt.
- 13. Slave reads the received data from SSPxBUF clearing BF.
- 14. Steps 7-14 are the same for each received data byte.
- 15. Communication is ended by either the slave sending an \overline{ACK} = 1, or the master sending a Stop condition. If a Stop is sent and Interrupt on Stop Detect is disabled, the slave will only know by polling the P bit of the SSTSTAT register.









25.5.2 SLAVE TRANSMISSION

When the R/\overline{W} bit of the incoming address byte is set and an address match occurs, the R/\overline{W} bit of the SSPxSTAT register is set. The received address is loaded into the SSPxBUF register, and an \overline{ACK} pulse is sent by the slave on the ninth bit.

Following the ACK, slave hardware clears the CKP bit and the SCLx pin is held low (see **Section 25.5.5** "Clock Stretching" for more detail). By stretching the clock, the master will be unable to assert another clock pulse until the slave is done preparing the transmit data.

The transmit data must be loaded into the SSPxBUF register which also loads the SSPxSR register. Then the SCLx pin should be released by setting the CKP bit of the SSPxCON1 register. The eight data bits are shifted out on the falling edge of the SCLx input. This ensures that the SDAx signal is valid during the SCLx high time.

The ACK pulse from the master-receiver is latched on the rising edge of the ninth SCLx input pulse. This \overline{ACK} value is copied to the ACKSTAT bit of the SSPxCON2 register. If ACKSTAT is set (not \overline{ACK}), then the data transfer is complete. In this case, when the not \overline{ACK} is latched by the slave, the slave goes Idle and waits for another occurrence of the Start bit. If the SDAx line was low (\overline{ACK}), the next transmit data must be loaded into the SSPxBUF register. Again, the SCLx pin must be released by setting bit CKP.

An MSSPx interrupt is generated for each data transfer byte. The SSPxIF bit must be cleared by software and the SSPxSTAT register is used to determine the status of the byte. The SSPxIF bit is set on the falling edge of the ninth clock pulse.

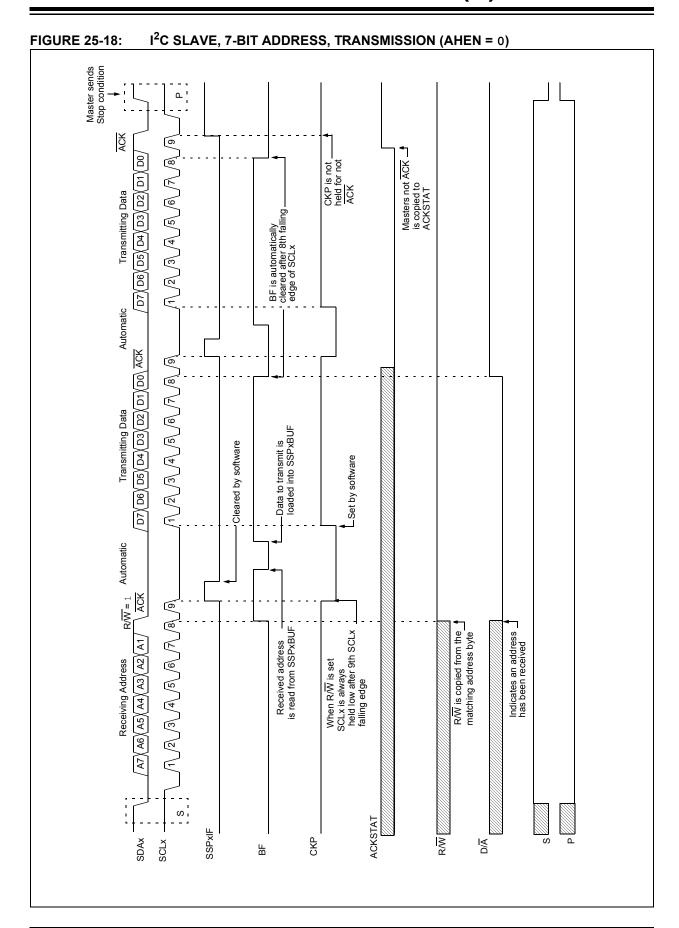
25.5.2.1 Slave Mode Bus Collision

A slave receives a Read request and begins shifting data out on the SDAx line. If a bus collision is detected and the SBCDE bit of the SSPxCON3 register is set, the BCLxIF bit of the PIRx register is set. Once a bus collision is detected, the slave goes Idle and waits to be addressed again. User software can use the BCLxIF bit to handle a slave bus collision.

25.5.2.2 7-bit Transmission

A master device can transmit a read request to a slave, and then clock data out of the slave. The list below outlines what software for a slave will need to do to accomplish a standard transmission. Figure 25-18 can be used as a reference to this list.

- Master sends a Start condition on SDAx and SCLx.
- S bit of SSPxSTAT is set; SSPxIF is set if interrupt on Start detect is enabled.
- 3. Matching address with R/W bit set is received by the Slave setting SSPxIF bit.
- Slave hardware generates an ACK and sets SSPxIF.
- 5. SSPxIF bit is cleared by user.
- Software reads the received address from SSPxBUF, clearing BF.
- 7. R/W is set so CKP was automatically cleared after the ACK.
- The slave software loads the transmit data into SSPxBUF.
- 9. CKP bit is set releasing SCLx, allowing the master to clock the data out of the slave.
- 10. SSPxIF is set after the ACK response from the master is loaded into the ACKSTAT register.
- 11. SSPxIF bit is cleared.
- 12. The slave software checks the ACKSTAT bit to see if the master wants to clock out more data.
 - **Note 1:** If the master $\overline{\mathsf{ACK}}$ s the clock will be stretched.
 - **2:** ACKSTAT is the only bit updated on the rising edge of SCLx (9th) rather than the falling.
- 13. Steps 9-13 are repeated for each transmitted byte.
- 14. If the master sends a not \overline{ACK} ; the clock is not held, but SSPxIF is still set.
- 15. The master sends a Restart condition or a Stop.
- 16. The slave is no longer addressed.



25.5.2.3 7-bit Transmission with Address Hold Enabled

Setting the AHEN bit of the SSPxCON3 register enables additional clock stretching and interrupt generation after the 8th falling edge of a received matching address. Once a matching address has been clocked in, CKP is cleared and the SSPxIF interrupt is set.

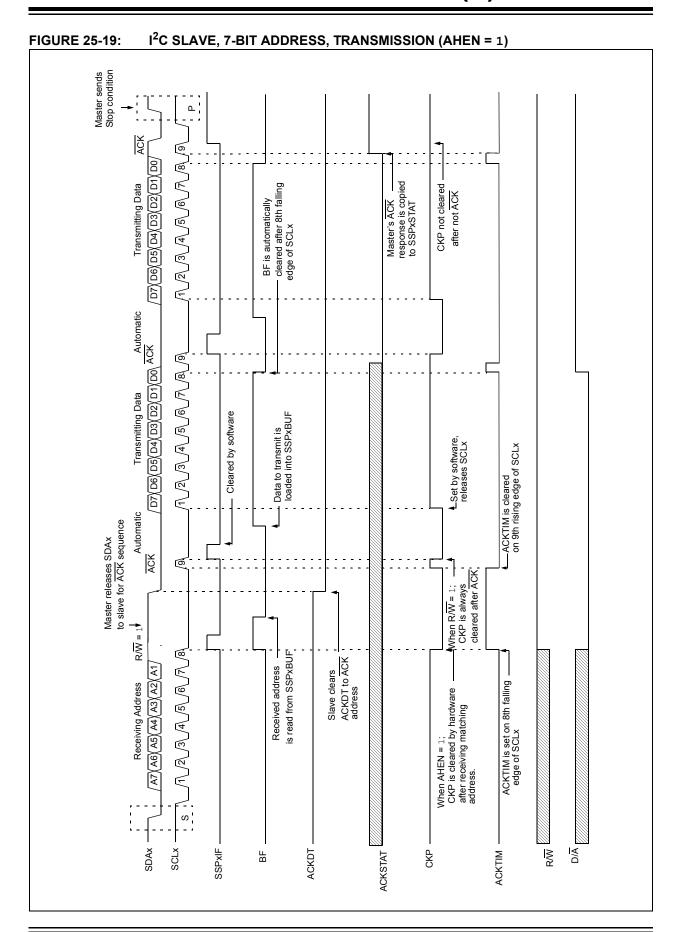
Figure 25-19 displays a standard waveform of a 7-bit Address Slave Transmission with AHEN enabled.

- 1. Bus starts Idle.
- Master sends Start condition; the S bit of SSPxSTAT is set; SSPxIF is set if interrupt on Start detect is enabled.
- Master sends matching address with R/W bit set. After the 8th falling edge of the SCLx line the CKP bit is cleared and SSPxIF interrupt is generated.
- 4. Slave software clears SSPxIF.
- Slave software reads ACKTIM bit of SSPxCON3 register, and R/W and D/A of the SSPxSTAT register to determine the source of the interrupt.
- 6. Slave reads the address value from the SSPxBUF register clearing the BF bit.
- Slave software decides from this information if it wishes to ACK or not ACK and sets the ACKDT bit of the SSPxCON2 register accordingly.
- 8. Slave sets the CKP bit releasing SCLx.
- 9. Master clocks in the ACK value from the slave.
- Slave hardware automatically clears the CKP bit and sets SSPxIF after the ACK if the R/W bit is set.
- 11. Slave software clears SSPxIF.
- 12. Slave loads value to transmit to the master into SSPxBUF setting the BF bit.

Note: $\frac{\text{SSPxBUF}}{\text{ACK}}$ cannot be loaded until after the

- 13. Slave sets CKP bit releasing the clock.
- 14. Master clocks out the data from the slave and sends an \overline{ACK} value on the 9th SCLx pulse.
- 15. Slave hardware copies the ACK value into the ACKSTAT bit of the SSPxCON2 register.
- 16. Steps 10-15 are repeated for each byte transmitted to the master from the slave.
- 17. If the master sends a not \overline{ACK} the slave releases the bus allowing the master to send a Stop and end the communication.

Note: Master must send a not ACK on the last byte to ensure that the slave releases the SCLx line to receive a Stop.



25.5.3 SLAVE MODE 10-BIT ADDRESS RECEPTION

This section describes a standard sequence of events for the MSSPx module configured as an I²C Slave in 10-bit Addressing mode.

Figure 25-20 is used as a visual reference for this description.

This is a step by step process of what must be done by slave software to accomplish I²C communication.

- Bus starts Idle.
- Master sends Start condition; S bit of SSPxSTAT is set; SSPxIF is set if interrupt on Start detect is enabled
- Master sends matching high address with the R/W bit clear; UA bit of the SSPxSTAT register is set.
- 4. Slave sends ACK and SSPxIF is set.
- 5. Software clears the SSPxIF bit.
- Software reads received address from SSPxBUF clearing the BF flag.
- Slave loads low address into SSPxADD, releasing SCLx.
- Master sends matching low address byte to the slave; UA bit is set.

Note: Updates to the SSPxADD register are not allowed until after the ACK sequence.

Slave sends ACK and SSPxIF is set.

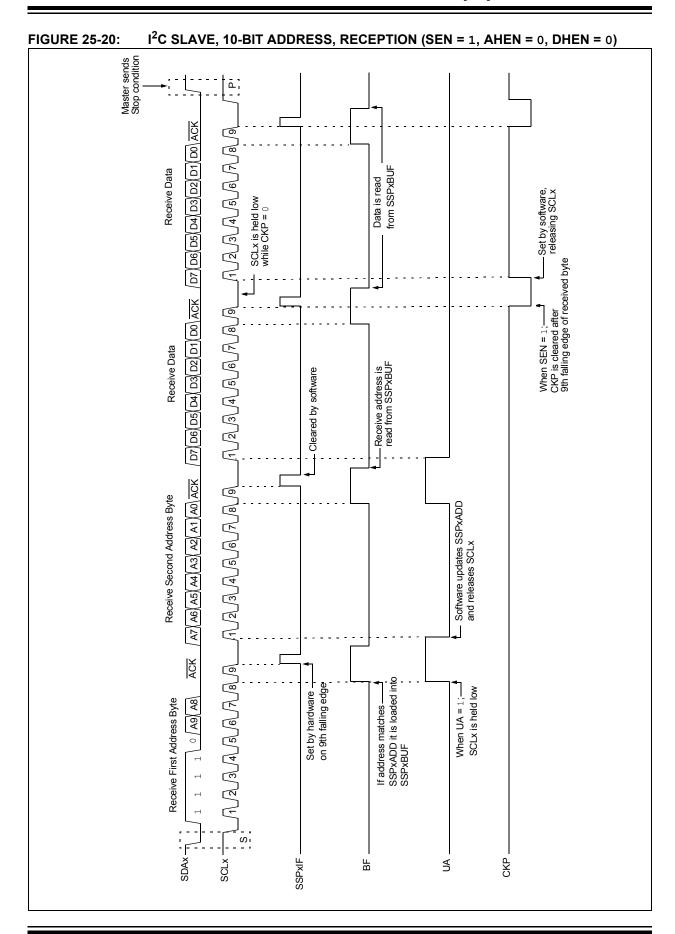
Note: If the low address does not match, SSPxIF and UA are still set so that the slave software can set SSPxADD back to the high address. BF is not set because there is no match. CKP is unaffected.

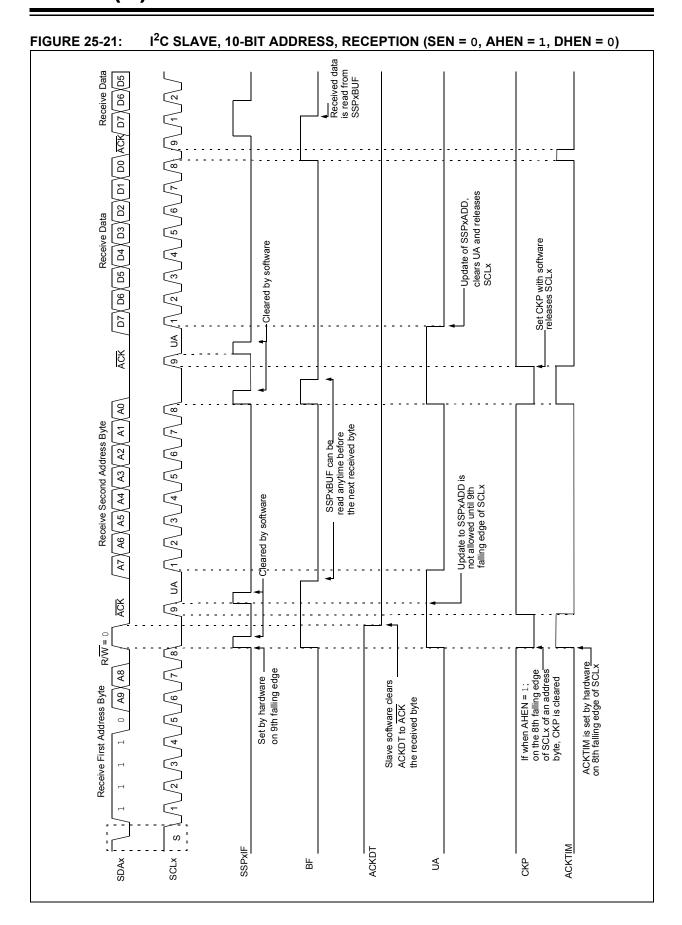
- 10. Slave clears SSPxIF.
- 11. Slave reads the received matching address from SSPxBUF clearing BF.
- 12. Slave loads high address into SSPxADD.
- Master clocks a data <u>byte</u> to the slave and clocks out the slaves ACK on the 9th SCLx pulse; SSPxIF is set.
- 14. If SEN bit of SSPxCON2 is set, CKP is cleared by hardware and the clock is stretched.
- 15. Slave clears SSPxIF.
- Slave reads the received byte from SSPxBUF clearing BF.
- If SEN is set the slave sets CKP to release the SCLx.
- 18. Steps 13-17 repeat for each received byte.
- 19. Master sends Stop to end the transmission.

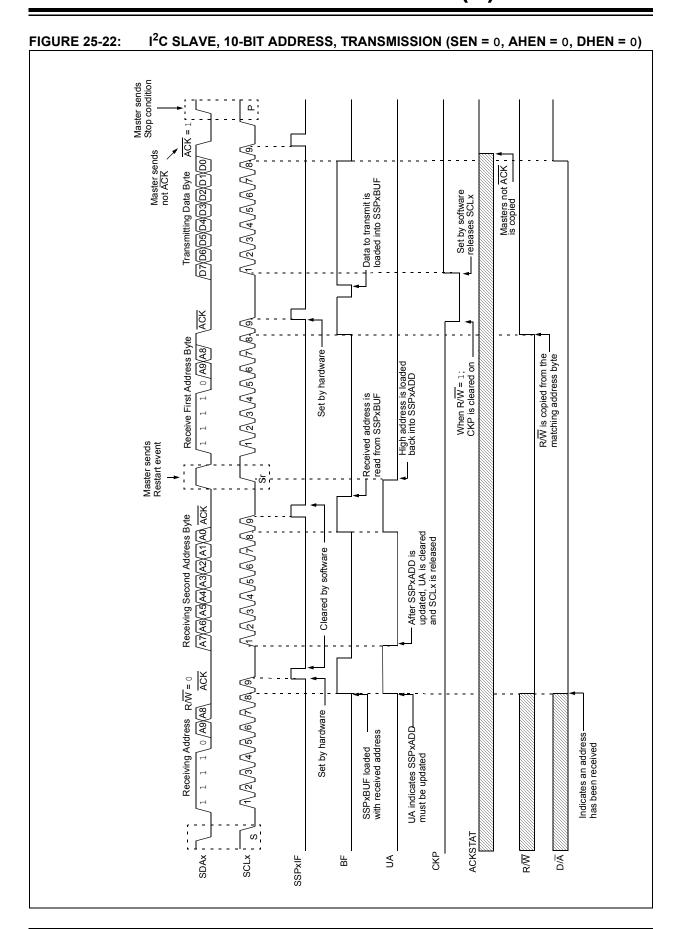
25.5.4 10-BIT ADDRESSING WITH ADDRESS OR DATA HOLD

Reception using 10-bit addressing with AHEN or DHEN set is the same as with 7-bit modes. The only difference is the need to update the SSPxADD register using the UA bit. All functionality, specifically when the CKP bit is cleared and SCLx line is held low are the same. Figure 25-21 can be used as a reference of a slave in 10-bit addressing with AHEN set.

Figure 25-22 shows a standard waveform for a slave transmitter in 10-bit Addressing mode.







25.5.5 CLOCK STRETCHING

Clock stretching occurs when a device on the bus holds the SCLx line low effectively pausing communication. The slave may stretch the clock to allow more time to handle data or prepare a response for the master device. A master device is not concerned with stretching as anytime it is active on the bus and not transferring data it is stretching. Any stretching done by a slave is invisible to the master software and handled by the hardware that generates SCLx.

The CKP bit of the SSPxCON1 register is used to control stretching in software. Any time the CKP bit is cleared, the module will wait for the SCLx line to go low and then hold it. Setting CKP will release SCLx and allow more communication.

25.5.5.1 Normal Clock Stretching

Following an ACK if the R/W bit of SSPxSTAT is set, a read request, the slave hardware will clear CKP. This allows the slave time to update SSPxBUF with data to transfer to the master. If the SEN bit of SSPxCON2 is set, the slave hardware will always stretch the clock after the ACK sequence. Once the slave is ready; CKP is set by software and communication resumes.

- Note 1: The BF bit has no effect on if the clock will be stretched or not. This is different than previous versions of the module that would not stretch the clock, clear CKP, if SSPxBUF was read before the 9th falling edge of SCLx.
 - 2: Previous versions of the module did not stretch the clock for a transmission if SSPxBUF was loaded before the 9th falling edge of SCLx. It is now always cleared for read requests.

25.5.5.2 10-bit Addressing Mode

In 10-bit Addressing mode, when the UA bit is set, the clock is always stretched. This is the only time the SCLx is stretched without CKP being cleared. SCLx is released immediately after a write to SSPxADD.

Note: Previous versions of the module did not stretch the clock if the second address byte did not match.

25.5.5.3 Byte NACKing

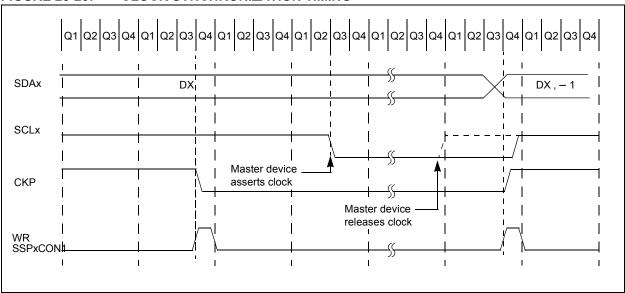
When AHEN bit of SSPxCON3 is set; CKP is cleared by hardware after the 8th falling edge of SCLx for a received matching address byte. When DHEN bit of SSPxCON3 is set; CKP is cleared after the 8th falling edge of SCLx for received data.

Stretching after the 8th falling edge of SCLx allows the slave to look at the received address or data and decide if it wants to ACK the received data.

25.5.6 CLOCK SYNCHRONIZATION AND THE CKP BIT

Any time the CKP bit is cleared, the module will wait for the SCLx line to go low and then hold it. However, clearing the CKP bit will not assert the SCLx output low until the SCLx output is already sampled low. Therefore, the CKP bit will not assert the SCLx line until an external I²C master device has already asserted the SCLx line. The SCLx output will remain low until the CKP bit is set and all other devices on the I²C bus have released SCLx. This ensures that a write to the CKP bit will not violate the minimum high time requirement for SCLx (see Figure 25-23).





25.5.7 GENERAL CALL ADDRESS SUPPORT

The addressing procedure for the I²C bus is such that the first byte after the Start condition usually determines which device will be the slave addressed by the master device. The exception is the general call address which can address all devices. When this address is used, all devices should, in theory, respond with an acknowledge.

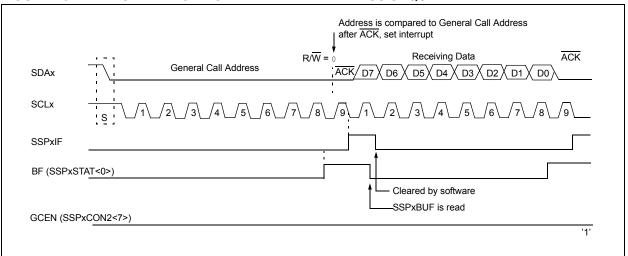
The general call address is a reserved address in the I^2C protocol, defined as address 0x00. When the GCEN bit of the SSPxCON2 register is set, the slave module will automatically \overline{ACK} the reception of this address regardless of the value stored in SSPxADD. After the slave clocks in an address of all zeros with the R/W bit clear, an interrupt is generated and slave

software can read SSPxBUF and respond. Figure 25-24 shows a general call reception sequence.

In 10-bit Address mode, the UA bit will not be set on the reception of the general call address. The slave will prepare to receive the second byte as data, just as it would in 7-bit mode.

If the AHEN bit of the SSPxCON3 register is set, just as with any other address reception, the slave hardware will stretch the clock after the 8th falling edge of SCLx. The slave must then set its ACKDT value and release the clock with communication progressing as it would normally.





25.5.8 SSPX MASK REGISTER

An SSPx Mask (SSPMSK) register (Register 25-5) is available in I²C Slave mode as a mask for the value held in the SSPxSR register during an address comparison operation. A zero ('0') bit in the SSPMSK register has the effect of making the corresponding bit of the received address a "don't care".

This register is reset to all '1's upon any Reset condition and, therefore, has no effect on standard SSPx operation until written with a mask value.

The SSPx Mask register is active during:

- 7-bit Address mode: address compare of A<7:1>.
- 10-bit Address mode: address compare of A<7:0>
 only. The SSPx mask has no effect during the
 reception of the first (high) byte of the address.

25.6 I²C MASTER MODE

Master mode is enabled by setting and clearing the appropriate SSPM bits in the SSPxCON1 register and by setting the SSPEN bit. In Master mode, the SDAx and SCKx pins must be configured as inputs. The MSSP peripheral hardware will override the output driver TRIS controls when necessary to drive the pins low.

Master mode of operation is supported by interrupt generation on the detection of the Start and Stop conditions. The Stop (P) and Start (S) bits are cleared from a Reset or when the MSSPx module is disabled. Control of the $\rm I^2C$ bus may be taken when the P bit is set, or the bus is Idle.

In Firmware Controlled Master mode, user code conducts all I²C bus operations based on Start and Stop bit condition detection. Start and Stop condition detection is the only active circuitry in this mode. All other communication is done by the user software directly manipulating the SDAx and SCLx lines.

The following events will cause the SSPx Interrupt Flag bit, SSPxIF, to be set (SSPx interrupt, if enabled):

- · Start condition detected
- · Stop condition detected
- · Data transfer byte transmitted/received
- · Acknowledge transmitted/received
- · Repeated Start generated
 - Note 1: The MSSPx module, when configured in I²C Master mode, does not allow queueing of events. For instance, the user is not allowed to initiate a Start condition and immediately write the SSPxBUF register to initiate transmission before the Start condition is complete. In this case, the SSPxBUF will not be written to and the WCOL bit will be set, indicating that a write to the SSPxBUF did not occur
 - 2: When in Master mode, Start/Stop detection is masked and an interrupt is generated when the SEN/PEN bit is cleared and the generation is complete.

25.6.1 I²C MASTER MODE OPERATION

The master device generates all of the serial clock pulses and the Start and Stop conditions. A transfer is ended with a Stop condition or with a Repeated Start condition. Since the Repeated Start condition is also the beginning of the next serial transfer, the I²C bus will not be released.

In Master Transmitter mode, serial data is output through SDAx, while SCLx outputs the serial clock. The first byte transmitted contains the slave address of the receiving device (7 bits) and the Read/Write (R/W) bit. In this case, the R/W bit will be logic '0'. Serial data is transmitted eight bits at a time. After each byte is transmitted, an Acknowledge bit is received. Start and Stop conditions are output to indicate the beginning and the end of a serial transfer.

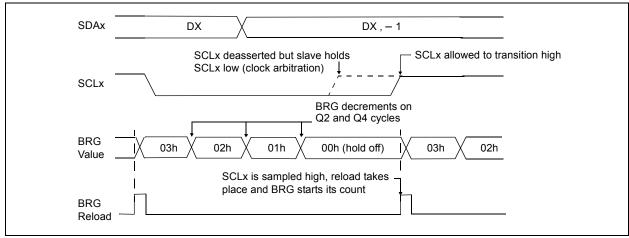
In Master Receive mode, the first byte transmitted contains the slave address of the transmitting device (7 bits) and the R/W bit. In this case, the R/W bit will be logic '1'. Thus, the first byte transmitted is a 7-bit slave address followed by a '1' to indicate the receive bit. Serial data is received via SDAx, while SCLx outputs the serial clock. Serial data is received eight bits at a time. After each byte is received, an Acknowledge bit is transmitted. Start and Stop conditions indicate the beginning and end of transmission.

A Baud Rate Generator is used to set the clock frequency output on SCLx. See Section 25.7 "Baud Rate Generator" for more detail.

25.6.2 CLOCK ARBITRATION

Clock arbitration occurs when the master, during any receive, transmit or Repeated Start/Stop condition, releases the SCLx pin (SCLx allowed to float high). When the SCLx pin is allowed to float high, the Baud Rate Generator (BRG) is suspended from counting until the SCLx pin is actually sampled high. When the SCLx pin is sampled high, the Baud Rate Generator is reloaded with the contents of SSPxADD<7:0> and begins counting. This ensures that the SCLx high time will always be at least one BRG rollover count in the event that the clock is held low by an external device (Figure 25-25).





25.6.3 WCOL STATUS FLAG

If the user writes the SSPxBUF when a Start, Restart, Stop, Receive or Transmit sequence is in progress, the WCOL bit is set and the contents of the buffer are unchanged (the write does not occur). Any time the WCOL bit is set it indicates that an action on SSPxBUF was attempted while the module was not Idle.

Note: Because queueing of events is not allowed, writing to the lower five bits of SSPxCON2 is disabled until the Start condition is complete.

25.6.4 I²C MASTER MODE START CONDITION TIMING

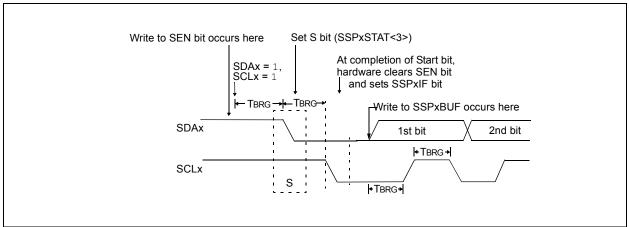
To initiate a Start condition (Figure 25-26), the user sets the Start Enable bit, SEN bit of the SSPxCON2 register. If the SDAx and SCLx pins are sampled high, the Baud Rate Generator is reloaded with the contents of SSPxADD<7:0> and starts its count. If SCLx and SDAx are both sampled high when the Baud Rate Generator times out (TBRG), the SDAx pin is driven low. The action of the SDAx being driven low while SCLx is high is the Start condition and causes the S bit of the SSPxSTAT1 register to be set. Following this, the Baud Rate Generator is reloaded with the contents of SSPxADD<7:0> and resumes its count. When the Baud Rate Generator times out (TBRG), the SEN bit of the SSPxCON2 register will be automatically cleared

by hardware; the Baud Rate Generator is suspended, leaving the SDAx line held low and the Start condition is complete.

Note 1: If at the beginning of the Start condition, the SDAx and SCLx pins are already sampled low, or if during the Start condition, the SCLx line is sampled low before the SDAx line is driven low, a bus collision occurs, the Bus Collision Interrupt Flag, BCLxIF, is set, the Start condition is aborted and the I²C module is reset into its Idle state.

2: The Philips I²C[™] Specification states that a bus collision cannot occur on a Start.

FIGURE 25-26: FIRST START BIT TIMING



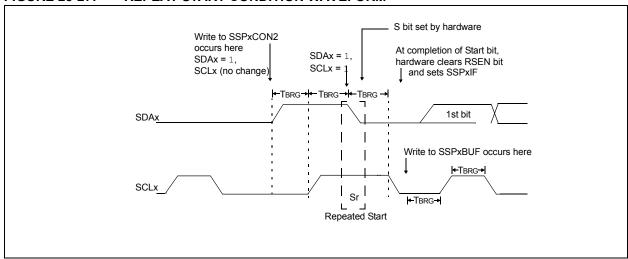
25.6.5 I²C MASTER MODE REPEATED START CONDITION TIMING

A Repeated Start condition (Figure 25-27) occurs when the RSEN bit of the SSPxCON2 register is programmed high and the Master state machine is no longer active. When the RSEN bit is set, the SCLx pin is asserted low. When the SCLx pin is sampled low, the Baud Rate Generator is loaded and begins counting. The SDAx pin is released (brought high) for one Baud Rate Generator count (TBRG). When the Baud Rate Generator times out, if SDAx is sampled high, the SCLx pin will be deasserted (brought high). When SCLx is sampled high, the Baud Rate Generator is reloaded and begins counting. SDAx and SCLx must be sampled high for one TBRG. This action is then followed by assertion of the SDAx pin (SDAx = 0) for one TBRG while SCLx is high. SCLx is asserted low. Following this, the RSEN bit of the SSPxCON2 register will be

automatically cleared and the Baud Rate Generator will not be reloaded, leaving the SDAx pin held low. As soon as a Start condition is detected on the SDAx and SCLx pins, the S bit of the SSPxSTAT register will be set. The SSPxIF bit will not be set until the Baud Rate Generator has timed out.

- **Note 1:** If RSEN is programmed while any other event is in progress, it will not take effect.
 - **2:** A bus collision during the Repeated Start condition occurs if:
 - SDAx is sampled low when SCLx goes from low-to-high.
 - SCLx goes low before SDAx is asserted low. This may indicate that another master is attempting to transmit a data '1'.

FIGURE 25-27: REPEAT START CONDITION WAVEFORM



25.6.6 I²C MASTER MODE TRANSMISSION

Transmission of a data byte, a 7-bit address or the other half of a 10-bit address is accomplished by simply writing a value to the SSPxBUF register. This action will set the Buffer Full flag bit, BF and allow the Baud Rate Generator to begin counting and start the next transmission. Each bit of address/data will be shifted out onto the SDAx pin after the falling edge of SCLx is asserted. SCLx is held low for one Baud Rate Generator rollover count (TBRG). Data should be valid before SCLx is released high. When the SCLx pin is released high, it is held that way for TBRG. The data on the SDAx pin must remain stable for that duration and some hold time after the next falling edge of SCLx. After the eighth bit is shifted out (the falling edge of the eighth clock), the BF flag is cleared and the master releases SDAx. This allows the slave device being addressed to respond with an ACK bit during the ninth bit time if an address match occurred, or if data was received properly. The status of ACK is written into the ACKSTAT bit on the rising edge of the ninth clock. If the master receives an Acknowledge, the Acknowledge Status bit, ACKSTAT, is cleared. If not, the bit is set. After the ninth clock, the SSPxIF bit is set and the master clock (Baud Rate Generator) is suspended until the next data byte is loaded into the SSPxBUF, leaving SCLx low and SDAx unchanged (Figure 25-28).

After the write to the SSPxBUF, each bit of the address will be shifted out on the falling edge of SCLx until all seven address bits and the R/W bit are completed. On the falling edge of the eighth clock, the master will release the SDAx pin, allowing the slave to respond with an Acknowledge. On the falling edge of the ninth clock, the master will sample the SDAx pin to see if the address was recognized by a slave. The status of the ACK bit is loaded into the ACKSTAT Status bit of the SSPxCON2 register. Following the falling edge of the ninth clock transmission of the address, the SSPxIF is set, the BF flag is cleared and the Baud Rate Generator is turned off until another write to the SSPxBUF takes place, holding SCLx low and allowing SDAx to float.

25.6.6.1 BF Status Flag

In Transmit mode, the BF bit of the SSPxSTAT register is set when the CPU writes to SSPxBUF and is cleared when all eight bits are shifted out.

25.6.6.2 WCOL Status Flag

If the user writes the SSPxBUF when a transmit is already in progress (i.e., SSPxSR is still shifting out a data byte), the WCOL bit is set and the contents of the buffer are unchanged (the write does not occur).

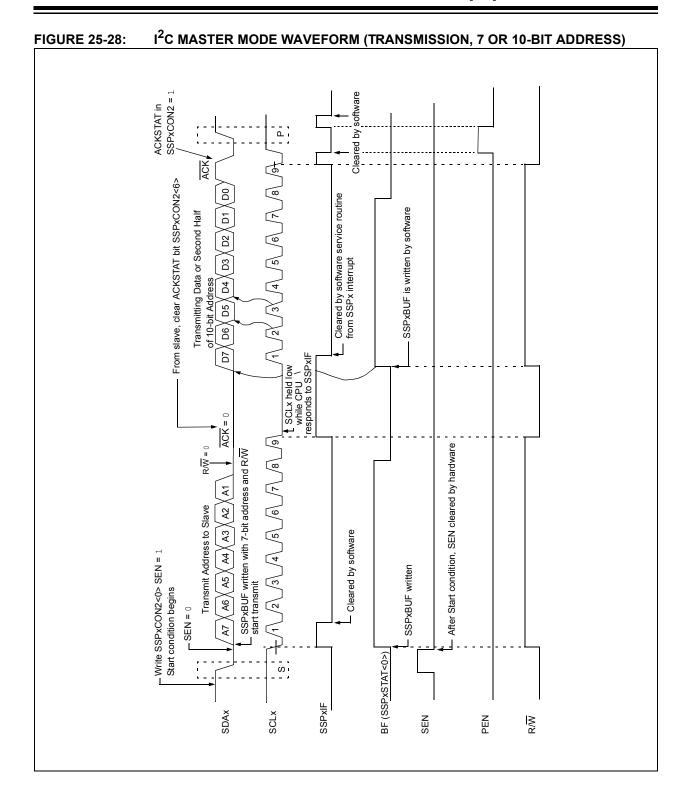
WCOL must be cleared by software before the next transmission.

25.6.6.3 ACKSTAT Status Flag

In Transmit mode, the ACKSTAT bit of the SSPxCON2 register is cleared when the slave has sent an Acknowledge ($\overline{ACK} = 0$) and is set when the slave does not Acknowledge ($\overline{ACK} = 1$). A slave sends an Acknowledge when it has recognized its address (including a general call), or when the slave has properly received its data.

25.6.6.4 Typical Transmit Sequence:

- 1. The user generates a Start condition by setting the SEN bit of the SSPxCON2 register.
- SSPxIF is set by hardware on completion of the Start.
- 3. SSPxIF is cleared by software.
- 4. The MSSPx module will wait the required start time before any other operation takes place.
- The user loads the SSPxBUF with the slave address to transmit.
- Address is shifted out the SDAx pin until all eight bits are transmitted. Transmission begins as soon as SSPxBUF is written to.
- 7. The MSSPx module shifts in the ACK bit from the slave device and writes its value into the ACKSTAT bit of the SSPxCON2 register.
- The MSSPx module generates an interrupt at the end of the ninth clock cycle by setting the SSPxIF bit.
- The user loads the SSPxBUF with eight bits of data.
- Data is shifted out the SDAx pin until all eight bits are transmitted.
- The MSSPx module shifts in the ACK bit from the slave device and writes its value into the ACKSTAT bit of the SSPxCON2 register.
- 12. Steps 8-11 are repeated for all transmitted data bytes.
- 13. The user generates a Stop or Restart condition by setting the PEN or RSEN bits of the SSPxCON2 register. Interrupt is generated once the Stop/Restart condition is complete.



25.6.7 I²C MASTER MODE RECEPTION

Master mode reception (Figure 25-29) is enabled by programming the Receive Enable bit, RCEN bit of the SSPxCON2 register.

Note: The MSSPx module must be in an Idle state before the RCEN bit is set or the RCEN bit will be disregarded.

The Baud Rate Generator begins counting and on each rollover, the state of the SCLx pin changes (high-to-low/low-to-high) and data is shifted into the SSPxSR. After the falling edge of the eighth clock, the receive enable flag is automatically cleared, the contents of the SSPxSR are loaded into the SSPxBUF, the BF flag bit is set, the SSPxIF flag bit is set and the Baud Rate Generator is suspended from counting, holding SCLx low. The MSSPx is now in Idle state awaiting the next command. When the buffer is read by the CPU, the BF flag bit is automatically cleared. The user can then send an Acknowledge bit at the end of reception by setting the Acknowledge Sequence Enable, ACKEN bit of the SSPxCON2 register.

25.6.7.1 BF Status Flag

In receive operation, the BF bit is set when an address or data byte is loaded into SSPxBUF from SSPxSR. It is cleared when the SSPxBUF register is read.

25.6.7.2 SSPOV Status Flag

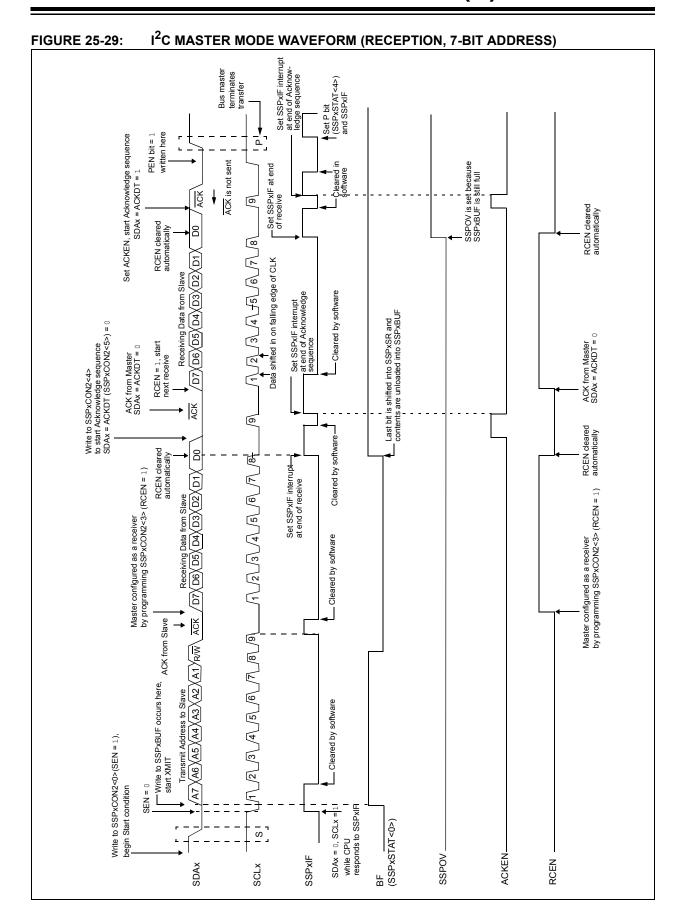
In receive operation, the SSPOV bit is set when eight bits are received into the SSPxSR and the BF flag bit is already set from a previous reception.

25.6.7.3 WCOL Status Flag

If the user writes the SSPxBUF when a receive is already in progress (i.e., SSPxSR is still shifting in a data byte), the WCOL bit is set and the contents of the buffer are unchanged (the write does not occur).

25.6.7.4 Typical Receive Sequence:

- The user generates a Start condition by setting the SEN bit of the SSPxCON2 register.
- SSPxIF is set by hardware on completion of the Start.
- 3. SSPxIF is cleared by software.
- User writes SSPxBUF with the slave address to transmit and the R/W bit set.
- 5. Address is shifted out the SDAx pin until all eight bits are transmitted. Transmission begins as soon as SSPxBUF is written to.
- The MSSPx module shifts in the ACK bit from the slave device and writes its value into the ACKSTAT bit of the SSPxCON2 register.
- The MSSPx module generates an interrupt at the end of the ninth clock cycle by setting the SSPxIF bit.
- User sets the RCEN bit of the SSPxCON2 register and the Master clocks in a byte from the slave.
- After the 8th falling edge of SCLx, SSPxIF and BF are set.
- Master clears SSPxIF and reads the received byte from SSPxUF, clears BF.
- Master sets ACK value sent to slave in ACKDT bit of the SSPxCON2 register and initiates the ACK by setting the ACKEN bit.
- 12. Masters ACK is clocked out to the slave and SSPxIF is set.
- 13. User clears SSPxIF.
- Steps 8-13 are repeated for each received byte from the slave.
- 15. Master sends a not ACK or Stop to end communication.



25.6.8 ACKNOWLEDGE SEQUENCE TIMING

An Acknowledge sequence is enabled by setting the Acknowledge Sequence Enable bit, ACKEN bit of the SSPxCON2 register. When this bit is set, the SCLx pin is pulled low and the contents of the Acknowledge data bit are presented on the SDAx pin. If the user wishes to generate an Acknowledge, then the ACKDT bit should be cleared. If not, the user should set the ACKDT bit before starting an Acknowledge sequence. The Baud Rate Generator then counts for one rollover period (TBRG) and the SCLx pin is deasserted (pulled high). When the SCLx pin is sampled high (clock arbitration), the Baud Rate Generator counts for TBRG. The SCLx pin is then pulled low. Following this, the ACKEN bit is automatically cleared, the Baud Rate Generator is turned off and the MSSPx module then goes into Idle mode (Figure 25-30).

25.6.8.1 WCOL Status Flag

If the user writes the SSPxBUF when an Acknowledge sequence is in progress, then the WCOL bit is set and the contents of the buffer are unchanged (the write does not occur).

25.6.9 STOP CONDITION TIMING

A Stop bit is asserted on the SDAx pin at the end of a receive/transmit by setting the Stop Sequence Enable bit, PEN bit of the SSPxCON2 register. At the end of a receive/transmit, the SCLx line is held low after the falling edge of the ninth clock. When the PEN bit is set, the master will assert the SDAx line low. When the SDAx line is sampled low, the Baud Rate Generator is reloaded and counts down to '0'. When the Baud Rate Generator times out, the SCLx pin will be brought high and one TBRG (Baud Rate Generator rollover count) later, the SDAx pin will be deasserted. When the SDAx pin is sampled high while SCLx is high, the P bit of the SSPxSTAT register is set. A TBRG later, the PEN bit is cleared and the SSPxIF bit is set (Figure 25-31).

25.6.9.1 WCOL Status Flag

If the user writes the SSPxBUF when a Stop sequence is in progress, then the WCOL bit is set and the contents of the buffer are unchanged (the write does not occur).

FIGURE 25-30: ACKNOWLEDGE SEQUENCE WAVEFORM

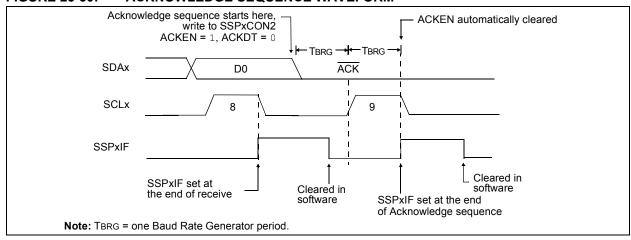
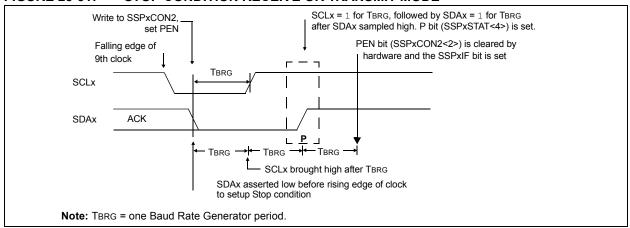


FIGURE 25-31: STOP CONDITION RECEIVE OR TRANSMIT MODE



25.6.10 SLEEP OPERATION

While in Sleep mode, the I²C slave module can receive addresses or data and when an address match or complete byte transfer occurs, wake the processor from Sleep (if the MSSPx interrupt is enabled).

25.6.11 EFFECTS OF A RESET

A Reset disables the MSSPx module and terminates the current transfer.

25.6.12 MULTI-MASTER MODE

In Multi-Master mode, the interrupt generation on the detection of the Start and Stop conditions allows the determination of when the bus is free. The Stop (P) and Start (S) bits are cleared from a Reset or when the MSSPx module is disabled. Control of the I²C bus may be taken when the P bit of the SSPxSTAT register is set, or the bus is Idle, with both the S and P bits clear. When the bus is busy, enabling the SSPx interrupt will generate the interrupt when the Stop condition occurs.

In multi-master operation, the SDAx line must be monitored for arbitration to see if the signal level is the expected output level. This check is performed by hardware with the result placed in the BCLxIF bit.

The states where arbitration can be lost are:

- · Address Transfer
- · Data Transfer
- · A Start Condition
- · A Repeated Start Condition
- An Acknowledge Condition

25.6.13 MULTI -MASTER COMMUNICATION, BUS COLLISION AND BUS ARBITRATION

Multi-Master mode support is achieved by bus arbitration. When the master outputs address/data bits onto the SDAx pin, arbitration takes place when the master outputs a '1' on SDAx, by letting SDAx float high and another master asserts a '0'. When the SCLx pin floats high, data should be stable. If the expected data on SDAx is a '1' and the data sampled on the SDAx pin is '0', then a bus collision has taken place. The master will set the Bus Collision Interrupt Flag, BCLxIF and reset the I²C port to its Idle state (Figure 25-32).

If a transmit was in progress when the bus collision occurred, the transmission is halted, the BF flag is cleared, the SDAx and SCLx lines are deasserted and the SSPxBUF can be written to. When the user services the bus collision Interrupt Service Routine and if the $\rm I^2C$ bus is free, the user can resume communication by asserting a Start condition.

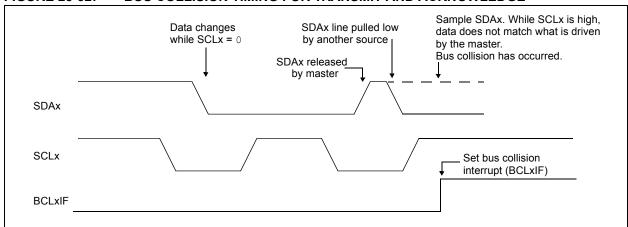
If a Start, Repeated Start, Stop or Acknowledge condition was in progress when the bus collision occurred, the condition is aborted, the SDAx and SCLx lines are deasserted and the respective control bits in the SSPxCON2 register are cleared. When the user services the bus collision Interrupt Service Routine and if the I²C bus is free, the user can resume communication by asserting a Start condition.

The master will continue to monitor the SDAx and SCLx pins. If a Stop condition occurs, the SSPxIF bit will be set.

A write to the SSPxBUF will start the transmission of data at the first data bit, regardless of where the transmitter left off when the bus collision occurred.

In Multi-Master mode, the interrupt generation on the detection of Start and Stop conditions allows the determination of when the bus is free. Control of the I²C bus can be taken when the P bit is set in the SSPxSTAT register, or the bus is Idle and the S and P bits are cleared.





25.6.13.1 Bus Collision During a Start Condition

During a Start condition, a bus collision occurs if:

- a) SDAx or SCLx are sampled low at the beginning of the Start condition (Figure 25-33).
- b) SCLx is sampled low before SDAx is asserted low (Figure 25-34).

During a Start condition, both the SDAx and the SCLx pins are monitored.

If the SDAx pin is already low, or the SCLx pin is already low, then all of the following occur:

- · the Start condition is aborted,
- · the BCLxIF flag is set and
- the MSSPx module is reset to its Idle state (Figure 25-33).

The Start condition begins with the SDAx and SCLx pins deasserted. When the SDAx pin is sampled high, the Baud Rate Generator is loaded and counts down. If the SCLx pin is sampled low while SDAx is high, a bus collision occurs because it is assumed that another master is attempting to drive a data '1' during the Start condition.

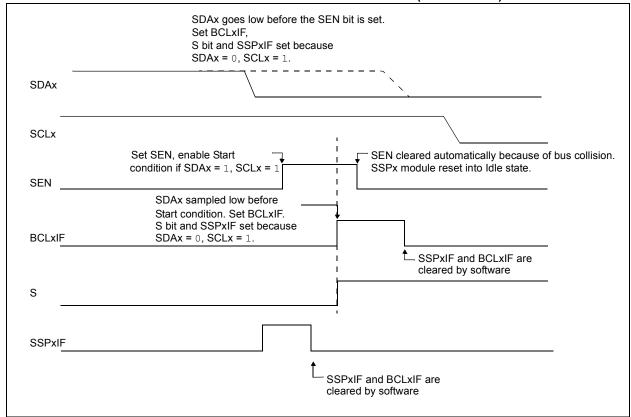
If the SDAx pin is sampled low during this count, the BRG is reset and the SDAx line is asserted early (Figure 25-35). If, however, a '1' is sampled on the

SDAx pin, the SDAx pin is asserted low at the end of the BRG count. The Baud Rate Generator is then reloaded and counts down to zero; if the SCLx pin is sampled as '0' during this time, a bus collision does not occur. At the end of the BRG count, the SCLx pin is asserted low.

Note:

The reason that bus collision is not a factor during a Start condition is that no two bus masters can assert a Start condition at the exact same time. Therefore, one master will always assert SDAx before the other. This condition does not cause a bus collision because the two masters must be allowed to arbitrate the first address following the Start condition. If the address is the same, arbitration must be allowed to continue into the data portion, Repeated Start or Stop conditions.

FIGURE 25-33: BUS COLLISION DURING START CONDITION (SDAX ONLY)





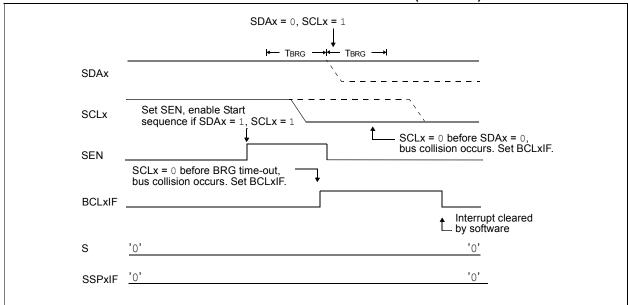
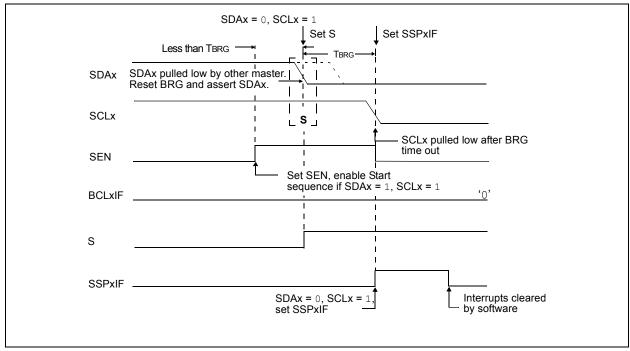


FIGURE 25-35: BRG RESET DUE TO SDA ARBITRATION DURING START CONDITION



25.6.13.2 Bus Collision During a Repeated Start Condition

During a Repeated Start condition, a bus collision occurs if:

- a) A low level is sampled on SDAx when SCLx goes from low level to high level. (CASE 1)
- b) SCLx goes low before SDAx is asserted low, indicating that another master is attempting to transmit a data '1'. (CASE 2)

When the user releases SDAx and the pin is allowed to float high, the BRG is loaded with SSPxADD and counts down to zero. The SCLx pin is then deasserted and when sampled high, the SDAx pin is sampled.

If SDAx is low, a bus collision has occurred (i.e., another master is attempting to transmit a data '0' (Figure 25-36). If SDAx is sampled high, the BRG is reloaded and begins counting. If SDAx goes from high-to-low before the BRG times out, no bus collision occurs because no two masters can assert SDAx at exactly the same time.

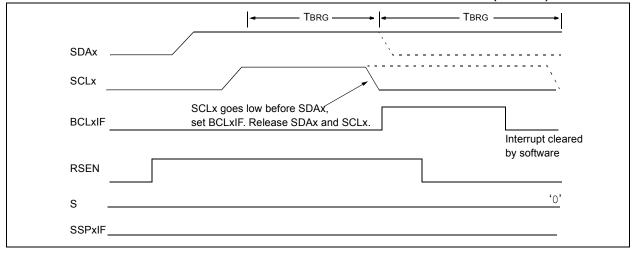
If SCLx goes from high-to-low before the BRG times out and SDAx has not already been asserted, a bus collision occurs. In this case, another master is attempting to transmit a data '1' during the Repeated Start condition (Figure 25-37).

If, at the end of the BRG time-out, both SCLx and SDAx are still high, the SDAx pin is driven low and the BRG is reloaded and begins counting. At the end of the count, regardless of the status of the SCLx pin, the SCLx pin is driven low and the Repeated Start condition is complete.

FIGURE 25-36: BUS COLLISION DURING A REPEATED START CONDITION (CASE 1)



FIGURE 25-37: BUS COLLISION DURING REPEATED START CONDITION (CASE 2)



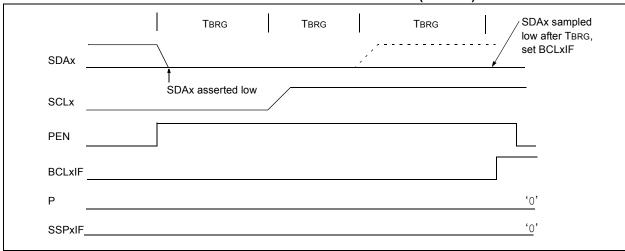
25.6.13.3 Bus Collision During a Stop Condition

Bus collision occurs during a Stop condition if:

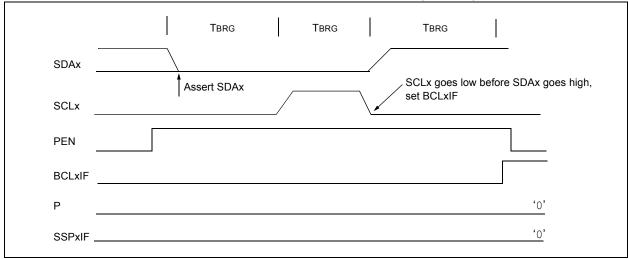
- After the SDAx pin has been deasserted and allowed to float high, SDAx is sampled low after the BRG has timed out. (CASE 1)
- b) After the SCLx pin is deasserted, SCLx is sampled low before SDAx goes high. (CASE 2)

The Stop condition begins with SDAx asserted low. When SDAx is sampled low, the SCLx pin is allowed to float. When the pin is sampled high (clock arbitration), the Baud Rate Generator is loaded with SSPxADD and counts down to zero. After the BRG times out, SDAx is sampled. If SDAx is sampled low, a bus collision has occurred. This is due to another master attempting to drive a data '0' (Figure 25-38). If the SCLx pin is sampled low before SDAx is allowed to float high, a bus collision occurs. This is another case of another master attempting to drive a data '0' (Figure 25-39).

FIGURE 25-38: BUS COLLISION DURING A STOP CONDITION (CASE 1)







SUMMARY OF REGISTERS ASSOCIATED WITH I²CTM OPERATION **TABLE 25-3:**

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset Values on Page
INLVLA	_	_	INLVLA5 ⁽¹⁾	INLVLA4	INLVLA3 ⁽²⁾	INLVLA2	INLVLA1	INLVLA0	127
INLVLB ⁽¹⁾	INLVLB7	INLVLB6	INLVLB5	INLVLB4	_	-	_	_	132
INLVLC	INLVLC7 ⁽¹⁾	INLVLC6 ⁽¹⁾	INLVLC5	INLVLC4	INLVLC3 ⁽²⁾	INLVLC2 ⁽²⁾	INLVLC1	INLVLC0	138
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	90
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	91
PIE2	OSFIE	C2IE	C1IE	EEIE	BCL1IE	_	_	CCP2IE	92
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	95
PIR2	OSFIF	C2IF	C1IF	EEIF	BCL1IF	_	_	CCP2IF	96
SSP1ADD	ADD7	ADD6	ADD5	ADD4	ADD3	ADD2	ADD1	ADD0	292
SSP1BUF	Synchronous Serial Port Receive Buffer/Transmit Register							245*	
SSP1CON1	WCOL	SSPOV	SSPEN	CKP	SSPM<3:0>				289
SSP1CON2	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	290
SSP1CON3	ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	291
SSP1MSK	MSK7	MSK6	MSK5	MSK4	MSK3	MSK2	MSK1	MSK0	292
SSP1STAT	SMP	CKE	D/Ā	Р	S	R/W	UA	BF	288
TRISA	_	_	TRISA5 ⁽¹⁾	TRISA4	TRISA3 ⁽²⁾	TRISA2	TRISA1	TRISA0	125
TRISB ⁽¹⁾	TRISB7	TRISB6	TRISB5	TRISB4	_	_	_	_	131
TRISC	TRISC7 ⁽¹⁾	TRISC6 ⁽¹⁾	TRISC5	TRISC4	TRISC3 ⁽²⁾	TRISC2 ⁽²⁾	TRISC1	TRISC0	136

— Unimplemented location, read as '0'. Shaded cells are not used by the MSSP module in I²C™ mode.

* Page provides register information.
PIC16(L)F1829 only.
PIC16(L)F1825 only. Legend:

Note 1:

25.7 BAUD RATE GENERATOR

The MSSPx module has a Baud Rate Generator available for clock generation in both I^2C and SPI Master modes. The Baud Rate Generator (BRG) reload value is placed in the SSPxADD register (Register 25-6). When a write occurs to SSPxBUF, the Baud Rate Generator will automatically begin counting down.

Once the given operation is complete, the internal clock will automatically stop counting and the clock pin will remain in its last state.

An internal signal "Reload" in Figure 25-40 triggers the value from SSPxADD to be loaded into the BRG counter. This occurs twice for each oscillation of the

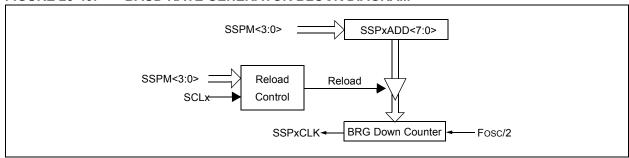
module clock line. The logic dictating when the reload signal is asserted depends on the mode the MSSPx is being operated in.

Table 25-4 demonstrates clock rates based on instruction cycles and the BRG value loaded into SSPxADD.

EQUATION 25-1:

$$FCLOCK = \frac{FOSC}{(SSPxADD + I)(4)}$$

FIGURE 25-40: BAUD RATE GENERATOR BLOCK DIAGRAM



Note: Values of 0x00, 0x01 and 0x02 are not valid for SSPxADD when used as a Baud Rate Generator for I²C. This is an implementation limitation.

TABLE 25-4: MSSPX CLOCK RATE W/BRG

Fosc	Fcy	BRG Value	FCLOCK (2 Rollovers of BRG)	
32 MHz	8 MHz	13h	400 kHz ⁽¹⁾	
32 MHz	8 MHz	19h	308 kHz	
32 MHz	8 MHz	4Fh	100 kHz	
16 MHz	4 MHz	09h	400 kHz ⁽¹⁾	
16 MHz	4 MHz	0Ch	308 kHz	
16 MHz	4 MHz	27h	100 kHz	
4 MHz	1 MHz	09h	100 kHz	

Note 1: The I²C interface does not conform to the 400 kHz I²C specification (which applies to rates greater than 100 kHz) in all details, but may be used with care where higher rates are required by the application.

REGISTER 25-1: SSPxSTAT: SSPx STATUS REGISTER

R/W-0/0	R/W-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0
SMP	CKE	D/ A	Р	S	R/W	UA	BF
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

bit 7 SMP: SPI Data Input Sample bit

SPI Master mode:

1 = Input data sampled at end of data output time

0 = Input data sampled at middle of data output time

SPI Slave mode:

SMP must be cleared when SPI is used in Slave mode

In I²C Master or Slave mode:

1 = Slew rate control disabled for standard speed mode (100 kHz and 1 MHz)

0 = Slew rate control enabled for high speed mode (400 kHz)

bit 6 CKE: SPI Clock Edge Select bit (SPI mode only)

In SPI Master or Slave mode:

 $\ensuremath{\mathtt{1}}$ = Transmit occurs on transition from active to Idle clock state

 \circ = Transmit occurs on transition from Idle to active clock state

In I²C™ mode only:

1 = Enable input logic so that thresholds are compliant with SMbus specification

0 = Disable SMbus specific inputs

bit 5 **D/A**: Data/Address bit (I²C mode only)

1 = Indicates that the last byte received or transmitted was data

 $\ensuremath{\textsc{0}}$ = Indicates that the last byte received or transmitted was address

bit 4 P: Stop bit

 $(l^2C \text{ mode only. This bit is cleared when the MSSPx module is disabled, SSPEN is cleared.)}$

1 = Indicates that a Stop bit has been detected last (this bit is '0' on Reset)

0 = Stop bit was not detected last

bit 3 S: Start bit

(I²C mode only. This bit is cleared when the MSSPx module is disabled, SSPEN is cleared.)

1 = Indicates that a Start bit has been detected last (this bit is '0' on Reset)

0 = Start bit was not detected last

bit 2 **R/W**: Read/Write bit information (I²C mode only)

This bit holds the R/\overline{W} bit information following the last address match. This bit is only valid from the address match

to the next Start bit, Stop bit, or not ACK bit.

In I²C Slave mode:

1 = Read

0 = Write

In I²C Master mode:

1 = Transmit is in progress

0 = Transmit is not in progress

OR-ing this bit with SEN, RSEN, PEN, RCEN or ACKEN will indicate if the MSSPx is in Idle mode.

bit 1 **UA:** Update Address bit (10-bit I²C mode only)

1 = Indicates that the user needs to update the address in the SSPxADD register

0 = Address does not need to be updated

bit 0 BF: Buffer Full Status bit

Receive (SPI and I²C modes):

1 = Receive complete, SSPxBUF is full

0 = Receive not complete, SSPxBUF is empty

Transmit (I²C mode only):

1 = Data transmit in progress (does not include the ACK and Stop bits), SSPxBUF is full

0 = Data transmit complete (does not include the ACK and Stop bits), SSPxBUF is empty

REGISTER 25-2: SSPxCON1: SSPx CONTROL REGISTER 1

R/C/HS-0/0	R/C/HS-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
WCOL	SSPOV	SSPEN	CKP	SSPM<3:0>			
bit 7							

Legend: R = Readable bit W = Writable bit U = Unimplemented bit, read as '0' u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets HS = Bit is set by hardware '1' = Bit is set '0' = Bit is cleared C = User cleared

bit 7 WCOL: Write Collision Detect bit

Master mode:

1 = A write to the SSPxBUF register was attempted while the I²C conditions were not valid for a transmission to be started

n = No collision

Slave mode:

1 = The SSPxBUF register is written while it is still transmitting the previous word (must be cleared in software)

0 = No collision

SSPOV: Receive Overflow Indicator bit(1) bit 6

In SPI mode:

A new byte is received while the SSPxBUF register is still holding the previous data. In case of overflow, the data in SSPxSR is lost. Overflow can only occur in Slave mode. In Slave mode, the user must read the SSPxBUF, even if only transmitting data, to avoid setting overflow. In Master mode, the overflow bit is not set since each new reception (and transmission) is initiated by writing to the SSPxBUF register (must be cleared in software).

No overflow

In 12C mode:

1 = A byte is received while the SSPxBUF register is still holding the previous byte. SSPOV is a "don't care" in Transmit mode

No overflow

bit 5 SSPEN: Synchronous Serial Port Enable bit

In both modes, when enabled, these pins must be properly configured as input or output

In SPI mode:

1 = Enables serial port and configures SCKx, SDOx, SDIx and SSx as the source of the serial port pins⁽²⁾

Disables serial port and configures these pins as I/O port pins

Enables the serial port and configures the SDAx and SCLx pins as the source of the serial port pins⁽³⁾

0 = Disables serial port and configures these pins as I/O port pins

bit 4 CKP: Clock Polarity Select bit

In SPI mode:
1 = Idle state for clock is a high level

0 = Idle state for clock is a low level

In I2C Slave mode:

1 = Enable clock

0 = Holds clock low (clock stretch). (Used to ensure data setup time.)

In I²C Master mode: Unused in this mode

bit 3-0 SSPM<3:0>: Synchronous Serial Port Mode Select bits

0000 = SPI Master mode, clock = Fosc/4

0001 = SPI Master mode, clock = Fosc/16

0010 = SPI Master mode, clock = Fosc/64

0011 = SPI Master mode, clock = TMR2 output/2 0100 = SPI Slave mode, clock = SCKx pin, <u>SSx</u> pin control enabled

0101 = SPI Slave mode, clock = SCKx pin, SSx pin control disabled, SSx can be used as I/O pin

 $0110 = I^2C$ Slave mode, 7-bit address

 $0111 = I^2C$ Slave mode, 10-bit address

1000 = I^2C Master mode, clock = Fosc / $(4 * (SSPxADD+1))^{(4)}$

1001 = Reserved

1010 = SPI Master mode, clock = Fosc/(4 * (SSPxADD+1))(5)

1011 = I²C firmware controlled Master mode (Slave Idle)

1100 = Reserved

1101 = Reserved

1110 = I²C Slave mode, 7-bit address with Start and Stop bit interrupts enabled

1111 = I²C Slave mode, 10-bit address with Start and Stop bit interrupts enabled

Note 1: In Master mode, the overflow bit is not set since each new reception (and transmission) is initiated by writing to the SSPxBUF register.

- 2: When enabled, these pins must be properly configured as input or output.
- When enabled, the SDAx and SCLx pins must be configured as inputs. 3:
- 4: SSPxADD values of 0, 1 or 2 are not supported for I²C Mode
- SSPxADD value of '0' is not supported. Use SSPM = 0000 instead.

REGISTER 25-3: SSPxCON2: SSPx CONTROL REGISTER 2

R/W-0/0	R-0/0	R/W-0/0	R/S/HS-0/0	R/S/HS-0/0	R/S/HS-0/0	R/S/HS-0/0	R/W/HS-0/0
GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared HC = Cleared by hardware S = User set

bit 7 **GCEN:** General Call Enable bit (in I²C Slave mode only)

1 = Enable interrupt when a general call address (0x00 or 00h) is received in the SSPxSR

0 = General call address disabled

bit 6 **ACKSTAT:** Acknowledge Status bit (in I²C mode only)

1 = Acknowledge was not received0 = Acknowledge was received

bit 5 **ACKDT:** Acknowledge Data bit (in I²C mode only)

In Receive mode:

Value transmitted when the user initiates an Acknowledge sequence at the end of a receive

1 = Not Acknowledge

0 = Acknowledge

bit 4 **ACKEN:** Acknowledge Sequence Enable bit (in I²C Master mode only)

In Master Receive mode:

1 = Initiate Acknowledge sequence on SDAx and SCLx pins, and transmit ACKDT data bit. Automatically cleared by hardware.

0 = Acknowledge sequence Idle

bit 3 **RCEN:** Receive Enable bit (in I²C Master mode only)

1 = Enables Receive mode for I²C

0 = Receive Idle

bit 2 **PEN:** Stop Condition Enable bit (in I²C Master mode only)

SCKx Release Control:

1 = Initiate Stop condition on SDAx and SCLx pins. Automatically cleared by hardware.

0 = Stop condition Idle

bit 1 **RSEN:** Repeated Start Condition Enabled bit (in I²C Master mode only)

1 = Initiate Repeated Start condition on SDAx and SCLx pins. Automatically cleared by hardware.

0 = Repeated Start condition Idle

bit 0 SEN: Start Condition Enable/Stretch Enable bit

In Master mode:

1 = Initiate Start condition on SDAx and SCLx pins. Automatically cleared by hardware.

0 = Start condition Idle

In Slave mode:

1 = Clock stretching is enabled for both slave transmit and slave receive (stretch enabled)

0 = Clock stretching is disabled

Note 1: For bits ACKEN, RCEN, PEN, RSEN, SEN: If the I²C module is not in the Idle mode, this bit may not be set (no spooling) and the SSPxBUF may not be written (or writes to the SSPxBUF are disabled).

REGISTER 25-4: SSPxCON3: SSPx CONTROL REGISTER 3

R-0/0	R/W-0/0						
ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7 ACKTIM: Acknowledge Time Status bit (I²C mode only)⁽³⁾

1 = Indicates the I²C bus is in an Acknowledge sequence, set on 8TH falling edge of SCLx clock

0 = Not an Acknowledge sequence, cleared on 9TH rising edge of SCLx clock

bit 6 **PCIE**: Stop Condition Interrupt Enable bit (I²C mode only)

1 = Enable interrupt on detection of Stop condition

0 = Stop detection interrupts are disabled⁽²⁾

bit 5 SCIE: Start Condition Interrupt Enable bit (I²C mode only)

1 = Enable interrupt on detection of Start or Restart conditions

0 = Start detection interrupts are disabled⁽²⁾

bit 4 **BOEN:** Buffer Overwrite Enable bit

In SPI Slave mode: (1)

1 = SSPxBUF updates every time that a new data byte is shifted in ignoring the BF bit

0 = If new byte is received with BF bit of the SSPxSTAT register already set, SSPOV bit of the SSPxCON1 register is set, and the buffer is not updated

In I²C Master mode and SPI Master mode:

This bit is ignored.

In I²C Slave mode:

- 1 = SSPxBUF is updated and \overline{ACK} is generated for a received address/data byte, ignoring the state of the SSPOV bit only if the BF bit = 0.
- 0 = SSPxBUF is only updated when SSPOV is clear
- bit 3 SDAHT: SDAx Hold Time Selection bit (I²C mode only)
 - 1 = Minimum of 300 ns hold time on SDAx after the falling edge of SCLx
 - 0 = Minimum of 100 ns hold time on SDAx after the falling edge of SCLx
- bit 2 SBCDE: Slave Mode Bus Collision Detect Enable bit (I²C Slave mode only)

If on the rising edge of SCLx, SDAx is sampled low when the module is outputting a high state, the BCLxIF bit of the PIR2 register is set, and bus goes Idle

- 1 = Enable slave bus collision interrupts
- 0 = Slave bus collision interrupts are disabled
- bit 1 AHEN: Address Hold Enable bit (I²C Slave mode only)
 - 1 = Following the 8th falling edge of SCLx for a matching received address byte; CKP bit of the SSPxCON1 register will be cleared and the SCLx will be held low.
 - 0 = Address holding is disabled
- bit 0 **DHEN:** Data Hold Enable bit (I²C Slave mode only)
 - 1 = Following the 8th falling edge of SCLx for a received data byte; slave hardware clears the CKP bit of the SSPxCON1 register and SCLx is held low.
 - 0 = Data holding is disabled
- **Note 1:** For daisy-chained SPI operation; allows the user to ignore all but the last received byte. SSPOV is still set when a new byte is received and BF = 1, but hardware continues to write the most recent byte to SSPxBUF.
 - 2: This bit has no effect in Slave modes that Start and Stop condition detection is explicitly listed as enabled.
 - 3: The ACKTIM Status bit is only active when the AHEN bit or DHEN bit is set.

REGISTER 25-5: SSPMSK: SSPx MASK REGISTER

| R/W-1/1 |
|---------|---------|---------|---------|---------|---------|---------|---------|
| | | | MSK | <7:0> | | | |
| bit 7 | | | | | | | bit 0 |

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

bit 7-1 **MSK<7:1>:** Mask bits

1 = The received address bit n is compared to SSPxADD<n> to detect I²C address match

0 = The received address bit n is not used to detect I²C address match

bit 0 MSK<0>: Mask bit for I²C Slave mode, 10-bit Address

 I^2C Slave mode, 10-bit address (SSPM<3:0> = 0111 or 1111):

1 = The received address bit 0 is compared to SSPxADD<0> to detect I²C address match

0 = The received address bit 0 is not used to detect I²C address match

I²C Slave mode, 7-bit address, the bit is ignored

REGISTER 25-6: SSPxADD: MSSPx ADDRESS AND BAUD RATE REGISTER (I²C MODE)

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

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

Master mode:

bit 7-0 ADD<7:0>: Baud Rate Clock Divider bits

SCLx pin clock period = ((ADD<7:0> + 1) *4)/Fosc

10-Bit Slave mode — Most Significant Address byte:

bit 7-3 Not used: Unused for Most Significant Address byte. Bit state of this register is a "don't care". Bit

pattern sent by master is fixed by I²C specification and must be equal to '11110'. However, those bits

are compared by hardware and are not affected by the value in this register.

bit 2-1 ADD<2:1>: Two Most Significant bits of 10-bit address

bit 0 **Not used:** Unused in this mode. Bit state is a "don't care".

10-Bit Slave mode — Least Significant Address byte:

bit 7-0 ADD<7:0>: Eight Least Significant bits of 10-bit address

7-Bit Slave mode:

bit 7-1 **ADD<7:1>:** 7-bit address

bit 0 **Not used:** Unused in this mode. Bit state is a "don't care".

26.0 ENHANCED UNIVERSAL SYNCHRONOUS ASYNCHRONOUS RECEIVER TRANSMITTER (EUSART)

The Enhanced Universal Synchronous Asynchronous Receiver Transmitter (EUSART) module is a serial I/O communications peripheral. It contains all the clock generators, shift registers and data buffers necessary to perform an input or output serial data transfer independent of device program execution. The EUSART, also known as a Serial Communications Interface (SCI), can be configured as a full-duplex asynchronous system or half-duplex synchronous system. Full-Duplex mode is useful communications with peripheral systems, such as CRT terminals and personal computers. Half-Duplex Synchronous mode is intended for communications with peripheral devices, such as A/D or D/A integrated circuits, serial EEPROMs or other microcontrollers. These devices typically do not have internal clocks for baud rate generation and require the external clock signal provided by a master synchronous device.

The EUSART module includes the following capabilities:

- Full-duplex asynchronous transmit and receive
- Two-character input buffer
- · One-character output buffer
- · Programmable 8-bit or 9-bit character length
- · Address detection in 9-bit mode
- · Input buffer overrun error detection
- Received character framing error detection
- · Half-duplex synchronous master
- · Half-duplex synchronous slave
- Programmable clock polarity in synchronous modes
- · Sleep operation

The EUSART module implements the following additional features, making it ideally suited for use in Local Interconnect Network (LIN) bus systems:

- · Automatic detection and calibration of the baud rate
- · Wake-up on Break reception
- · 13-bit Break character transmit

Block diagrams of the EUSART transmitter and receiver are shown in Figure 26-1 and Figure 26-2.

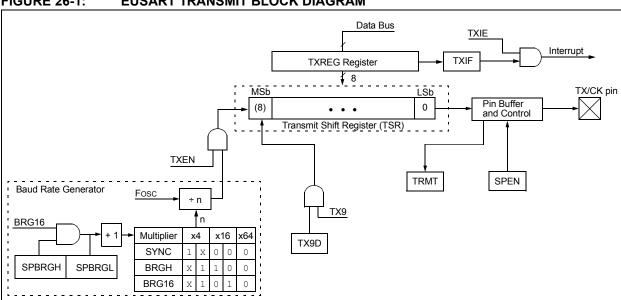


FIGURE 26-1: EUSART TRANSMIT BLOCK DIAGRAM

SPEN CREN **OERR** RCIDL RX/DT pin MSb RSR Register LSb Pin Buffer and Control Data Recovery Stop (8) START Baud Rate Generator Fosc RX9 ÷ n BRG16 Multiplier x16 x64 SYNC 1 X 0 0 0 FIFO **SPBRGH** SPBRGL **BRGH** 0 FERR RX9D RCREG Register BRG16 Data Bus Interrupt RCIE

FIGURE 26-2: EUSART RECEIVE BLOCK DIAGRAM

The operation of the EUSART module is controlled through three registers:

- Transmit Status and Control (TXSTA)
- · Receive Status and Control (RCSTA)
- Baud Rate Control (BAUDCON)

These registers are detailed in Register 26-1, Register 26-2 and Register 26-3, respectively.

When the receiver or transmitter section is not enabled then the corresponding RX or TX pin may be used for general purpose input and output.

26.1 EUSART Asynchronous Mode

The EUSART transmits and receives data using the standard non-return-to-zero (NRZ) format. NRZ is implemented with two levels: a VOH mark state which represents a '1' data bit, and a Vol space state which represents a '0' data bit. NRZ refers to the fact that consecutively transmitted data bits of the same value stay at the output level of that bit without returning to a neutral level between each bit transmission. An NRZ transmission port idles in the mark state. Each character transmission consists of one Start bit followed by eight or nine data bits and is always terminated by one or more Stop bits. The Start bit is always a space and the Stop bits are always marks. The most common data format is eight bits. Each transmitted bit persists for a period of 1/(Baud Rate). An on-chip dedicated 8-bit/16-bit Baud Rate Generator is used to derive standard baud rate frequencies from the system oscillator. See Table 26-5 for examples of baud rate configurations.

The EUSART transmits and receives the LSb first. The EUSART's transmitter and receiver are functionally independent, but share the same data format and baud rate. Parity is not supported by the hardware, but can be implemented in software and stored as the ninth data bit.

26.1.1 EUSART ASYNCHRONOUS TRANSMITTER

The EUSART transmitter block diagram is shown in Figure 26-1. The heart of the transmitter is the serial Transmit Shift Register (TSR), which is not directly accessible by software. The TSR obtains its data from the transmit buffer, which is the TXREG register.

26.1.1.1 Enabling the Transmitter

The EUSART transmitter is enabled for asynchronous operations by configuring the following three control bits:

- TXEN = 1
- SYNC = 0
- SPEN = 1

All other EUSART control bits are assumed to be in their default state.

Setting the TXEN bit of the TXSTA register enables the transmitter circuitry of the EUSART. Clearing the SYNC bit of the TXSTA register configures the EUSART for asynchronous operation. Setting the SPEN bit of the RCSTA register enables the EUSART and automatically configures the TX/CK I/O pin as an output. If the TX/CK pin is shared with an analog peripheral, the analog I/O function must be disabled by clearing the corresponding ANSEL bit.

Note 1: The TXIF Transmitter Interrupt flag is set when the TXEN enable bit is set.

26.1.1.2 Transmitting Data

A transmission is initiated by writing a character to the TXREG register. If this is the first character, or the previous character has been completely flushed from the TSR, the data in the TXREG is immediately transferred to the TSR register. If the TSR still contains all or part of a previous character, the new character data is held in the TXREG until the Stop bit of the previous character has been transmitted. The pending character in the TXREG is then transferred to the TSR in one Tcy immediately following the Stop bit transmission. The transmission of the Start bit, data bits and Stop bit sequence commences immediately following the transfer of the data to the TSR from the TXREG.

26.1.1.3 Transmit Data Polarity

The polarity of the transmit data can be controlled with the SCKP bit of the BAUDCON register. The default state of this bit is '0' which selects high true transmit idle and data bits. Setting the SCKP bit to '1' will invert the transmit data resulting in low true idle and data bits. The SCKP bit controls transmit data polarity in Asynchronous mode only. In Synchronous mode, the SCKP bit has a different function. See Section 26.4.1.2 "Clock Polarity".

26.1.1.4 Transmit Interrupt Flag

The TXIF interrupt flag bit of the PIR1 register is set whenever the EUSART transmitter is enabled and no character is being held for transmission in the TXREG. In other words, the TXIF bit is only clear when the TSR is busy with a character and a new character has been queued for transmission in the TXREG. The TXIF flag bit is not cleared immediately upon writing TXREG. TXIF becomes valid in the second instruction cycle following the write execution. Polling TXIF immediately following the TXREG write will return invalid results. The TXIF bit is read-only, it cannot be set or cleared by software.

The TXIF interrupt can be enabled by setting the TXIE interrupt enable bit of the PIE1 register. However, the TXIF flag bit will be set whenever the TXREG is empty, regardless of the state of TXIE enable bit.

To use interrupts when transmitting data, set the TXIE bit only when there is more data to send. Clear the TXIE interrupt enable bit upon writing the last character of the transmission to the TXREG.

26.1.1.5 TSR Status

The TRMT bit of the TXSTA register indicates the status of the TSR register. This is a read-only bit. The TRMT bit is set when the TSR register is empty and is cleared when a character is transferred to the TSR register from the TXREG. The TRMT bit remains clear until all bits have been shifted out of the TSR register. No interrupt logic is tied to this bit, so the user has to poll this bit to determine the TSR status.

Note: The TSR register is not mapped in data memory, so it is not available to the user.

26.1.1.6 Transmitting 9-bit Characters

The EUSART supports 9-bit character transmissions. When the TX9 bit of the TXSTA register is set, the EUSART will shift nine bits out for each character transmitted. The TX9D bit of the TXSTA register is the ninth, and Most Significant, data bit. When transmitting 9-bit data, the TX9D data bit must be written before writing the eight Least Significant bits into the TXREG. All nine bits of data will be transferred to the TSR shift register immediately after the TXREG is written.

A special 9-bit Address mode is available for use with multiple receivers. See **Section 26.1.2.7 "Address Detection"** for more information on the address mode.

26.1.1.7 Asynchronous Transmission Setup:

- Initialize the SPBRGH, SPBRGL register pair and the BRGH and BRG16 bits to achieve the desired baud rate (see Section 26.3 "EUSART Baud Rate Generator (BRG)".
- Enable the asynchronous serial port by clearing the SYNC bit and setting the SPEN bit.
- If 9-bit transmission is desired, set the TX9 control bit. A set ninth data bit will indicate that the eight Least Significant data bits are an address when the receiver is set for address detection.
- 4. Set SCKP bit if inverted transmit is desired.
- Enable the transmission by setting the TXEN control bit. This will cause the TXIF interrupt bit to be set.
- If interrupts are desired, set the TXIE interrupt enable bit of the PIE1 register. An interrupt will occur immediately provided that the GIE and PEIE bits of the INTCON register are also set.
- 7. If 9-bit transmission is selected, the ninth bit should be loaded into the TX9D data bit.
- Load 8-bit data into the TXREG register. This will start the transmission.



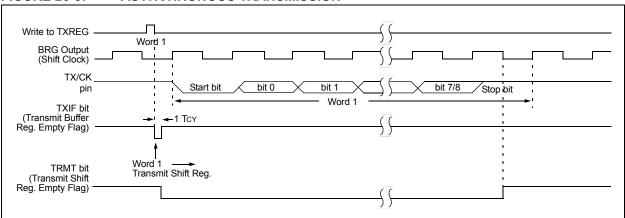


FIGURE 26-4: ASYNCHRONOUS TRANSMISSION (BACK-TO-BACK)

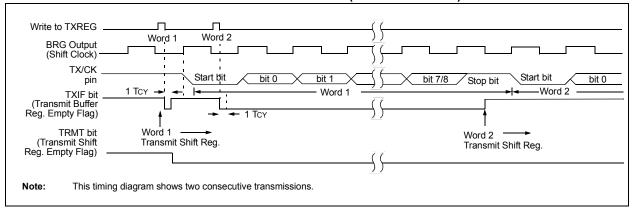


TABLE 26-1: SUMMARY OF REGISTERS ASSOCIATED WITH ASYNCHRONOUS TRANSMISSION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
APFCON0	RXDTSEL	SDO1SEL ⁽²⁾	SS1SEL ⁽²⁾	_	T1GSEL	TXCKSEL	_	_	121
BAUDCON	ABDOVF	RCIDL	-	SCKP	BRG16	_	WUE	ABDEN	304
INLVLA ⁽³⁾	_	_	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	127
INLVLB ⁽¹⁾	INLVLB7	INLVLB6	INLVLB5	INLVLB4	_	_	ı	_	132
INLVLC	INLVLC7 ⁽¹⁾	INLVLC6 ⁽¹⁾	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	138
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	90
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	91
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	95
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	303
SPBRGL				SPBRG	G<7:0>				305*
SPBRGH				SPBRG	<15:8>				305*
TRISA	_	-	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	125
TRISB ⁽¹⁾	TRISB7	TRISB6	TRISB5	TRISB4	_	_	_	_	131
TRISC	TRISC7 ⁽¹⁾	TRISC6 ⁽¹⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	136
TXREG	EUSART Tra	nsmit Data Re	gister						295
TXSTA	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	302

Legend: — Unimplemented location, read as '0'. Shaded cells are not used for asynchronous transmission.

* Page provides register information.

Note 1: PIC16(L)F1829 only. 2: PIC16(L)F1825 only.

3: Unshaded cells apply to PIC16(L)F1825 only.

26.1.2 EUSART ASYNCHRONOUS RECEIVER

The Asynchronous mode is typically used in RS-232 systems. The receiver block diagram is shown in Figure 26-2. The data is received on the RX/DT pin and drives the data recovery block. The data recovery block is actually a high-speed shifter operating at 16 times the baud rate, whereas the serial Receive Shift Register (RSR) operates at the bit rate. When all eight or nine bits of the character have been shifted in, they are immediately transferred to a two character First-In-First-Out (FIFO) memory. The FIFO buffering allows reception of two complete characters and the start of a third character before software must start servicing the EUSART receiver. The FIFO and RSR registers are not directly accessible by software. Access to the received data is via the RCREG register.

26.1.2.1 Enabling the Receiver

The EUSART receiver is enabled for asynchronous operation by configuring the following three control bits:

- CREN = 1
- SYNC = 0
- SPEN = 1

All other EUSART control bits are assumed to be in their default state.

Setting the CREN bit of the RCSTA register enables the receiver circuitry of the EUSART. Clearing the SYNC bit of the TXSTA register configures the EUSART for asynchronous operation. Setting the SPEN bit of the RCSTA register enables the EUSART. The programmer must set the corresponding TRIS bit to configure the RX/DT I/O pin as an input.

Note 1: If the RX/DT function is on an analog pin, the corresponding ANSEL bit must be cleared for the receiver to function.

26.1.2.2 Receiving Data

The receiver data recovery circuit initiates character reception on the falling edge of the first bit. The first bit, also known as the Start bit, is always a zero. The data recovery circuit counts one-half bit time to the center of the Start bit and verifies that the bit is still a zero. If it is not a zero then the data recovery circuit aborts character reception, without generating an error, and resumes looking for the falling edge of the Start bit. If the Start bit zero verification succeeds then the data recovery circuit counts a full bit time to the center of the next bit. The bit is then sampled by a majority detect circuit and the resulting '0' or '1' is shifted into the RSR. This repeats until all data bits have been sampled and shifted into the RSR. One final bit time is measured and the level sampled. This is the Stop bit, which is always a '1'. If the data recovery circuit samples a '0' in the Stop bit position then a framing error is set for this character, otherwise the framing error is cleared for this character. See Section 26.1.2.4 "Receive Framing **Error**"" for more information on framing errors.

Immediately after all data bits and the Stop bit have been received, the character in the RSR is transferred to the EUSART receive FIFO and the RCIF interrupt flag bit of the PIR1 register is set. The top character in the FIFO is transferred out of the FIFO by reading the RCREG register.

Note: If the receive FIFO is overrun, no additional characters will be received until the overrun condition is cleared. See Section 26.1.2.5
"Receive Overrun Error" for more information on overrun errors.

26.1.2.3 Receive Interrupts

The RCIF interrupt flag bit of the PIR1 register is set whenever the EUSART receiver is enabled and there is an unread character in the receive FIFO. The RCIF interrupt flag bit is read-only, it cannot be set or cleared by software.

RCIF interrupts are enabled by setting all of the following bits:

- RCIE interrupt enable bit of the PIE1 register
- PEIE Peripheral Interrupt Enable bit of the INTCON register
- GIE Global Interrupt Enable bit of the INTCON register

The RCIF interrupt flag bit will be set when there is an unread character in the FIFO, regardless of the state of interrupt enable bits.

26.1.2.4 Receive Framing Error

Each character in the receive FIFO buffer has a corresponding framing error Status bit. A framing error indicates that a Stop bit was not seen at the expected time. The framing error status is accessed via the FERR bit of the RCSTA register. The FERR bit represents the status of the top unread character in the receive FIFO. Therefore, the FERR bit must be read before reading the RCREG.

The FERR bit is read-only and only applies to the top unread character in the receive FIFO. A framing error (FERR = 1) does not preclude reception of additional characters. It is not necessary to clear the FERR bit. Reading the next character from the FIFO buffer will advance the FIFO to the next character and the next corresponding framing error.

The FERR bit can be forced clear by clearing the SPEN bit of the RCSTA register which resets the EUSART. Clearing the CREN bit of the RCSTA register does not affect the FERR bit. A framing error by itself does not generate an interrupt.

Note:

If all receive characters in the receive FIFO have framing errors, repeated reads of the RCREG will not clear the FERR bit.

26.1.2.5 Receive Overrun Error

The receive FIFO buffer can hold two characters. An overrun error will be generated if a third character, in its entirety, is received before the FIFO is accessed. When this happens the OERR bit of the RCSTA register is set. The characters already in the FIFO buffer can be read but no additional characters will be received until the error is cleared. The error must be cleared by either clearing the CREN bit of the RCSTA register or by resetting the EUSART by clearing the SPEN bit of the RCSTA register.

26.1.2.6 Receiving 9-bit Characters

The EUSART supports 9-bit character reception. When the RX9 bit of the RCSTA register is set the EUSART will shift nine bits into the RSR for each character received. The RX9D bit of the RCSTA register is the ninth and Most Significant data bit of the top unread character in the receive FIFO. When reading 9-bit data from the receive FIFO buffer, the RX9D data bit must be read before reading the eight Least Significant bits from the RCREG.

26.1.2.7 Address Detection

A special Address Detection mode is available for use when multiple receivers share the same transmission line, such as in RS-485 systems. Address detection is enabled by setting the ADDEN bit of the RCSTA register.

Address detection requires 9-bit character reception. When address detection is enabled, only characters with the ninth data bit set will be transferred to the receive FIFO buffer, thereby setting the RCIF interrupt bit. All other characters will be ignored.

Upon receiving an address character, user software determines if the address matches its own. Upon address match, user software must disable address detection by clearing the ADDEN bit before the next Stop bit occurs. When user software detects the end of the message, determined by the message protocol used, software places the receiver back into the Address Detection mode by setting the ADDEN bit.

26.1.2.8 Asynchronous Reception Setup:

- Initialize the SPBRGH, SPBRGL register pair and the BRGH and BRG16 bits to achieve the desired baud rate (see Section 26.3 "EUSART Baud Rate Generator (BRG)").
- Clear the ANSEL bit for the RX pin (if applicable).
- Enable the serial port by setting the SPEN bit.
 The SYNC bit must be clear for asynchronous operation.
- If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
- 5. If 9-bit reception is desired, set the RX9 bit.
- 6. Enable reception by setting the CREN bit.
- The RCIF interrupt flag bit will be set when a character is transferred from the RSR to the receive buffer. An interrupt will be generated if the RCIE interrupt enable bit was also set.
- Read the RCSTA register to get the error flags and, if 9-bit data reception is enabled, the ninth data bit.
- Get the received eight Least Significant data bits from the receive buffer by reading the RCREG register.
- If an overrun occurred, clear the OERR flag by clearing the CREN receiver enable bit.

26.1.2.9 9-bit Address Detection Mode Setup

This mode would typically be used in RS-485 systems. To set up an Asynchronous Reception with Address Detect Enable:

- Initialize the SPBRGH, SPBRGL register pair and the BRGH and BRG16 bits to achieve the desired baud rate (see Section 26.3 "EUSART Baud Rate Generator (BRG)").
- Clear the ANSEL bit for the RX pin (if applicable).
- Enable the serial port by setting the SPEN bit.
 The SYNC bit must be clear for asynchronous operation.
- If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
- 5. Enable 9-bit reception by setting the RX9 bit.
- Enable address detection by setting the ADDEN bit.
- 7. Enable reception by setting the CREN bit.
- The RCIF interrupt flag bit will be set when a character with the ninth bit set is transferred from the RSR to the receive buffer. An interrupt will be generated if the RCIE interrupt enable bit was also set.
- Read the RCSTA register to get the error flags. The ninth data bit will always be set.
- Get the received eight Least Significant data bits from the receive buffer by reading the RCREG register. Software determines if this is the device's address.
- 11. If an overrun occurred, clear the OERR flag by clearing the CREN receiver enable bit.
- 12. If the device has been addressed, clear the ADDEN bit to allow all received data into the receive buffer and generate interrupts.

FIGURE 26-5: ASYNCHRONOUS RECEPTION

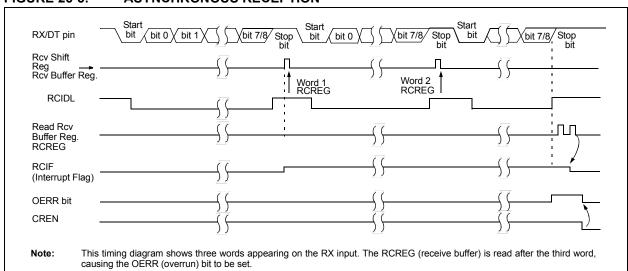


TABLE 26-2: SUMMARY OF REGISTERS ASSOCIATED WITH ASYNCHRONOUS RECEPTION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
APFCON0	RXDTSEL	SDO1SEL ⁽²⁾	SS1SEL ⁽²⁾	_	T1GSEL	TXCKSEL	_	_	121
BAUDCON	ABDOVF	RCIDL	-	SCKP	BRG16	_	WUE	ABDEN	304
INLVLA ⁽³⁾	_	_	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	127
INLVLB ⁽¹⁾	INLVLB7	INLVLB6	INLVLB5	INLVLB4	_	_	ı	1	132
INLVLC	INLVLC7 ⁽¹⁾	INLVLC6 ⁽¹⁾	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	138
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	90
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	91
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	95
RCREG	EUSART Red	ceive Data Reg	jister						298*
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	303
SPBRGL				SPBRG	<7:0>				305*
SPBRGH				SPBRG	<15:8>				305*
TRISA	_	_	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	125
TRISB ⁽¹⁾	TRISB7	TRISB6	TRISB5	TRISB4	_	_	_	_	131
TRISC	TRISC7 ⁽¹⁾	TRISC6 ⁽¹⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	136
TXSTA	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	302

Legend: — Unimplemented location, read as '0'. Shaded cells are not used for asynchronous reception.

* Page provides register information.

Note 1: PIC16(L)F1829 only. 2: PIC16(L)F1825 only.

3: Unshaded cells apply to PIC16(L)F1825 only.

26.2 Clock Accuracy with Asynchronous Operation

The factory calibrates the internal oscillator block output (INTOSC). However, the INTOSC frequency may drift as VDD or temperature changes, and this directly affects the asynchronous baud rate. Two methods may be used to adjust the baud rate clock, but both require a reference clock source of some kind.

The first (preferred) method uses the OSCTUNE register to adjust the INTOSC output. Adjusting the value in the OSCTUNE register allows for fine resolution changes to the system clock source. See Section 5.2.2 "Internal Clock Sources" for more information.

The other method adjusts the value in the Baud Rate Generator. This can be done automatically with the Auto-Baud Detect feature (see Section 26.3.1 "Auto-Baud Detect"). There may not be fine enough resolution when adjusting the Baud Rate Generator to compensate for a gradual change in the peripheral clock frequency.

REGISTER 26-1: TXSTA: TRANSMIT STATUS AND CONTROL REGISTER

R/W-/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R-1/1	R/W-0/0
CSRC	TX9	TXEN ⁽¹⁾	SYNC	SENDB	BRGH	TRMT	TX9D
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7 CSRC: Clock Source Select bit

Asynchronous mode:

Don't care

Synchronous mode:

1 = Master mode (clock generated internally from BRG)

0 = Slave mode (clock from external source)

bit 6 **TX9:** 9-bit Transmit Enable bit

1 = Selects 9-bit transmission

= Selects 8-bit transmission

bit 5 **TXEN:** Transmit Enable bit⁽¹⁾

1 = Transmit enabled

0 = Transmit disabled

bit 4 SYNC: EUSART Mode Select bit

1 = Synchronous mode0 = Asynchronous mode

bit 3 **SENDB:** Send Break Character bit

Asynchronous mode:

1 = Send Sync Break on next transmission (cleared by hardware upon completion)

0 = Sync Break transmission completed

Synchronous mode:

Don't care

bit 2 BRGH: High Baud Rate Select bit

Asynchronous mode: 1 = High speed 0 = Low speed Synchronous mode:

Unused in this mode
bit 1 TRMT: Transmit Shift Register Status bit

1 = TSR empty
0 = TSR full

bit 0 **TX9D:** Ninth bit of Transmit Data

Can be address/data bit or a parity bit.

Note 1: SREN/CREN overrides TXEN in Sync mode.

REGISTER 26-2: RCSTA: RECEIVE STATUS AND CONTROL REGISTER (1)

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R-0/0	R-0/0	R-x/x
SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

bit 7 SPEN: Serial Port Enable bit

1 = Serial port enabled (configures RX/DT and TX/CK pins as serial port pins)

0 = Serial port disabled (held in Reset)

bit 6 **RX9:** 9-bit Receive Enable bit

1 = Selects 9-bit reception0 = Selects 8-bit reception

bit 5 SREN: Single Receive Enable bit

Asynchronous mode:

Don't care

Synchronous mode – Master:

1 = Enables single receive

0 = Disables single receive

This bit is cleared after reception is complete.

Synchronous mode - Slave

Don't care

bit 4 CREN: Continuous Receive Enable bit

Asynchronous mode:

1 = Enables receiver

0 = Disables receiver

Synchronous mode:

1 = Enables continuous receive until enable bit CREN is cleared (CREN overrides SREN)

0 = Disables continuous receive

bit 3 ADDEN: Address Detect Enable bit

Asynchronous mode 9-bit (RX9 = 1):

1 = Enables address detection, enable interrupt and load the receive buffer when RSR<8> is set

0 = Disables address detection, all bytes are received and ninth bit can be used as parity bit

Asynchronous mode 8-bit (RX9 = 0):

Don't care

bit 2 **FERR:** Framing Error bit

1 = Framing error (can be updated by reading RCREG register and receive next valid byte)

0 = No framing error

bit 1 **OERR:** Overrun Error bit

1 = Overrun error (can be cleared by clearing bit CREN)

0 = No overrun error

bit 0 **RX9D:** Ninth bit of Received Data

This can be address/data bit or a parity bit and must be calculated by user firmware.

REGISTER 26-3: BAUDCON: BAUD RATE CONTROL REGISTER

R-0/0	R-1/1	U-0	R/W-0/0	R/W-0/0	U-0	R/W-0/0	R/W-0/0
ABDOVF	RCIDL	_	SCKP	BRG16	_	WUE	ABDEN
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets '1' = Bit is set '0' = Bit is cleared

bit 7 ABDOVF: Auto-Baud Detect Overflow bit

Asynchronous mode:

1 = Auto-baud timer overflowed0 = Auto-baud timer did not overflow

Synchronous mode:

Don't care

bit 6 RCIDL: Receive Idle Flag bit

<u>Asynchronous mode</u>: 1 = Receiver is Idle

0 = Start bit has been received and the receiver is receiving

Synchronous mode:

Don't care

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

bit 4 SCKP: Synchronous Clock Polarity Select bit

Asynchronous mode:

1 = Transmit inverted data to the TX/CK pin 0 = Transmit non-inverted data to the TX/CK pin

Synchronous mode:

1 = Data is clocked on rising edge of the clock0 = Data is clocked on falling edge of the clock

bit 3 BRG16: 16-bit Baud Rate Generator bit

1 = 16-bit Baud Rate Generator is used0 = 8-bit Baud Rate Generator is used

bit 2 **Unimplemented:** Read as '0' bit 1 **WUE:** Wake-up Enable bit

Asynchronous mode:

1 = Receiver is waiting for a falling edge. No character will be received, byte RCIF will be set. WUE will automatically clear after RCIF is set.

0 = Receiver is operating normally

Synchronous mode:

Don't care

bit 0 ABDEN: Auto-Baud Detect Enable bit

Asynchronous mode:

1 = Auto-Baud Detect mode is enabled (clears when auto-baud is complete)

0 = Auto-Baud Detect mode is disabled

Synchronous mode:

Don't care

26.3 EUSART Baud Rate Generator (BRG)

The Baud Rate Generator (BRG) is an 8-bit or 16-bit timer that is dedicated to the support of both the asynchronous and synchronous EUSART operation. By default, the BRG operates in 8-bit mode. Setting the BRG16 bit of the BAUDCON register selects 16-bit mode.

The SPBRGH, SPBRGL register pair determines the period of the free running baud rate timer. In Asynchronous mode the multiplier of the baud rate period is determined by both the BRGH bit of the TXSTA register and the BRG16 bit of the BAUDCON register. In Synchronous mode, the BRGH bit is ignored.

Table 26-3 contains the formulas for determining the baud rate. Example 26-1 provides a sample calculation for determining the baud rate and baud rate error.

Typical baud rates and error values for various asynchronous modes have been computed for your convenience and are shown in Table 26-3. It may be advantageous to use the high baud rate (BRGH = 1), or the 16-bit BRG (BRG16 = 1) to reduce the baud rate error. The 16-bit BRG mode is used to achieve slow baud rates for fast oscillator frequencies.

Writing a new value to the SPBRGH, SPBRGL register pair causes the BRG timer to be reset (or cleared). This ensures that the BRG does not wait for a timer overflow before outputting the new baud rate.

If the system clock is changed during an active receive operation, a receive error or data loss may result. To avoid this problem, check the status of the RCIDL bit to make sure that the receive operation is Idle before changing the system clock.

EXAMPLE 26-1: CALCULATING BAUD RATE ERROR

For a device with Fosc of 16 MHz, desired baud rate of 9600, Asynchronous mode, 8-bit BRG:

Desired Baud Rate =
$$\frac{FOSC}{64([SPBRGH:SPBRGL] + 1)}$$

Solving for SPBRGH:SPBRGL:

Solving for SERGE.
$$X = \frac{FOSC}{\frac{Desired Baud Rate}{64}} - 1$$

$$= \frac{\frac{16000000}{9600}}{\frac{9600}{64}} - 1$$

$$= [25.042] = 25$$

$$Calculated Baud Rate = \frac{16000000}{64(25+1)}$$

$$= 9615$$

$$Error = \frac{Calc. Baud Rate - Desired Baud Rate}{Desired Baud Rate}$$

$$= \frac{(9615 - 9600)}{9600} = 0.16\%$$

TABLE 26-3: BAUD RATE FORMULAS

(Configuration Bi	ts	DDO/ELIOADT Marke	Baud Rate Formula
SYNC	BRG16	BRGH	BRG/EUSART Mode	Baud Rate Formula
0	0	0	8-bit/Asynchronous	Fosc/[64 (n+1)]
0	0	1	8-bit/Asynchronous	F(/40 (4))
0	1	0	16-bit/Asynchronous	Fosc/[16 (n+1)]
0	1	1	16-bit/Asynchronous	
1	0	Х	8-bit/Synchronous	Fosc/[4 (n+1)]
1	1	Х	16-bit/Synchronous	

Legend: x = Don't care, n = value of SPBRGH, SPBRGL register pair.

TABLE 26-4: SUMMARY OF REGISTERS ASSOCIATED WITH THE BAUD RATE GENERATOR

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
BAUDCON	ABDOVF	RCIDL	_	SCKP	BRG16	_	WUE	ABDEN	304
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	303
SPBRGL				SPBR	G<7:0>				305*
SPBRGH				SPBRG	G<15:8>				305*
TXSTA	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	302

Legend: — Unimplemented location, read as '0'. Shaded cells are not used for the Baud Rate Generator.

^{*} Page provides register information.

TABLE 26-5: BAUD RATES FOR ASYNCHRONOUS MODES

					SYNC	= 0, BRGH	I = 0, BRG	316 = 0				
BAUD	Fosc	= 32.00	0 MHz	Fosc	= 20.00	0 MHz	Fosc	= 18.43	2 MHz	Fosc	= 11.059	2 MHz
RATE	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)
300	_	_	_		_	_	_	_	_		_	_
1200	_	_	_	1221	1.73	255	1200	0.00	239	1200	0.00	143
2400	2404	0.16	207	2404	0.16	129	2400	0.00	119	2400	0.00	71
9600	9615	0.16	51	9470	-1.36	32	9600	0.00	29	9600	0.00	17
10417	10417	0.00	47	10417	0.00	29	10286	-1.26	27	10165	-2.42	16
19.2k	19.23k	0.16	25	19.53k	1.73	15	19.20k	0.00	14	19.20k	0.00	8
57.6k	55.55k	-3.55	3	_	_	_	57.60k	0.00	7	57.60k	0.00	2
115.2k	_	_	_	_	_	_	_	_	_	_	_	_

					SYNC	= 0, BRGH	I = 0, BRG	316 = 0				
BAUD	Fos	c = 8.000) MHz	Fos	c = 4.000) MHz	Fosc	= 3.686	4 MHz	Fos	c = 1.000) MHz
RATE	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)
300	_	_	_	300	0.16	207	300	0.00	191	300	0.16	51
1200	1202	0.16	103	1202	0.16	51	1200	0.00	47	1202	0.16	12
2400	2404	0.16	51	2404	0.16	25	2400	0.00	23	_	_	_
9600	9615	0.16	12	_	_	_	9600	0.00	5	_	_	_
10417	10417	0.00	11	10417	0.00	5	_	_	_	_	_	_
19.2k	_	_	_	_	_	_	19.20k	0.00	2	_	_	_
57.6k	_	_	_	_	_	_	57.60k	0.00	0	_	_	_
115.2k	_	_	_	_	_	_	1	_	_	_	_	_

					SYNC	C = 0, BRGH	l = 1, BRC	316 = 0				
BAUD	Fosc	= 32.00	0 MHz	Fosc	= 20.00	0 MHz	Fosc	= 18.43	2 MHz	Fosc	= 11.059	2 MHz
RATE	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)
300	_	_	_	_	_	_		_	_		_	_
1200	_	_	_	_	_	_	_	_	_	_	_	_
2400	_	_	_	_	_	_	_	_	_	_	_	_
9600	9615	0.16	207	9615	0.16	129	9600	0.00	119	9600	0.00	71
10417	10417	0.00	191	10417	0.00	119	10378	-0.37	110	10473	0.53	65
19.2k	19.23k	0.16	103	19.23k	0.16	64	19.20k	0.00	59	19.20k	0.00	35
57.6k	57.14k	-0.79	34	56.82k	-1.36	21	57.60k	0.00	19	57.60k	0.00	11
115.2k	117.64k	2.12	16	113.64k	-1.36	10	115.2k	0.00	9	115.2k	0.00	5

TABLE 26-5: BAUD RATES FOR ASYNCHRONOUS MODES (CONTINUED)

					SYNC	= 0, BRG	l = 1, BRO	316 = 0				
BAUD	Fos	c = 8.000) MHz	Fos	c = 4.000) MHz	Fosc	= 3.686	4 MHz	Fos	c = 1.000) MHz
RATE	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)
300	_	_	_	_	_	_		_	_	300	0.16	207
1200	_	_	_	1202	0.16	207	1200	0.00	191	1202	0.16	51
2400	2404	0.16	207	2404	0.16	103	2400	0.00	95	2404	0.16	25
9600	9615	0.16	51	9615	0.16	25	9600	0.00	23	_	_	_
10417	10417	0.00	47	10417	0.00	23	10473	0.53	21	10417	0.00	5
19.2k	19231	0.16	25	19.23k	0.16	12	19.2k	0.00	11	_	_	_
57.6k	55556	-3.55	8	_	_	_	57.60k	0.00	3	_	_	_
115.2k	_	_	_	_	_	_	115.2k	0.00	1	_	_	_

					SYNC	= 0, BRGH	H = 0, BRO	316 = 1				
BAUD	Fosc	= 32.00	0 MHz	Fosc	= 20.00	0 MHz	Fosc	= 18.43	2 MHz	Fosc	= 11.059	92 MHz
RATE	Actual Rate	% Error	SPBRG value (decimal)									
300	300.0	0.00	6666	300.0	-0.01	4166	300.0	0.00	3839	300.0	0.00	2303
1200	1200	-0.02	3332	1200	-0.03	1041	1200	0.00	959	1200	0.00	575
2400	2401	-0.04	832	2399	-0.03	520	2400	0.00	479	2400	0.00	287
9600	9615	0.16	207	9615	0.16	129	9600	0.00	119	9600	0.00	71
10417	10417	0.00	191	10417	0.00	119	10378	-0.37	110	10473	0.53	65
19.2k	19.23k	0.16	103	19.23k	0.16	64	19.20k	0.00	59	19.20k	0.00	35
57.6k	57.14k	-0.79	34	56.818	-1.36	21	57.60k	0.00	19	57.60k	0.00	11
115.2k	117.6k	2.12	16	113.636	-1.36	10	115.2k	0.00	9	115.2k	0.00	5

					SYNC	C = 0, BRGH	l = 0, BRG	316 = 1				
BAUD	Fos	c = 8.000) MHz	Fos	c = 4.000) MHz	Fosc	= 3.686	4 MHz	Fos	c = 1.000) MHz
RATE	Actual Rate	% Error	SPBRG value (decimal)									
300	299.9	-0.02	1666	300.1	0.04	832	300.0	0.00	767	300.5	0.16	207
1200	1199	-0.08	416	1202	0.16	207	1200	0.00	191	1202	0.16	51
2400	2404	0.16	207	2404	0.16	103	2400	0.00	95	2404	0.16	25
9600	9615	0.16	51	9615	0.16	25	9600	0.00	23	_	_	_
10417	10417	0.00	47	10417	0.00	23	10473	0.53	21	10417	0.00	5
19.2k	19.23k	0.16	25	19.23k	0.16	12	19.20k	0.00	11	_	_	_
57.6k	55556	-3.55	8	_	_	_	57.60k	0.00	3	_	_	_
115.2k			_	_	_	_	115.2k	0.00	1	_	_	_

TABLE 26-5: BAUD RATES FOR ASYNCHRONOUS MODES (CONTINUED)

				SYNC = 0	, BRGH	= 1, BRG16	= 1 or SY	'NC = 1,	BRG16 = 1			
BAUD	Fosc	= 32.00	0 MHz	Fosc	= 20.00	0 MHz	Fosc	= 18.43	2 MHz	Fosc	= 11.059	92 MHz
RATE	Actual Rate	% Error	SPBRG value (decimal)									
300	300.0	0.00	26666	300.0	0.00	16665	300.0	0.00	15359	300.0	0.00	9215
1200	1200	0.00	6666	1200	-0.01	4166	1200	0.00	3839	1200	0.00	2303
2400	2400	0.01	3332	2400	0.02	2082	2400	0.00	1919	2400	0.00	1151
9600	9604	0.04	832	9597	-0.03	520	9600	0.00	479	9600	0.00	287
10417	10417	0.00	767	10417	0.00	479	10425	0.08	441	10433	0.16	264
19.2k	19.18k	-0.08	416	19.23k	0.16	259	19.20k	0.00	239	19.20k	0.00	143
57.6k	57.55k	-0.08	138	57.47k	-0.22	86	57.60k	0.00	79	57.60k	0.00	47
115.2k	115.9k	0.64	68	116.3k	0.94	42	115.2k	0.00	39	115.2k	0.00	23

				SYNC = 0	, BRGH	= 1, BRG16	= 1 or SY	'NC = 1,	BRG16 = 1			
BAUD	Fos	c = 8.000	MHz	Fos	c = 4.000	MHz	Fosc	= 3.686	4 MHz	Fos	c = 1.000) MHz
RATE	Actual Rate	% Error	SPBRG value (decimal)									
300	300.0	0.00	6666	300.0	0.01	3332	300.0	0.00	3071	300.1	0.04	832
1200	1200	-0.02	1666	1200	0.04	832	1200	0.00	767	1202	0.16	207
2400	2401	0.04	832	2398	0.08	416	2400	0.00	383	2404	0.16	103
9600	9615	0.16	207	9615	0.16	103	9600	0.00	95	9615	0.16	25
10417	10417	0	191	10417	0.00	95	10473	0.53	87	10417	0.00	23
19.2k	19.23k	0.16	103	19.23k	0.16	51	19.20k	0.00	47	19.23k	0.16	12
57.6k	57.14k	-0.79	34	58.82k	2.12	16	57.60k	0.00	15	_	_	_
115.2k	117.6k	2.12	16	111.1k	-3.55	8	115.2k	0.00	7	_		_

26.3.1 AUTO-BAUD DETECT

The EUSART module supports automatic detection and calibration of the baud rate.

In the Auto-Baud Detect (ABD) mode, the clock to the BRG is reversed. Rather than the BRG clocking the incoming RX signal, the RX signal is timing the BRG. The Baud Rate Generator is used to time the period of a received 55h (ASCII "U") which is the Sync character for the LIN bus. The unique feature of this character is that it has five rising edges including the Stop bit edge.

Setting the ABDEN bit of the BAUDCON register starts the auto-baud calibration sequence (Figure 26-6). While the ABD sequence takes place, the EUSART state machine is held in Idle. On the first rising edge of the receive line, after the Start bit, the SPBRG begins counting up using the BRG counter clock as shown in Figure 26-6. The fifth rising edge will occur on the RX pin at the end of the eighth bit period. At that time, an accumulated value totaling the proper BRG period is left in the SPBRGH, SPBRGL register pair, the ABDEN bit is automatically cleared and the RCIF interrupt flag is set. The value in the RCREG needs to be read to clear the RCIF interrupt. RCREG content should be discarded. When calibrating for modes that do not use the SPBRGH register the user can verify that the SPBRGL register did not overflow by checking for 00h in the SPBRGH register.

The BRG auto-baud clock is determined by the BRG16 and BRGH bits as shown in Table 26-6. During ABD, both the SPBRGH and SPBRGL registers are used as a 16-bit counter, independent of the BRG16 bit setting. While calibrating the baud rate period, the SPBRGH

and SPBRGL registers are clocked at 1/8th the BRG base clock rate. The resulting byte measurement is the average bit time when clocked at full speed.

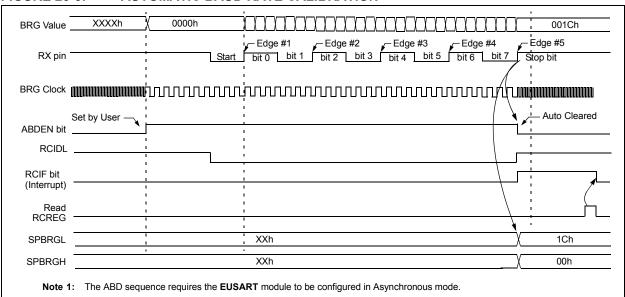
- Note 1: If the WUE bit is set with the ABDEN bit, auto-baud detection will occur on the byte following the Break character (see Section 26.3.3 "Auto-Wake-up on Break").
 - 2: It is up to the user to determine that the incoming character baud rate is within the range of the selected BRG clock source. Some combinations of oscillator frequency and EUSART baud rates are not possible.
 - 3: During the auto-baud process, the auto-baud counter starts counting at 1. Upon completion of the auto-baud sequence, to achieve maximum accuracy, subtract 1 from the SPBRGH:SPBRGL register pair.

TABLE 26-6: BRG COUNTER CLOCK RATES

BRG16	BRGH	BRG Base Clock	BRG ABD Clock
0	0	Fosc/64	Fosc/512
0	1	Fosc/16	Fosc/128
1	0	Fosc/16	Fosc/128
1	1	Fosc/4	Fosc/32

Note: During the ABD sequence, SPBRGL and SPBRGH registers are both used as a 16-bit counter, independent of BRG16 setting.

FIGURE 26-6: AUTOMATIC BAUD RATE CALIBRATION



26.3.2 AUTO-BAUD OVERFLOW

During the course of automatic baud detection, the ABDOVF bit of the BAUDCON register will be set if the baud rate counter overflows before the fifth rising edge is detected on the RX pin. The ABDOVF bit indicates that the counter has exceeded the maximum count that can fit in the 16 bits of the SPBRGH:SPBRGL register pair. After the ABDOVF has been set, the counter continues to count until the fifth rising edge is detected on the RX pin. Upon detecting the fifth RX edge, the hardware will set the RCIF interrupt flag and clear the ABDEN bit of the BAUDCON register. The RCIF flag can be subsequently cleared by reading the RCREG register. The ABDOVF flag of the BAUDCON register can be cleared by software directly.

To terminate the auto-baud process before the RCIF flag is set, clear the ABDEN bit then clear the ABDOVF bit of the BAUDCON register. The ABDOVF bit will remain set if the ABDEN bit is not cleared first.

26.3.3 AUTO-WAKE-UP ON BREAK

During Sleep mode, all clocks to the EUSART are suspended. Because of this, the Baud Rate Generator is inactive and a proper character reception cannot be performed. The Auto-Wake-up feature allows the controller to wake-up due to activity on the RX/DT line. This feature is available only in Asynchronous mode.

The Auto-Wake-up feature is enabled by setting the WUE bit of the BAUDCON register. Once set, the normal receive sequence on RX/DT is disabled, and the EUSART remains in an Idle state, monitoring for a wake-up event independent of the CPU mode. A wake-up event consists of a high-to-low transition on the RX/DT line. (This coincides with the start of a Sync Break or a wake-up signal character for the LIN protocol.)

The EUSART module generates an RCIF interrupt coincident with the wake-up event. The interrupt is generated synchronously to the Q clocks in normal CPU operating modes (Figure 26-7), and asynchronously if the device is in Sleep mode (Figure 26-8). The interrupt condition is cleared by reading the RCREG register.

The WUE bit is automatically cleared by the low-to-high transition on the RX line at the end of the Break. This signals to the user that the Break event is over. At this point, the EUSART module is in Idle mode waiting to receive the next character.

26.3.3.1 Special Considerations

Break Character

To avoid character errors or character fragments during a wake-up event, the wake-up character must be all zeros.

When the wake-up is enabled the function works independent of the low time on the data stream. If the WUE bit is set and a valid non-zero character is received, the low time from the Start bit to the first rising edge will be interpreted as the wake-up event. The remaining bits in the character will be received as a fragmented character and subsequent characters can result in framing or overrun errors.

Therefore, the initial character in the transmission must be all '0's. This must be 10 or more bit times, 13-bit times recommended for LIN bus, or any number of bit times for standard RS-232 devices.

Oscillator Start-up Time

Oscillator start-up time must be considered, especially in applications using oscillators with longer start-up intervals (i.e., LP, XT or HS/PLL mode). The Sync Break (or wake-up signal) character must be of sufficient length, and be followed by a sufficient interval, to allow enough time for the selected oscillator to start and provide proper initialization of the EUSART.

WUE Bit

The wake-up event causes a receive interrupt by setting the RCIF bit. The WUE bit is cleared in hardware by a rising edge on RX/DT. The interrupt condition is then cleared in software by reading the RCREG register and discarding its contents.

To ensure that no actual data is lost, check the RCIDL bit to verify that a receive operation is not in process before setting the WUE bit. If a receive operation is not occurring, the WUE bit may then be set just prior to entering the Sleep mode.

AUTO-WAKE-UP BIT (WUE) TIMING DURING NORMAL OPERATION FIGURE 26-7:

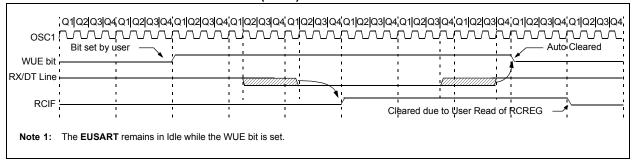
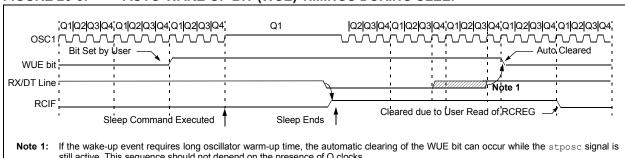


FIGURE 26-8: AUTO-WAKE-UP BIT (WUE) TIMINGS DURING SLEEP



still active. This sequence should not depend on the presence of Q clocks.

2: The EUSART remains in Idle while the WUE bit is set.

26.3.4 BREAK CHARACTER SEQUENCE

The EUSART module has the capability of sending the special Break character sequences that are required by the LIN bus standard. A Break character consists of a Start bit, followed by 12 '0' bits and a Stop bit.

To send a Break character, set the SENDB and TXEN bits of the TXSTA register. The Break character transmission is then initiated by a write to the TXREG. The value of data written to TXREG will be ignored and all '0's will be transmitted.

The SENDB bit is automatically reset by hardware after the corresponding Stop bit is sent. This allows the user to preload the transmit FIFO with the next transmit byte following the Break character (typically, the Sync character in the LIN specification).

The TRMT bit of the TXSTA register indicates when the transmit operation is active or Idle, just as it does during normal transmission. See Figure 26-9 for the timing of the Break character sequence.

26.3.4.1 Break and Sync Transmit Sequence

The following sequence will start a message frame header made up of a Break, followed by an auto-baud Sync byte. This sequence is typical of a LIN bus master.

- 1. Configure the EUSART for the desired mode.
- Set the TXEN and SENDB bits to enable the Break sequence.
- 3. Load the TXREG with a dummy character to initiate transmission (the value is ignored).
- 4. Write '55h' to TXREG to load the Sync character into the transmit FIFO buffer.
- After the Break has been sent, the SENDB bit is reset by hardware and the Sync character is then transmitted.

When the TXREG becomes empty, as indicated by the TXIF, the next data byte can be written to TXREG.

26.3.5 RECEIVING A BREAK CHARACTER

The Enhanced EUSART module can receive a Break character in two ways.

The first method to detect a Break character uses the FERR bit of the RCSTA register and the Received data as indicated by RCREG. The Baud Rate Generator is assumed to have been initialized to the expected baud rate.

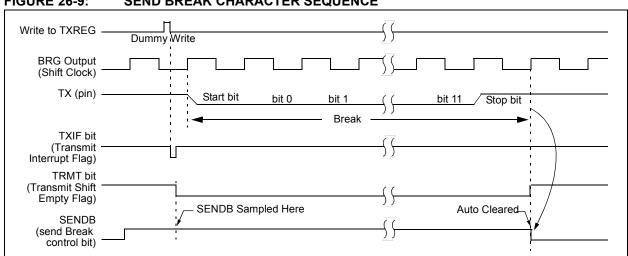
A Break character has been received when;

- · RCIF bit is set
- · FERR bit is set
- RCREG = 00h

The second method uses the Auto-Wake-up feature described in Section 26.3.3 "Auto-Wake-up on Break". By enabling this feature, the EUSART will sample the next two transitions on RX/DT, cause an RCIF interrupt, and receive the next data byte followed by another interrupt.

Note that following a Break character, the user will typically want to enable the Auto-Baud Detect feature. For both methods, the user can set the ABDEN bit of the BAUDCON register before placing the EUSART in Sleep mode.





26.4 EUSART Synchronous Mode

Synchronous serial communications are typically used in systems with a single master and one or more slaves. The master device contains the necessary circuitry for baud rate generation and supplies the clock for all devices in the system. Slave devices can take advantage of the master clock by eliminating the internal clock generation circuitry.

There are two signal lines in Synchronous mode: a bidirectional data line and a clock line. Slaves use the external clock supplied by the master to shift the serial data into and out of their respective receive and transmit shift registers. Since the data line is bidirectional, synchronous operation is half-duplex only. Half-duplex refers to the fact that master and slave devices can receive and transmit data but not both simultaneously. The EUSART can operate as either a master or slave device.

Start and Stop bits are not used in synchronous transmissions.

26.4.1 SYNCHRONOUS MASTER MODE

The following bits are used to configure the EUSART for Synchronous Master operation:

- SYNC = 1
- CSRC = 1
- SREN = 0 (for transmit); SREN = 1 (for receive)
- CREN = 0 (for transmit); CREN = 1 (for receive)
- SPFN = 1

Setting the SYNC bit of the TXSTA register configures the device for synchronous operation. Setting the CSRC bit of the TXSTA register configures the device as a master. Clearing the SREN and CREN bits of the RCSTA register ensures that the device is in the Transmit mode, otherwise the device will be configured to receive. Setting the SPEN bit of the RCSTA register enables the EUSART.

26.4.1.1 Master Clock

Synchronous data transfers use a separate clock line, which is synchronous with the data. A device configured as a master transmits the clock on the TX/CK line. The TX/CK pin output driver is automatically enabled when the EUSART is configured for synchronous transmit or receive operation. Serial data bits change on the leading edge to ensure they are valid at the trailing edge of each clock. One clock cycle is generated for each data bit. Only as many clock cycles are generated as there are data bits.

26.4.1.2 Clock Polarity

A clock polarity option is provided for Microwire compatibility. Clock polarity is selected with the SCKP bit of the BAUDCON register. Setting the SCKP bit sets the clock Idle state as high. When the SCKP bit is set, the data changes on the falling edge of each clock.

Clearing the SCKP bit sets the Idle state as low. When the SCKP bit is cleared, the data changes on the rising edge of each clock.

26.4.1.3 Synchronous Master Transmission

Data is transferred out of the device on the RX/DT pin. The RX/DT and TX/CK pin output drivers are automatically enabled when the EUSART is configured for synchronous master transmit operation.

A transmission is initiated by writing a character to the TXREG register. If the TSR still contains all or part of a previous character the new character data is held in the TXREG until the last bit of the previous character has been transmitted. If this is the first character, or the previous character has been completely flushed from the TSR, the data in the TXREG is immediately transferred to the TSR. The transmission of the character commences immediately following the transfer of the data to the TSR from the TXREG.

Each data bit changes on the leading edge of the master clock and remains valid until the subsequent leading clock edge.

Note: The TSR register is not mapped in data memory, so it is not available to the user.

26.4.1.4 Synchronous Master Transmission Setup:

- Initialize the SPBRGH, SPBRGL register pair and the BRGH and BRG16 bits to achieve the desired baud rate (see Section 26.3 "EUSART Baud Rate Generator (BRG)").
- Enable the synchronous master serial port by setting bits SYNC, SPEN and CSRC.
- Disable Receive mode by clearing bits SREN and CREN.
- 4. Enable Transmit mode by setting the TXEN bit.
- 5. If 9-bit transmission is desired, set the TX9 bit.
- If interrupts are desired, set the TXIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
- If 9-bit transmission is selected, the ninth bit should be loaded in the TX9D bit.
- Start transmission by loading data to the TXREG register.

FIGURE 26-10: SYNCHRONOUS TRANSMISSION

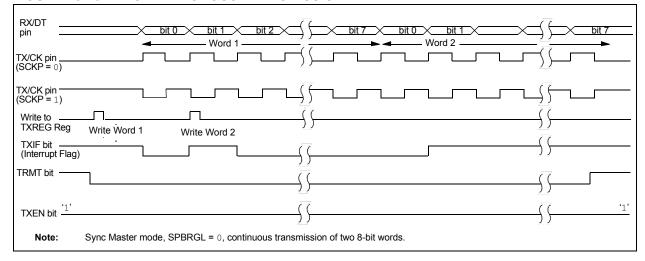


FIGURE 26-11: SYNCHRONOUS TRANSMISSION (THROUGH TXEN)

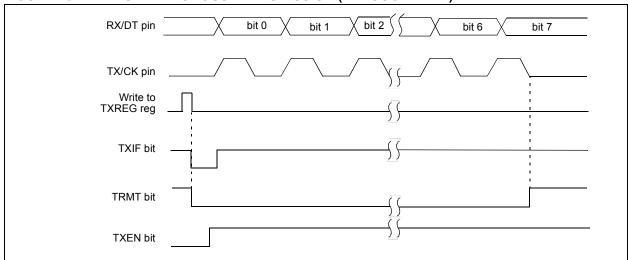


TABLE 26-7: SUMMARY OF REGISTERS ASSOCIATED WITH SYNCHRONOUS MASTER **TRANSMISSION**

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
APFCON0	RXDTSEL	SDO1SEL ⁽¹⁾	SS1SEL ⁽¹⁾	_	T1GSEL	TXCKSEL	_	_	121
BAUDCON	ABDOVF	RCIDL	_	SCKP	BRG16	_	WUE	ABDEN	304
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	90
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	91
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	95
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	303
SPBRGL				SPBRG	G<7:0>				305*
SPBRGH				SPBRG	<15:8>				305*
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	136
TXREG	EUSART Tran	nsmit Data Regi	ster						295*
TXSTA	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	302

— Unimplemented location, read as '0'. Shaded cells are not used for synchronous master transmission. Legend:

* Page provides register information. PIC16(L)F1825 only.

1: Note

26.4.1.5 Synchronous Master Reception

Data is received at the RX/DT pin. The RX/DT pin output driver is automatically disabled when the EUSART is configured for synchronous master receive operation.

In Synchronous mode, reception is enabled by setting either the Single Receive Enable bit (SREN of the RCSTA register) or the Continuous Receive Enable bit (CREN of the RCSTA register).

When SREN is set and CREN is clear, only as many clock cycles are generated as there are data bits in a single character. The SREN bit is automatically cleared at the completion of one character. When CREN is set, clocks are continuously generated until CREN is cleared. If CREN is cleared in the middle of a character the CK clock stops immediately and the partial character is discarded. If SREN and CREN are both set, then SREN is cleared at the completion of the first character and CREN takes precedence.

To initiate reception, set either SREN or CREN. Data is sampled at the RX/DT pin on the trailing edge of the TX/CK clock pin and is shifted into the Receive Shift Register (RSR). When a complete character is received into the RSR, the RCIF bit is set and the character is automatically transferred to the two character receive FIFO. The Least Significant eight bits of the top character in the receive FIFO are available in RCREG. The RCIF bit remains set as long as there are unread characters in the receive FIFO.

Note: If the RX/DT function is on an analog pin, the corresponding ANSEL bit must be cleared for the receiver to function.

26.4.1.6 Slave Clock

Synchronous data transfers use a separate clock line, which is synchronous with the data. A device configured as a slave receives the clock on the TX/CK line. The TX/CK pin output driver is automatically disabled when the device is configured for synchronous slave transmit or receive operation. Serial data bits change on the leading edge to ensure they are valid at the trailing edge of each clock. One data bit is transferred for each clock cycle. Only as many clock cycles should be received as there are data bits.

Note: If the device is configured as a slave and the TX/CK function is on an analog pin, the corresponding ANSEL bit must be cleared.

26.4.1.7 Receive Overrun Error

The receive FIFO buffer can hold two characters. An overrun error will be generated if a third character, in its entirety, is received before RCREG is read to access the FIFO. When this happens the OERR bit of the RCSTA register is set. Previous data in the FIFO will not be overwritten. The two characters in the FIFO buffer can be read, however, no additional characters will be received until the error is cleared. The OERR bit can only be cleared by clearing the overrun condition. If the overrun error occurred when the SREN bit is set and CREN is clear then the error is cleared by reading RCREG. If the overrun occurred when the CREN bit is set then the error condition is cleared by either clearing the CREN bit of the RCSTA register or by clearing the SPEN bit which resets the EUSART.

26.4.1.8 Receiving 9-bit Characters

The EUSART supports 9-bit character reception. When the RX9 bit of the RCSTA register is set the EUSART will shift 9-bits into the RSR for each character received. The RX9D bit of the RCSTA register is the ninth, and Most Significant, data bit of the top unread character in the receive FIFO. When reading 9-bit data from the receive FIFO buffer, the RX9D data bit must be read before reading the eight Least Significant bits from the RCREG.

26.4.1.9 Synchronous Master Reception Setup:

- Initialize the SPBRGH, SPBRGL register pair for the appropriate baud rate. Set or clear the BRGH and BRG16 bits, as required, to achieve the desired baud rate.
- 2. Clear the ANSEL bit for the RX pin (if applicable).
- 3. Enable the synchronous master serial port by setting bits SYNC, SPEN and CSRC.
- 4. Ensure bits CREN and SREN are clear.
- If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
- 6. If 9-bit reception is desired, set bit RX9.
- 7. Start reception by setting the SREN bit or for continuous reception, set the CREN bit.
- Interrupt flag bit RCIF will be set when reception of a character is complete. An interrupt will be generated if the enable bit RCIE was set.
- Read the RCSTA register to get the ninth bit (if enabled) and determine if any error occurred during reception.
- 10. Read the 8-bit received data by reading the RCREG register.
- If an overrun error occurs, clear the error by either clearing the CREN bit of the RCSTA register or by clearing the SPEN bit which resets the EUSART.



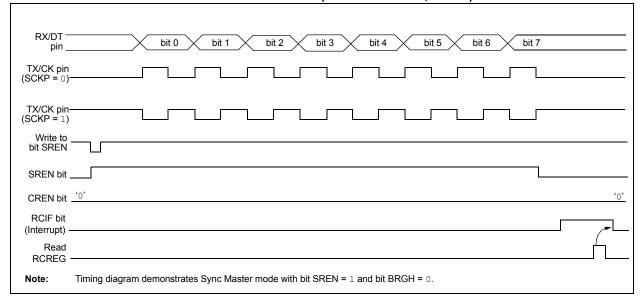


TABLE 26-8: SUMMARY OF REGISTERS ASSOCIATED WITH SYNCHRONOUS MASTER RECEPTION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
APFCON0	RXDTSEL	SDO1SEL ⁽¹⁾	SS1SEL ⁽¹⁾	_	T1GSEL	TXCKSEL	-	_	121
BAUDCON	ABDOVF	RCIDL	-	SCKP	BRG16	_	WUE	ABDEN	304
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	90
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	91
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	95
RCREG	EUSART Receive Data Register							298*	
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	303
SPBRGL	SPBRG<7:0>							305*	
SPBRGH	SPBRG<15:8>							305*	
TXSTA	CSRC TX9 TXEN SYNC SENDB BRGH TRMT TX9D						302		

Legend: — Unimplemented location, read as '0'. Shaded cells are not used for synchronous master reception.

* Page provides register information.

Note 1: PIC16(L)F1825 only.

26.4.2 SYNCHRONOUS SLAVE MODE

The following bits are used to configure the EUSART for synchronous slave operation:

- SYNC = 1
- CSRC = 0
- SREN = 0 (for transmit); SREN = 1 (for receive)
- CREN = 0 (for transmit); CREN = 1 (for receive)
- SPEN = 1

Setting the SYNC bit of the TXSTA register configures the device for synchronous operation. Clearing the CSRC bit of the TXSTA register configures the device as a slave. Clearing the SREN and CREN bits of the RCSTA register ensures that the device is in the Transmit mode, otherwise the device will be configured to receive. Setting the SPEN bit of the RCSTA register enables the EUSART.

26.4.2.1 EUSART Synchronous Slave Transmit

The operation of the Synchronous Master and Slave modes are identical (see Section 27.4.1.3 "Synchronous Master Transmission"), except in the case of the Sleep mode.

If two words are written to the TXREG and then the SLEEP instruction is executed, the following will occur:

- The first character will immediately transfer to the TSR register and transmit.
- 2. The second word will remain in TXREG register.
- 3. The TXIF bit will not be set.
- After the first character has been shifted out of TSR, the TXREG register will transfer the second character to the TSR and the TXIF bit will now be set.
- If the PEIE and TXIE bits are set, the interrupt will wake the device from Sleep and execute the next instruction. If the GIE bit is also set, the program will call the Interrupt Service Routine.

26.4.2.2 Synchronous Slave Transmission Setup:

- Set the SYNC and SPEN bits and clear the CSRC bit.
- 2. Clear the ANSEL bit for the CK pin (if applicable).
- 3. Clear the CREN and SREN bits.
- If interrupts are desired, set the TXIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
- 5. If 9-bit transmission is desired, set the TX9 bit.
- 6. Enable transmission by setting the TXEN bit.
- 7. If 9-bit transmission is selected, insert the Most Significant bit into the TX9D bit.
- 8. Start transmission by writing the Least Significant eight bits to the TXREG register.

TABLE 26-9: SUMMARY OF REGISTERS ASSOCIATED WITH SYNCHRONOUS SLAVE TRANSMISSION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
APFCON0	RXDTSEL	SDO1SEL ⁽¹⁾	SS1SEL ⁽¹⁾	_	T1GSEL	TXCKSEL	_	_	121
BAUDCON	ABDOVF	RCIDL		SCKP	BRG16	_	WUE	ABDEN	304
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	90
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	91
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	95
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	303
TXREG	EUSART Transmit Data Register							295*	
TXSTA	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	302

Legend: — Unimplemented location, read as '0'. Shaded cells are not used for Synchronous Slave Transmission.

* Page provides register information.

Note 1: PIC16(L)F1825 only.

26.4.2.3 EUSART Synchronous Slave Reception

The operation of the Synchronous Master and Slave modes is identical (Section 26.4.1.5 "Synchronous Master Reception"), with the following exceptions:

- Sleep
- CREN bit is always set, therefore the receiver is never Idle
- · SREN bit, which is a "don't care" in Slave mode

A character may be received while in Sleep mode by setting the CREN bit prior to entering Sleep. Once the word is received, the RSR register will transfer the data to the RCREG register. If the RCIE enable bit is set, the interrupt generated will wake the device from Sleep and execute the next instruction. If the GIE bit is also set, the program will branch to the interrupt vector.

26.4.2.4 Synchronous Slave Reception Setup:

- Set the SYNC and SPEN bits and clear the CSRC bit.
- Clear the ANSEL bit for both the CK and DT pins (if applicable).
- If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
- 4. If 9-bit reception is desired, set the RX9 bit.
- 5. Set the CREN bit to enable reception.
- The RCIF bit will be set when reception is complete. An interrupt will be generated if the RCIE bit was set.
- If 9-bit mode is enabled, retrieve the Most Significant bit from the RX9D bit of the RCSTA register.
- Retrieve the eight Least Significant bits from the receive FIFO by reading the RCREG register.
- If an overrun error occurs, clear the error by either clearing the CREN bit of the RCSTA register or by clearing the SPEN bit which resets the EUSART.

TABLE 26-10: SUMMARY OF REGISTERS ASSOCIATED WITH SYNCHRONOUS SLAVE RECEPTION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
APFCON0	RXDTSEL	SDO1SEL ⁽¹⁾	SS1SEL ⁽¹⁾	-	T1GSEL	TXCKSEL	-	_	121
BAUDCON	ABDOVF	RCIDL	_	SCKP	BRG16	_	WUE	ABDEN	304
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	90
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	91
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	95
RCREG	EUSART Receive Data Register							298*	
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	303
TXSTA	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	302

Legend: — Unimplemented location, read as '0'. Shaded cells are not used for synchronous slave reception.

Note 1: PIC16(L)F1825 only.

^{*} Page provides register information.

26.5 EUSART Operation During Sleep

The EUSART will remain active during Sleep only in the Synchronous Slave mode. All other modes require the system clock and therefore cannot generate the necessary signals to run the Transmit or Receive Shift registers during Sleep.

Synchronous Slave mode uses an externally generated clock to run the Transmit and Receive Shift registers.

26.5.1 SYNCHRONOUS RECEIVE DURING SI FEP

To receive during Sleep, all the following conditions must be met before entering Sleep mode:

- RCSTA and TXSTA Control registers must be configured for Synchronous Slave Reception (see Section 26.4.2.4 "Synchronous Slave Reception Setup:").
- If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
- The RCIF interrupt flag must be cleared by reading RCREG to unload any pending characters in the receive buffer.

Upon entering Sleep mode, the device will be ready to accept data and clocks on the RX/DT and TX/CK pins, respectively. When the data word has been completely clocked in by the external device, the RCIF interrupt flag bit of the PIR1 register will be set. Thereby, waking the processor from Sleep.

Upon waking from Sleep, the instruction following the SLEEP instruction will be executed. If the GIE, Global Interrupt Enable, bit of the INTCON register is also set, then the Interrupt Service Routine at address 004h will be called.

26.5.2 SYNCHRONOUS TRANSMIT DURING SLEEP

To transmit during Sleep, all the following conditions must be met before entering Sleep mode:

- RCSTA and TXSTA Control registers must be configured for synchronous slave transmission (see Section 26.4.2.2 "Synchronous Slave Transmission Setup:").
- The TXIF interrupt flag must be cleared by writing the output data to the TXREG, thereby filling the TSR and transmit buffer.
- If interrupts are desired, set the TXIE bit of the PIE1 register and the PEIE bit of the INTCON register.
- Interrupt enable bits TXIE of the PIE1 register and PEIE of the INTCON register must set.

Upon entering Sleep mode, the device will be ready to accept clocks on TX/CK pin and transmit data on the RX/DT pin. When the data word in the TSR has been completely clocked out by the external device, the pending byte in the TXREG will transfer to the TSR and the TXIF flag will be set. Thereby, waking the processor from Sleep. At this point, the TXREG is available to accept another character for transmission, which will clear the TXIF flag.

Upon waking from Sleep, the instruction following the SLEEP instruction will be executed. If the Global Interrupt Enable (GIE) bit is also set then the Interrupt Service Routine at address 0004h will be called.

26.5.3 ALTERNATE PIN LOCATIONS

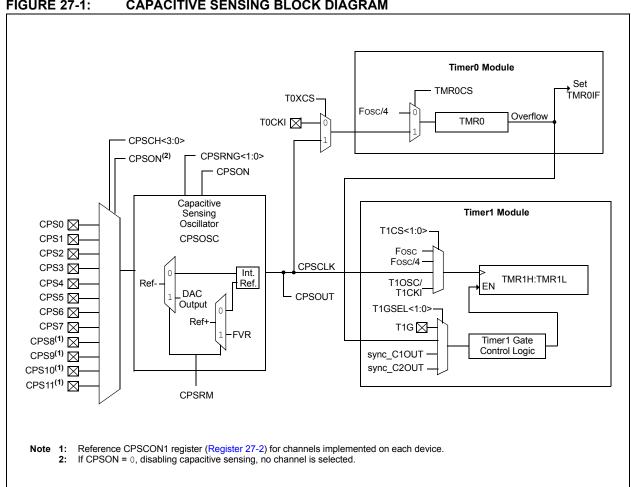
This module incorporates I/O pins that can be moved to other locations with the use of the alternate pin function registers, APFCON0 and APFCON1. To determine which pins can be moved and what their default locations are upon a Reset, see Section 12.1 "Alternate Pin Function" for more information.

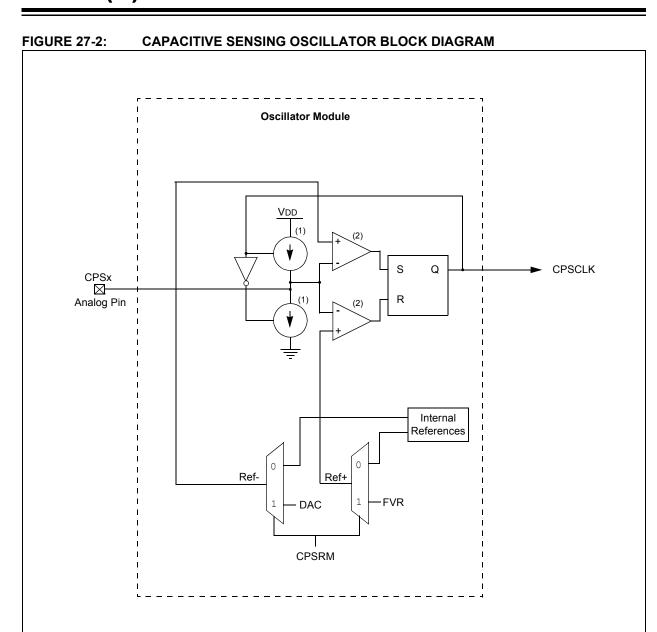
27.0 CAPACITIVE SENSING (CPS) **MODULE**

The Capacitive Sensing (CPS) module allows for an interaction with an end user without a mechanical interface. In a typical application, the CPS module is attached to a pad on a Printed Circuit Board (PCB), which is electrically isolated from the end user. When the end user places their finger over the PCB pad, a capacitive load is added, causing a frequency shift in the CPS module. The CPS module requires software and at least one timer resource to determine the change in frequency. Key features of this module include:

- · Analog MUX for monitoring multiple inputs
- · Capacitive sensing oscillator
- · Multiple Power modes
- · High-power range with variable voltage references
- · Multiple timer resources
- · Software control
- · Operation during Sleep

FIGURE 27-1: CAPACITIVE SENSING BLOCK DIAGRAM





2: Comparators remain active in Noise Detection mode.

27.1 Analog MUX

The CPS module can monitor up to four inputs for the PIC16(L)F1825 (CPSCH<7:0>) and up to eight inputs for the PIC16(L)F1829 (CPSCH<11:0>). See Register 27-2 for details. The capacitive sensing inputs are defined as CPS<11:0>, as applicable to device. To determine if a frequency change has occurred the user must:

- Select the appropriate CPS pin by setting the CPSCH<3:0> bits of the CPSCON1 register.
- · Set the corresponding ANSEL bit.
- · Set the corresponding TRIS bit.
- · Run the software algorithm.

The CPS module can monitor up to four inputs for the PIC16(L)F1825 (CPSCH<7:0>) and up to eight inputs for the PIC16(L)F1829 (CPSCH<11:0>). See Register 27-2 for details. The capacitive sensing inputs are defined as CPS<11:0>, as applicable to device. To determine if a frequency change has occurred the user must:

- Select the appropriate CPS pin by setting the appropriate CPSCH bits of the CPSCON1 register.
- · Set the corresponding ANSEL bit.
- · Set the corresponding TRIS bit.
- · Run the software algorithm.

Selection of the CPSx pin while the module is enabled will cause the capacitive sensing oscillator to be on the CPSx pin. Failure to set the corresponding ANSEL and TRIS bits can cause the capacitive sensing oscillator to stop, leading to false frequency readings.

27.2 Capacitive Sensing Oscillator

The capacitive sensing oscillator consists of a constant current source and a constant current sink, to produce a triangle waveform. The CPSOUT bit of the CPSCON0 register shows the status of the capacitive sensing oscillator, whether it is a sinking or sourcing current. The oscillator is designed to drive a capacitive load (single PCB pad) and at the same time, be a clock source to either Timer0 or Timer1. The oscillator has three different current settings as defined by CPSRNG<1:0> of the CPSCON0 register. The different current settings for the oscillator serve two purposes:

- Maximize the number of counts in a timer for a fixed time base.
- Maximize the count differential in the timer during a change in frequency.

27.3 Voltage References

The capacitive sensing oscillator uses voltage references to provide two voltage thresholds for oscillation. The upper voltage threshold is referred to as Ref+ and the lower voltage threshold is referred to as Ref-.

The user can elect to use Fixed Voltage References, which are internal to the capacitive sensing oscillator, or variable voltage references, which are supplied by the Fixed Voltage Reference (FVR) module and the Digital-to-Analog Converter (DAC) module.

When the Fixed Voltage References are used, the Vss voltage determines the lower threshold level (Ref-) and the VDD voltage determines the upper threshold level (Ref+).

When the variable voltage references are used, the DAC voltage determines the lower threshold level (Ref-) and the FVR voltage determines the upper threshold level (Ref+). An advantage of using these reference sources is that oscillation frequency remains constant with changes in VDD.

Different oscillation frequencies can be obtained through the use of these variable voltage references. The more the upper voltage reference level is lowered and the more the lower voltage reference level is raised, the higher the capacitive sensing oscillator frequency becomes.

Selection between the voltage references is controlled by the CPSRM bit of the CPSCON0 register. Setting this bit selects the variable voltage references and clearing this bit selects the Fixed Voltage References.

Please see Section 14.0 "Fixed Voltage Reference (FVR)" and Section 17.0 "Digital-to-Analog Converter (DAC) Module" for more information on configuring the variable voltage levels.

27.4 Power Modes

The capacitive sensing oscillator can operate in one of seven different power modes. The power modes are separated into two ranges; the low range and the high range.

When the oscillator's low range is selected, the fixed internal voltage references of the capacitive sensing oscillator are being used. When the oscillator's high range is selected, the variable voltage references supplied by the FVR and DAC modules are being used. Selection between the voltage references is controlled by the CPSRM bit of the CPSCON0 register. See Section 27.3 "Voltage References" for more information.

Within each range there are three distinct Power modes; low, medium and high. Current consumption is dependent upon the range and mode selected. Selecting Power modes within each range is accomplished by configuring the CPSRNG <1:0> bits in the CPSCON0 register. See Table 27-1 for proper Power mode selection.

The remaining mode is a Noise Detection mode that resides within the high range. The Noise Detection mode is unique in that it disables the sinking and sourcing of current on the analog pin but leaves the rest of the oscillator circuitry active. This reduces the oscillation frequency on the analog pin to zero and also greatly reduces the current consumed by the oscillator module.

When noise is introduced onto the pin, the oscillator is driven at the frequency determined by the noise. This produces a detectable signal at the comparator output, indicating the presence of activity on the pin.

Figure 27-2 shows a more detailed drawing of the current sources and comparators associated with the oscillator.

TABLE 27-1: POWER MODE SELECTION

CPSRM	Range	CPSRNG<1:0>	Mode	Nominal Current ⁽¹⁾	
		00	Off	0.0 μΑ	
0	Low	01	Low	0.1 μΑ	
0		10	Medium	1.2 μΑ	
		11	High	18 μΑ	
		00	Noise Detection	0.0 μΑ	
1	Lliab	01	Low	9 μΑ	
Ţ	High	10	Medium	30 μΑ	
l		11	High	100 μΑ	

Note 1: See Section 30.0 "Electrical Specifications" for more information.

27.5 Timer Resources

To measure the change in frequency of the capacitive sensing oscillator, a fixed time base is required. For the period of the fixed time base, the capacitive sensing oscillator is used to clock either Timer0 or Timer1. The frequency of the capacitive sensing oscillator is equal to the number of counts in the timer divided by the period of the fixed time base.

27.6 Fixed Time Base

To measure the frequency of the capacitive sensing oscillator, a fixed time base is required. Any timer resource or software loop can be used to establish the fixed time base. It is up to the end user to determine the method in which the fixed time base is generated.

Note:	The fixed time base can not be generated				
	by the timer resource that the capacitive				
	sensing oscillator is clocking.				

27.6.1 TIMER0

To select Timer0 as the timer resource for the CPS module:

- · Set the T0XCS bit of the CPSCON0 register.
- Clear the TMR0CS bit of the OPTION_REG register.

When Timer0 is chosen as the timer resource, the capacitive sensing oscillator will be the clock source for Timer0. Refer to **Section 20.0 "Timer0 Module"** for additional information.

27.6.2 TIMER1

To select Timer1 as the timer resource for the CPS module, set the TMR1CS<1:0> of the T1CON register to '11'. When Timer1 is chosen as the timer resource, the capacitive sensing oscillator will be the clock source for Timer1. Because the Timer1 module has a gate control, developing a time base for the frequency measurement can be simplified by using the Timer0 overflow flag.

It is recommend that the Timer0 overflow flag, in conjunction with the Toggle mode of the Timer1 Gate, be used to develop the fixed time base required by the software portion of the CPS module. Refer to Section 21.12 "Timer1 Gate Control Register" for additional information.

TABLE 27-2: TIMER1 ENABLE FUNCTION

TMR10N	TMR1GE	Timer1 Operation
0	0	Off
0	1	Off
1	0	On
1	1	Count Enabled by input

27.7 Software Control

The software portion of the CPS module is required to determine the change in frequency of the capacitive sensing oscillator. This is accomplished by the following:

- Setting a fixed time base to acquire counts on Timer0 or Timer1.
- Establishing the nominal frequency for the capacitive sensing oscillator.
- Establishing the reduced frequency for the capacitive sensing oscillator due to an additional capacitive load.
- · Set the frequency threshold.

27.7.1 NOMINAL FREQUENCY (NO CAPACITIVE LOAD)

To determine the nominal frequency of the capacitive sensing oscillator:

- Remove any extra capacitive load on the selected CPSx pin.
- At the start of the fixed time base, clear the timer resource.
- At the end of the fixed time base save the value in the timer resource.

The value of the timer resource is the number of oscillations of the capacitive sensing oscillator for the given time base. The frequency of the capacitive sensing oscillator is equal to the number of counts on in the timer divided by the period of the fixed time base.

27.7.2 REDUCED FREQUENCY (ADDITIONAL CAPACITIVE LOAD)

The extra capacitive load will cause the frequency of the capacitive sensing oscillator to decrease. To determine the reduced frequency of the capacitive sensing oscillator:

- Add a typical capacitive load on the selected CPSx pin.
- Use the same fixed time base as the nominal frequency measurement.
- At the start of the fixed time base, clear the timer resource.
- At the end of the fixed time base save the value in the timer resource.

The value of the timer resource is the number of oscillations of the capacitive sensing oscillator with an additional capacitive load. The frequency of the capacitive sensing oscillator is equal to the number of counts on in the timer divided by the period of the fixed time base. This frequency should be less than the value obtained during the nominal frequency measurement.

27.7.3 FREQUENCY THRESHOLD

The frequency threshold should be placed midway between the value of nominal frequency and the reduced frequency of the capacitive sensing oscillator. Refer to Application Note AN1103, "Software Handling for Capacitive Sensing" (DS01103) for more detailed information on the software required for CPS module.

Note: For more information on general capacitive sensing refer to Application Notes:

- AN1101, "Introduction to Capacitive Sensing" (DS01101)
- AN1102, "Layout and Physical Design Guidelines for Capacitive Sensing" (DS01102)

27.8 Operation during Sleep

The capacitive sensing oscillator will continue to run as long as the module is enabled, independent of the part being in Sleep. In order for the software to determine if a frequency change has occurred, the part must be awake. However, the part does not have to be awake when the timer resource is acquiring counts.

Note: Timer0 does not operate when in Sleep, and therefore cannot be used for capacitive sense measurements in Sleep.

REGISTER 27-1: CPSCON0: CAPACITIVE SENSING CONTROL REGISTER 0

R/W-0/0	R/W-0/0	U-0	U-0	R/W-0/0	R/W-0/0	R-0/0	R/W-0/0
CPSON	CPSRM	_	_	CPSRNG<1:0>		CPSOUT	T0XCS
bit 7							bit 0

Legend:

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

u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set '0' = Bit is cleared

bit 7 CPSON: CPS Module Enable bit

1 = CPS module is enabled

0 = CPS module is disabled

bit 6 CPSRM: Capacitive Sensing Reference Mode bit

1 = CPS module is in high range. DAC and FVR provide oscillator voltage references.
 0 = CPS module is in the low range. Internal oscillator voltage references are used.

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

bit 3-2 CPSRNG<1:0>: Capacitive Sensing Current Range bit

If CPSRM = 0 (low range):

00 = Oscillator is off

01 = Oscillator is in Low Range. Charge/Discharge Current is nominally 0.1 μA

10 = Oscillator is in Medium Range. Charge/Discharge Current is nominally 1.2 μA

11 = Oscillator is in High Range. Charge/Discharge Current is nominally 18 μA

If CPSRM = 1 (high range):

00 = Oscillator is on. Noise Detection mode. No Charge/Discharge current is supplied.

01 = Oscillator is in Low Range. Charge/Discharge Current is nominally 9 μA

10 = Oscillator is in Medium Range. Charge/Discharge Current is nominally 30 μA

11 = Oscillator is in High Range. Charge/Discharge Current is nominally 100 μA

bit 1 CPSOUT: Capacitive Sensing Oscillator Status bit

1 = Oscillator is sourcing current (Current flowing out of the pin)

0 = Oscillator is sinking current (Current flowing into the pin)

bit 0 TOXCS: Timer0 External Clock Source Select bit

<u>If TMR0CS = 1:</u>

The T0XCS bit controls which clock external to the core/Timer0 module supplies Timer0:

1 = Timer0 clock source is the capacitive sensing oscillator

0 = Timer0 clock source is the T0CKI pin

If TMROCS = 0:

Timer0 clock source is controlled by the core/Timer0 module and is Fosc/4

REGISTER 27-2: **CPSCON1: CAPACITIVE SENSING CONTROL REGISTER 1**

U-0	U-0	U-0	U-0	R/W-0/0 ⁽¹⁾	R/W-0/0	R/W-0/0	R/W-0/0
_	_	_	_	CPSC	H<3:2>	CPSCI	H<1:0>
bit 7							bit 0

Legend: R = Readable bit W = Writable bit U = Unimplemented bit, read as '0' u = Bit is unchanged -n/n = Value at POR and BOR/Value at all other Resets x = Bit is unknown '1' = Bit is set '0' = Bit is cleared

bit 7-4 Unimplemented: Read as '0'

bit 3-0 CPSCH<3:0>: Capacitive Sensing Channel Select bits

These bits are ignored. No channel is selected.

If CPSON = 1:

0000 = channel 0, (CPS0)

0001 = channel 1, (CPS1)

0010 = channel 2, (CPS2)

0011 = channel 3, (CPS3)

0100 = channel 4, (CPS4)

0101 = channel 5, (CPS5)

0110 = channel 6, (CPS6)

0111 = channel 7, (CPS7)

1000 = channel 8, (CPS4)(1)

1001 = channel 9, (CPS5)⁽¹⁾

1010 = channel 10, (CPS6)⁽¹⁾

1011 = channel 11, (CPS7)⁽¹⁾

1100 = Reserved. Do not use.

1111 = Reserved. Do not use.

Note 1: These channels are only implemented on the PIC16(L)F1829.

TABLE 27-3: SUMMARY OF REGISTERS ASSOCIATED WITH CAPACITIVE SENSING

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ANSELA	_	_	_	ANSA4	_	ANSA2	ANSA1	ANSA0	126
ANSELC	ANSC7 ⁽¹⁾	ANSC6 ⁽¹⁾	_	_	ANSC3	ANSC2	ANSC1	ANSC0	137
CPSCON0	CPSON	CPSRM	_	_	CPSRN	IG<1:0>	CPSOUT	T0XCS	327
CPSCON1	_	_	_	_		CPSCI	H<3:0>		328
INLVLA	_	_	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	127
INLVLB ⁽¹⁾	INLVLB7	INLVLB6	INLVLB5	INLVLB4	1	1	1	1	132
INLVLC	INLVLC7 ⁽¹⁾	INLVLC6 ⁽¹⁾	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	138
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	90
OPTION_REG	WPUEN	INTEDG	TMR0CS	TMR0SE	PSA	PS2	PS1	PS0	185
T1CON	TMR1CS1	TMR1CS0	T1CKPS1	T1CKPS0	T10SCEN	T1SYNC		TMR10N	195
TRISA	_	_	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	125
TRISB ⁽¹⁾	TRISB7	TRISB6	TRISB5	TRISB4	-	-	1	-	131
TRISC	TRISC7 ⁽¹⁾	TRISC6 ⁽¹⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	136

Legend: — Unimplemented loc Note 1: PIC16(L)F1829 only. Unimplemented locations, read as '0'. Shaded cells are not used by the CPS module.

28.0 IN-CIRCUIT SERIAL PROGRAMMING™ (ICSP™)

ICSPTM programming allows customers to manufacture circuit boards with unprogrammed devices. Programming can be done after the assembly process allowing the device to be programmed with the most recent firmware or a custom firmware. Five pins are needed for ICSPTM programming:

- ICSPCLK
- ICSPDAT
- MCLR/VPP
- VDD
- Vss

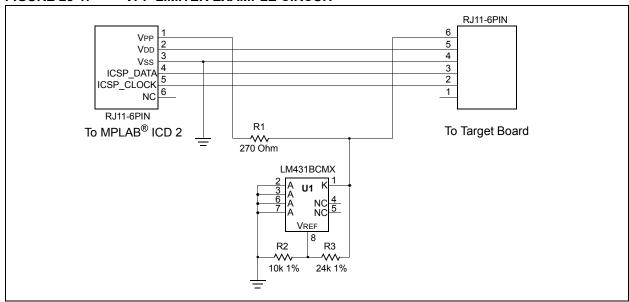
In Program/Verify mode the Program Memory, User IDs and the Configuration Words are programmed through serial communications. The ICSPDAT pin is a bidirectional I/O used for transferring the serial data and the ICSPCLK pin is the clock input. For more information on ICSP™ refer to the "PIC16F/LF182X/PIC12F/LF1822 Memory Programming Specification" (DS41403).

28.1 High-Voltage Programming Entry Mode

The device is placed into High-Voltage Programming Entry mode by holding the ICSPCLK and ICSPDAT pins low then raising the voltage on MCLR/VPP to VIHH.

Some programmers produce VPP greater than VIHH (9.0V), an external circuit is required to limit the VPP voltage. See Figure 28-1 for example circuit.

FIGURE 28-1: VPP LIMITER EXAMPLE CIRCUIT



Note: The MPLAB ICD 2 produces a VPP voltage greater than the maximum VPP specification of the PIC16(L)F1825/1829.

28.2 Low-Voltage Programming Entry Mode

The Low-Voltage Programming Entry mode allows the PIC16(L)F1825/1829 devices to be programmed using VDD only, without high voltage. When the LVP bit of Configuration Word 2 is set to '1', the low-voltage ICSP programming entry is enabled. To disable the Low-Voltage ICSP mode, the LVP bit must be programmed to '0'.

Entry into the Low-Voltage Programming Entry mode requires the following steps:

- 1. MCLR is brought to VIL.
- A 32-bit key sequence is presented on ICSPDAT, while clocking ICSPCLK.

Once the key sequence is complete, $\overline{\text{MCLR}}$ must be held at VIL for as long as Program/Verify mode is to be maintained.

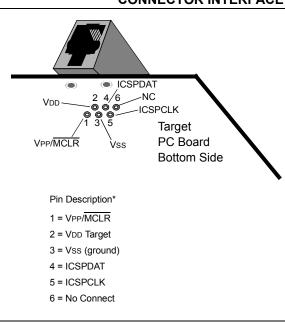
If low-voltage programming is enabled (LVP = 1), the $\overline{\text{MCLR}}$ Reset function is automatically enabled and cannot be disabled. See **Section 7.3 "MCLR"** for more information.

The LVP bit can only be reprogrammed to '0' by using the High-Voltage Programming mode.

28.3 Common Programming Interfaces

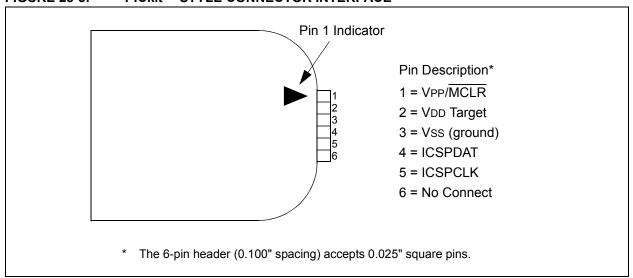
Connection to a target device is typically done through an ICSP™ header. A commonly found connector on development tools is the RJ-11 in the 6P6C (6-pin, 6 connector) configuration. See Figure 28-2.

FIGURE 28-2: ICD RJ-11 STYLE CONNECTOR INTERFACE



Another connector often found in use with the PICkit™ programmers is a standard 6-pin header with 0.1 inch spacing. Refer to Figure 28-3.

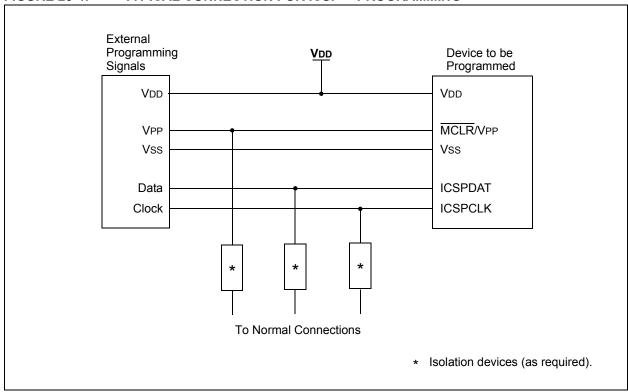
FIGURE 28-3: PICkit™ STYLE CONNECTOR INTERFACE



For additional interface recommendations, refer to your specific device programmer manual prior to PCB design.

It is recommended that isolation devices be used to separate the programming pins from other circuitry. The type of isolation is highly dependent on the specific application and may include devices such as resistors, diodes, or even jumpers. See Figure 28-4 for more information.

FIGURE 28-4: TYPICAL CONNECTION FOR ICSP™ PROGRAMMING



NOTES:

29.0 INSTRUCTION SET SUMMARY

Each PIC16 instruction is a 14-bit word containing the operation code (opcode) and all required operands. The opcodes are broken into three broad categories.

- · Byte Oriented
- · Bit Oriented
- · Literal and Control

The literal and control category contains the most varied instruction word format.

Table 29-3 lists the instructions recognized by the MPASM $^{\text{TM}}$ assembler.

All instructions are executed within a single instruction cycle, with the following exceptions, which may take two or three cycles:

- Subroutine takes two cycles (CALL, CALLW)
- Returns from interrupts or subroutines take two cycles (RETURN, RETLW, RETFIE)
- Program branching takes two cycles (GOTO, BRA, BRW, BTFSS, BTFSC, DECFSZ, INCSFZ)
- One additional instruction cycle will be used when any instruction references an indirect file register and the file select register is pointing to program memory.

One instruction cycle consists of four oscillator cycles; for an oscillator frequency of 4 MHz, this gives a nominal instruction execution rate of 1 MHz.

All instruction examples use the format '0xhh' to represent a hexadecimal number, where 'h' signifies a hexadecimal digit.

29.1 Read-Modify-Write Operations

Any instruction that specifies a file register as part of the instruction performs a Read-Modify-Write (R-M-W) operation. The register is read, the data is modified, and the result is stored according to either the instruction, or the destination designator 'd'. A read operation is performed on a register even if the instruction writes to that register.

TABLE 29-1: OPCODE FIELD DESCRIPTIONS

Field	Description
f	Register file address (0x00 to 0x7F)
W	Working register (accumulator)
b	Bit address within an 8-bit file register
k	Literal field, constant data or label
х	Don't care location (= 0 or 1). The assembler will generate code with x = 0. It is the recommended form of use for compatibility with all Microchip software tools.
d	Destination select; d = 0: store result in W, d = 1: store result in file register f. Default is d = 1.
n	FSR or INDF number. (0-1)
mm	Pre-post increment-decrement mode selection

TABLE 29-2: ABBREVIATION DESCRIPTIONS

Field	Description
PC	Program Counter
TO	Time-out bit
С	Carry bit
DC	Digit carry bit
Z	Zero bit
PD	Power-down bit

FIGURE 29-1: GENERAL FORMAT FOR INSTRUCTIONS

	IN					
Byte-oriented file	regi 8	ster (оре 6	ratio	าร	0
OPCODE		d		f	(FILE #	‡)
d = 0 for dest d = 1 for dest f = 7-bit file re	inati	on f		ss		
Bit-oriented file re	gist	er op	era	tions	;	
13	10	9	7	6		С
OPCODE		b (Bl	Τ#)	f (FILE	#)
b = 3-bit bit ac f = 7-bit file re			dres	ss		
Literal and contro	l op	eratio	ons			
General						
13		8	7		/!:	C
OPCODE				K	(literal)
k = 8-bit imme	ediat	e val	ue			
CALL and GOTO ins	truct	tions	only	,		
13 11	10					0
OPCODE			k	(litera	al)	
k = 11-bit imm	nedia	ate va	lue			
MOVLP instruction o	nhu					
MOVED INSUUCION C						
13	лпу		7	6		0
		e val			(literal)
OPCODE k = 7-bit imme	ediat	e val		k	(literal)
OPCODE k = 7-bit imme	ediat	e val				0
OPCODE k = 7-bit imme	ediat		ue	k	(literal	0
OPCODE k = 7-bit imme MOVLB instruction of 13 OPCODE	ediat		ue	k		0
ADVLB instruction of 13 OPCODE k = 7-bit imme OPCODE k = 5-bit imme BRA instruction only	ediat only ediat		ue	5 4	k (lite) 0 ral)
ADVLB instruction of 13 OPCODE k = 5-bit imme BRA instruction only	ediat only ediat	e val	ue	5 4) 0 ral)
ADVLB instruction of 13 OPCODE k = 7-bit imme OPCODE k = 5-bit imme BRA instruction only	ediat only ediat	e val	ue ue	5 4	k (lite) 0 ral)
ACCUPATION OF THE PROPERTY OF	ediat pnly ediat	e val	ue ue	5 4	k (lite) () ()
APPROVED BY A PROVIDE A STATE OF CODE k = 5-bit immed and of the control of the	ediat pnly ediat	e val	ue ue	k 5 4	k (lite) (ral)
A PCODE k = 7-bit immed MOVLB instruction of 13 OPCODE k = 5-bit immed BRA instruction only 13 OPCODE k = 9-bit immed FSR Offset instruction of 13	ediatonly ediatory ediators	e vali	ue ue 6 n	k 5 4	k (lite) (ral)
ADDED A TOP	ediatonly ediatons ediatons	e value 9 8 7 7 SSR tte value 10 10 10 10 10 10 10 10 10 10 10 10 10	ue ue 6 n	k 5 4	k (lite	0 (() () () ()
A PCODE k = 7-bit immed MOVLB instruction of 13 OPCODE k = 5-bit immed BRA instruction only 13 OPCODE k = 9-bit immed FSR Offset instruct 13 OPCODE n = appropriate k = 6-bit immed FSR Increment instructions	ediatonly ediatons ediatons	e value 9 8 7 7 SSR tte value 10 10 10 10 10 10 10 10 10 10 10 10 10	ue ue 6 n	5 4 F	k (literal	0 (() () () () ()
A PCODE k = 7-bit immed MOVLB instruction of 13 OPCODE k = 5-bit immed BRA instruction only 13 OPCODE k = 9-bit immed TSR Offset instruct OPCODE n = appropriate k = 6-bit immed FSR Increment inst 13	ediatonly ediatory ediatory ediators ed	9 8 7 SR ste val	ue ue 6 n	5 4 F	k (literal	0 (() () () () ()
A PCODE k = 7-bit imme MOVLB instruction of 13 OPCODE k = 5-bit imme BRA instruction only 13 OPCODE k = 9-bit imme FSR Offset instruct 13 OPCODE n = appropriat k = 6-bit imm FSR Increment inst 13 OPCODE n = appropria	ediatonly ediatory ediatory ediators ed	9 8 7 SR ste val	ue ue 6 n	5 4 F	k (literal) (ral)

TABLE 29-3: PIC16(L)F1825/1829 ENHANCED INSTRUCTION SET

Mnemonic,		December 2	Cycles		14-Bit	Opcode)	Status	Notes
Oper	ands	Description		MSb			LSb	Affected	Notes
		BYTE-ORIENTED FILE REGIS	TER OPE	RATIO	NS				
ADDWF	f, d	Add W and f	1	00	0111	dfff	ffff	C, DC, Z	2
ADDWFC	f, d	Add with Carry W and f	1	11	1101	dfff	ffff	C, DC, Z	2
ANDWF	f, d	AND W with f	1	00	0101	dfff	ffff	Z	2
ASRF	f, d	Arithmetic Right Shift	1	11	0111	dfff	ffff	C, Z	2
LSLF	f, d	Logical Left Shift	1	11	0101	dfff	ffff	C, Z	2
LSRF	f, d	Logical Right Shift	1	11	0110	dfff	ffff	C, Z	2
CLRF	f	Clear f	1	00	0001	lfff	ffff	Z	2
CLRW	_	Clear W	1	00	0001	0000	00xx	Z	
COMF	f, d	Complement f	1	0.0	1001	dfff	ffff	Z	2
DECF	f, d	Decrement f	1	0.0	0011	dfff	ffff	Z	2
INCF	f, d	Increment f	1	0.0	1010		ffff	Z	2
IORWF	f, d	Inclusive OR W with f	1	0.0	0100	dfff	ffff	Z	2
MOVF	f, d	Move f	1	0.0	1000	dfff	ffff	z	2
MOVWF	f	Move W to f	1	00	0000	1fff			2
RLF	f, d	Rotate Left f through Carry	1	00	1101		ffff	С	2
RRF	f, d	Rotate Right f through Carry	1	0.0	1100		ffff	C	2
SUBWF	f, d	Subtract W from f	1	00	0010		ffff	C, DC, Z	2
SUBWFB	f, d	Subtract with Borrow W from f	i	11	1011		ffff	C, DC, Z	2
SWAPF	f, d	Swap nibbles in f	i	0.0	1110	dfff	ffff	0, 50, 2	2
XORWF	f, d	Exclusive OR W with f	1	00	0110	dfff	ffff	Z	2
XOIWI	i, u	BYTE ORIENTED SKIP O			0110	dili			1-
DECECZ	f, d	Decrement f, Skip if 0	1(2)	00	1011	dfff	ffff		1, 2
DECFSZ INCFSZ	f, d	Increment f, Skip if 0	1(2)	00	1111	dfff	ffff		1, 2
		BIT-ORIENTED FILE REGIST	ER OPER	RATION	IS				
BCF	f, b	Bit Clear f	1	01	0.0bb	bfff	ffff		2
BSF	f, b	Bit Set f	1	01	01bb	bfff	ffff		2
		BIT-ORIENTED SKIP O	L PERATIO	NS	1				
BTFSC	f, b	Bit Test f, Skip if Clear	1 (2)	01	10bb	bfff	ffff		1, 2
BTFSS	f, b	Bit Test f, Skip if Set	1 (2)	01	11bb	bfff	ffff		1, 2
	PERATION	•	()						
ADDLW	k	Add literal and W	1	11	1110	kkkk	kkkk	C, DC, Z	
ANDLW	k	AND literal with W	1	11		kkkk		Z	
IORLW	k	Inclusive OR literal with W	1	11	1000	kkkk		Z	
MOVLB	k	Move literal to BSR	1	00	0000	001k		_	
MOVLP	k	Move literal to PCLATH		11	0001	1kkk			
MOVLW	k	Move literal to W		11	0000	kkkk			
SUBLW	k	Subtract W from literal	1	11	1100	kkkk		C, DC, Z	
XORLW	k	Exclusive OR literal with W		11		kkkk		Z	
		Counter (PC) is modified, or a conditional test is t							<u> </u>

Note 1:If the Program Counter (PC) is modified, or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a NOP.

^{2:} If this instruction addresses an INDF register and the MSb of the corresponding FSR is set, this instruction will require one additional instruction cycle.

TABLE 29-3: PIC16(L)F1825/1829 ENHANCED INSTRUCTION SET (CONTINUED)

Mnen	nonic,	Description	Cycles		14-Bit	Opcode)	Status	Notes
Oper	ands	Description		MSb			LSb	Affected	Notes
		CONTROL OPERA	TIONS						
BRA	k	Relative Branch	2	11	001k	kkkk	kkkk		
BRW	_	Relative Branch with W	2	00	0000	0000	1011		
CALL	k	Call Subroutine	2	10	0kkk	kkkk	kkkk		
CALLW	_	Call Subroutine with W	2	00	0000	0000	1010		
GOTO	k	Go to address	2	10	1kkk	kkkk	kkkk		
RETFIE	k	Return from interrupt	2	00	0000	0000	1001		
RETLW	k	Return with literal in W	2	11	0100	kkkk	kkkk		
RETURN	_	Return from Subroutine	2	00	0000	0000	1000		
		INHERENT OPERA	TIONS	•				•	
CLRWDT	_	Clear Watchdog Timer	1	00	0000	0110	0100	TO, PD	
NOP	_	No Operation	1	00	0000	0000	0000		
OPTION	_	Load OPTION_REG register with W	1	00	0000	0110	0010		
RESET	-	Software device Reset	1	00	0000	0000	0001		
SLEEP	-	Go into Standby mode	1	00	0000	0110	0011	TO, PD	
TRIS	f	Load TRIS register with W	1	00	0000	0110	Offf		
		C-COMPILER OPT	IMIZED	•				•	
ADDFSR	n, k	Add Literal k to FSRn	1	11	0001	0nkk	kkkk		
MOVIW	n mm	Move Indirect FSRn to W with pre/post inc/dec	1	0.0	0000	0001	0nmm	Z	2, 3
		modifier, mm							
	k[n]	Move INDFn to W, Indexed Indirect.	1	11	1111	0nkk	kkkk	Z	2
MOVWI	n mm	Move W to Indirect FSRn with pre/post inc/dec	1	00	0000	0001	1nmm		2, 3
	k[n]	modifier, mm Move W to INDFn, Indexed Indirect.	1	11	1111	1nkk	kkkk		2

Note 1:If the Program Counter (PC) is modified, or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a NOP.

^{2:} If this instruction addresses an INDF register and the MSb of the corresponding FSR is set, this instruction will require one additional instruction cycle.

^{3:} See Table in the MOVIW and MOVWI instruction descriptions.

29.2 Instruction Descriptions

ADDFSR	Add Literal to FSRn				
Syntax:	[label] ADDFSR FSRn, k				
Operands:	$-32 \le k \le 31$ $n \in [0, 1]$				
Operation:	$FSR(n) + k \rightarrow FSR(n)$				
Status Affected:	None				
Description:	The signed 6-bit literal 'k' is added to the contents of the FSRnH:FSRnL register pair.				
	FSRn is limited to the range 0000h - FFFFh. Moving beyond these bounds will cause the FSR to wrap around.				

ANDLW	AND literal with W
Syntax:	[label] ANDLW k
Operands:	$0 \leq k \leq 255$
Operation:	(W) .AND. $(k) \rightarrow (W)$
Status Affected:	Z
Description:	The contents of W register are AND'ed with the eight-bit literal 'k'. The result is placed in the W register.

ADDLW	Add literal and W
Syntax:	[label] ADDLW k
Operands:	$0 \leq k \leq 255$
Operation:	$(W) + k \rightarrow (W)$
Status Affected:	C, DC, Z
Description:	The contents of the W register are added to the eight-bit literal 'k' and the result is placed in the W register.

ANDWF	AND W with f
Syntax:	[label] ANDWF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(W) .AND. (f) \rightarrow (destination)
Status Affected:	Z
Description:	AND the W register with register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f'.

ADDWF	Add W and f
Syntax:	[label] ADDWF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(W) + (f) \rightarrow (destination)
Status Affected:	C, DC, Z
Description:	Add the contents of the W register with register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f'.

ASRF	Arithmetic Right Shift
Syntax:	[label] ASRF f {,d}
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	$(f<7>)\rightarrow dest<7>$ $(f<7:1>)\rightarrow dest<6:0>,$ $(f<0>)\rightarrow C,$
Status Affected:	C, Z
Description:	The contents of register 'f' are shifted one bit to the right through the Carry flag. The MSb remains unchanged. If 'd' is '0', the result is placed in W. If 'd' is '1', the result is stored back in register 'f'.

register f

ADDWFC	ADD W and CARRY bit to f
Syntax:	[label] ADDWFC f {,d}
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	$(W) + (f) + (C) \rightarrow dest$
Status Affected:	C, DC, Z
Description:	Add W, the Carry flag and data memory location 'f'. If 'd' is '0', the result is placed in W. If 'd' is '1', the result is placed in data memory location 'f'.

BCF	Bit Clear f
Syntax:	[label] BCF f,b
Operands:	$\begin{aligned} 0 &\leq f \leq 127 \\ 0 &\leq b \leq 7 \end{aligned}$
Operation:	$0 \rightarrow (f < b >)$
Status Affected:	None
Description:	Bit 'b' in register 'f' is cleared.

BTFSC	Bit Test f, Skip if Clear
Syntax:	[label] BTFSC f,b
Operands:	$\begin{aligned} 0 &\leq f \leq 127 \\ 0 &\leq b \leq 7 \end{aligned}$
Operation:	skip if $(f < b >) = 0$
Status Affected:	None
Description:	If bit 'b' in register 'f' is '1', the next instruction is executed. If bit 'b', in register 'f', is '0', the next instruction is discarded, and a NOP is executed instead, making this a 2-cycle instruction.

BRA	Relative Branch
Syntax:	[<i>label</i>]BRA label [<i>label</i>]BRA \$+k
Operands:	-256 ≤ label - PC + 1 ≤ 255 -256 ≤ k ≤ 255
Operation:	$(PC) + 1 + k \rightarrow PC$
Status Affected:	None
Description:	Add the signed 9-bit literal 'k' to the PC. Since the PC will have incremented to fetch the next instruction, the new address will be PC + 1 + k. This instruction is a two-cycle instruction.

Syrilax.	[label] BRA \$+k
Operands:	$-256 \le label - PC + 1 \le 255$ $-256 \le k \le 255$
Operation:	$(PC) + 1 + k \rightarrow PC$
Status Affected:	None
Description:	Add the signed 9-bit literal 'k' to the PC. Since the PC will have incremented to fetch the next instruction, the new address will be PC + 1 + k. This instruction is a two-cycle instruction. This branch has a limited range.

BRW	Relative Branch with W
Syntax:	[label] BRW
Operands:	None
Operation:	$(PC) + (W) \rightarrow PC$
Status Affected:	None
Description:	Add the contents of W (unsigned) to the PC. Since the PC will have incremented to fetch the next instruction, the new address will be PC + 1 + (W). This instruction is a two-cycle instruction.

BSF	Bit Set f
Syntax:	[label] BSF f,b
Operands:	$0 \le f \le 127$ $0 \le b \le 7$
Operation:	$1 \rightarrow (f < b >)$
Status Affected:	None
Description:	Bit 'b' in register 'f' is set.

BTFSS	Bit Test f, Skip if Set
Syntax:	[label]BTFSS f,b
Operands:	0 ≤ f ≤ 127 0 ≤ b < 7
Operation:	skip if (f) = 1
Status Affected:	None
Description:	If bit 'b' in register 'f' is '0', the next instruction is executed. If bit 'b' is '1', then the next instruction is discarded and a NOP is executed instead, making this a 2-cycle instruction.

CALL	Call Subroutine
Syntax:	[/abe/] CALL k
Operands:	$0 \leq k \leq 2047$
Operation:	(PC)+ 1→ TOS, k → PC<10:0>, (PCLATH<6:3>) → PC<14:11>
Status Affected:	None
Description:	Call Subroutine. First, return address (PC + 1) is pushed onto the stack. The eleven-bit immediate address is loaded into PC bits <10:0>. The upper bits of the PC are loaded from PCLATH. CALL is a two-cycle instruction.

CLRWDT	Clear Watchdog Timer
Syntax:	[label] CLRWDT
Operands:	None
Operation:	00h → WDT 0 → WDT prescaler, 1 → $\overline{10}$ 1 → \overline{PD}
Status Affected:	TO, PD
Description:	CLRWDT instruction resets the Watchdog Timer. It also resets the prescaler of the WDT. Status bits $\overline{\text{TO}}$ and $\overline{\text{PD}}$ are set.

CALLW	Subroutine Call With W
Syntax:	[label] CALLW
Operands:	None
Operation:	$ \begin{aligned} & (PC) + 1 \rightarrow TOS, \\ & (W) \rightarrow PC < 7:0>, \\ & (PCLATH < 6:0>) \rightarrow PC < 14:8> \end{aligned} $
Status Affected:	None
Description:	Subroutine call with W. First, the return address (PC + 1) is pushed onto the return stack. Then, the contents of W is loaded into PC<7:0>, and the contents of PCLATH into

PC<14:8>. CALLW is a two-cycle

COMF	Complement f
Syntax:	[label] COMF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	$(\overline{f}) \rightarrow (destination)$
Status Affected:	Z
Description:	The contents of register 'f' are complemented. If 'd' is '0', the result is stored in W. If 'd' is '1', the result is stored back in register 'f'.

CLRF	Clear f
Syntax:	[label] CLRF f
Operands:	$0 \leq f \leq 127$
Operation:	$00h \rightarrow (f)$ $1 \rightarrow Z$
Status Affected:	Z
Description:	The contents of register 'f' are cleared and the Z bit is set.

instruction.

DECF	Decrement f
Syntax:	[label] DECF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(f) - 1 \rightarrow (destination)
Status Affected:	Z
Description:	Decrement register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f'.

CLRW	Clear W
Syntax:	[label] CLRW
Operands:	None
Operation:	$00h \to (W)$ $1 \to Z$
Status Affected:	Z
Description:	W register is cleared. Zero bit (Z) is set.

DECFSZ	Decrement f, Skip if 0
Syntax:	[label] DECFSZ f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(f) - 1 \rightarrow (destination); skip if result = 0
Status Affected:	None
Description:	The contents of register 'f' are decremented. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'. If the result is '1', the next instruction is executed. If the result is '0', then a NOP is executed instead, making it a 2-cycle instruction.

INCFSZ	Increment f, Skip if 0
Syntax:	[label] INCFSZ f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(f) + 1 \rightarrow (destination), skip if result = 0
Status Affected:	None
Description:	The contents of register 'f' are incremented. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'. If the result is '1', the next instruction is executed. If the result is '0', a NOP is executed instead, making it a 2-cycle instruction.

GOTO	Unconditional Branch
Syntax:	[label] GOTO k
Operands:	$0 \leq k \leq 2047$
Operation:	$k \rightarrow PC<10:0>$ PCLATH<6:3> \rightarrow PC<14:11>
Status Affected:	None
Description:	GOTO is an unconditional branch. The eleven-bit immediate value is loaded into PC bits <10:0>. The upper bits of PC are loaded from PCLATH<4:3>.

IORLW	Inclusive OR literal with W
Syntax:	[label] IORLW k
Operands:	$0 \leq k \leq 255$
Operation:	(W) .OR. $k \rightarrow (W)$
Status Affected:	Z
Description:	The contents of the W register are OR'ed with the eight-bit literal 'k'. The result is placed in the W register.

INCF	Increment f
Syntax:	[label] INCF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(f) + 1 \rightarrow (destination)
Status Affected:	Z
Description:	The contents of register 'f' are incremented. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.

IORWF	Inclusive OR W with f	
Syntax:	[label] IORWF f,d	
Operands:	$0 \le f \le 127$ $d \in [0,1]$	
Operation:	(W) .OR. (f) \rightarrow (destination)	
Status Affected:	Z	
Description:	Inclusive OR the W register with register 'f'. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.	

LSLF	Logical Left Shift	
Syntax:	[label] LSLF f {,d}	
Operands:	$0 \le f \le 127$ $d \in [0,1]$	
Operation:	$(f<7>) \rightarrow C$ $(f<6:0>) \rightarrow dest<7:1>$ $0 \rightarrow dest<0>$	
Status Affected:	C, Z	
Description:	The contents of register 'f' are shifted one bit to the left through the Carry flag. A '0' is shifted into the LSb. If 'd' is '0', the result is placed in W. If 'd' is '1', the result is stored back in register 'f'.	
	C register f ←0	

LSRF	Logical Right Shift
Syntax:	[label] LSRF f {,d}
Operands:	$0 \le f \le 127$ d $\in [0,1]$
Operation:	$0 \rightarrow \text{dest}<7>$ $(f<7:1>) \rightarrow \text{dest}<6:0>,$ $(f<0>) \rightarrow C,$
Status Affected:	C, Z
Description:	The contents of register 'f' are shifted one bit to the right through the Carry flag. A '0' is shifted into the MSb. If 'd' is '0', the result is placed in W. If 'd' is '1', the result is stored back in register 'f'.
	0 → register f C

MOVF	Move f
Syntax:	[label] MOVF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	$(f) \rightarrow (dest)$
Status Affected:	Z
Description:	The contents of register f is moved to a destination dependent upon the status of d. If d = 0, destination is W register. If d = 1, the destination is file register f itself. d = 1 is useful to test a file register since status flag Z is affected.
Words:	1
Cycles:	1
Example:	MOVF FSR, 0
	After Instruction W = value in FSR register Z = 1

MOVIW	Move INDFn to W
Syntax:	[/abe/] MOVIW ++FSRn [/abe/] MOVIWFSRn [/abe/] MOVIW FSRn++ [/abe/] MOVIW FSRn [/abe/] MOVIW k[FSRn]
Operands:	$n \in [0,1]$ $mm \in [00,01, 10, 11]$ $-32 \le k \le 31$
Operation:	INDFn → W Effective address is determined by • FSR + 1 (preincrement) • FSR - 1 (predecrement) • FSR + k (relative offset) After the Move, the FSR value will be either: • FSR + 1 (all increments) • FSR - 1 (all decrements) • Unchanged
Status Affected:	Z

Mode	Syntax	mm
Preincrement	++FSRn	00
Predecrement	FSRn	01
Postincrement	FSRn++	10
Postdecrement	FSRn	11

Description: This instruction is used to move data

between W and one of the indirect registers (INDFn). Before/after this move, the pointer (FSRn) is updated by pre/post incrementing/decrementing it.

Note: The INDFn registers are not physical registers. Any instruction that accesses an INDFn register actually accesses the register at the address specified by the FSRn.

FSRn is limited to the range 0000h -FFFFh. Incrementing/decrementing it beyond these bounds will cause it to wrap

around.

MOVLB Move literal to BSR

Syntax: [label] MOVLB k

 $\begin{tabular}{lll} Operands: & 0 \le k \le 15 \\ Operation: & k \to BSR \\ Status \ Affected: & None \\ \end{tabular}$

Description: The five-bit literal 'k' is loaded into the

Bank Select Register (BSR).

MOVLP Move literal to PCLATH	
------------------------------	--

Syntax: [label] MOVLP k

Operands: $0 \le k \le 127$ Operation: $k \to PCLATH$

Status Affected: None

Description: The seven-bit literal 'k' is loaded into the

PCLATH register.

MOVLW Move literal to W

Syntax: [label] MOVLW k Operands: $0 \le k \le 255$ Operation: $k \to (W)$

None

Description: The eight-bit literal 'k' is loaded into W

register. The "don't cares" will

assemble as '0's.

Words: 1
Cycles: 1

Status Affected:

Example: MOVLW 0x5A

After Instruction

W = 0x5A

MOVWF Move W to f

Syntax: [label] MOVWF f

Operands: $0 \le f \le 127$ Operation: $(W) \to (f)$ Status Affected: None

Description: Move data from W register to register

'f'.

Words: 1
Cycles: 1

Example: MOVWF OPTION REG

Before Instruction

OPTION_REG = 0xFF

W = 0x4F

After Instruction

OPTION_REG = 0x4F W = 0x4F

MOVWI	Move W to INDFn
Syntax:	[label] MOVWI ++FSRn [label] MOVWIFSRn [label] MOVWI FSRn++ [label] MOVWI FSRn [label] MOVWI k[FSRn]
Operands:	$n \in [0,1]$ $mm \in [00,01, 10, 11]$ $-32 \le k \le 31$
Operation:	 W → INDFn Effective address is determined by FSR + 1 (preincrement) FSR - 1 (predecrement) FSR + k (relative offset) After the Move, the FSR value will be either: FSR + 1 (all increments) FSR - 1 (all decrements) Unchanged
Status Affected:	None

Mode	Syntax	mm
Preincrement	++FSRn	00
Predecrement	FSRn	01
Postincrement	FSRn++	10
Postdecrement	FSRn	11

Description:

This instruction is used to move data between W and one of the indirect registers (INDFn). Before/after this move, the pointer (FSRn) is updated by pre/post incrementing/decrementing it.

Note: The INDFn registers are not physical registers. Any instruction that accesses an INDFn register actually accesses the register at the address specified by the FSRn.

FSRn is limited to the range 0000h - FFFFh. Incrementing/decrementing it beyond these bounds will cause it to wrap around.

The increment/decrement operation on FSRn WILL NOT affect any Status bits.

NOP	No Operation
Syntax:	[label] NOP
Operands:	None
Operation:	No operation
Status Affected:	None
Description:	No operation.
Words:	1
Cycles:	1
Example:	NOP
OPTION	Load OPTION_REG Register

OPTION	Load OPTION_REG Register with W
Syntax:	[label] OPTION
Operands:	None
Operation:	$(W) \to OPTION_REG$
Status Affected:	None
Description:	Move data from W register to OPTION_REG register.
Words:	1
Cycles:	1
Example:	OPTION
	Before Instruction OPTION_REG = 0xFF W = 0x4F After Instruction OPTION_REG = 0x4F W = 0x4F

RESET	Software Reset
Syntax:	[label] RESET
Operands:	None
Operation:	Execute a device Reset. Resets the nRI flag of the PCON register.
Status Affected:	None
Description:	This instruction provides a way to execute a hardware Reset by software.

RETFIE	Return from Interrupt					
Syntax:	[label] RETFIE					
Operands:	None					
Operation:	$TOS \rightarrow PC$, 1 \rightarrow GIE					
Status Affected:	None					
Description:	Return from Interrupt. Stack is POPed and Top-of-Stack (TOS) is loaded in the PC. Interrupts are enabled by setting Global Interrupt Enable bit, GIE (INTCON<7>). This is a two-cycle instruction.					
Words:	1					
Cycles:	2					
Example:	RETFIE					
	After Interrupt PC = TOS GIE = 1					

RETURN	Return from Subroutine			
Syntax:	[label] RETURN			
Operands:	None			
Operation:	$TOS \to PC$			
Status Affected:	None			
Description:	None Return from subroutine. The stack is POPed and the top of the stack (TOS) is loaded into the program counter. This is a two-cycle instruction.			

RETLW	Return with literal in W					
Syntax:	[<i>label</i>] RETLW k					
Operands:	$0 \leq k \leq 255$					
Operation:	$k \rightarrow (W);$ TOS \rightarrow PC					
Status Affected:	None					
Description:	The W register is loaded with the eight bit literal 'k'. The program counter is loaded from the top of the stack (the return address). This is a two-cycle instruction.					
Words:	1					
Cycles:	2					
Example:	CALL TABLE; W contains table ;offset value • ;W now has table value					
TABLE	. ADDWF PC ;W = offset RETLW k1 ;Begin table RETLW k2 ; RETLW kn ; End of table					
	Before Instruction W = 0x07					
	After Instruction					

W = value of k8

RLF	Rotate Left f through Carry					
Syntax:	[label] RLF f,d					
Operands:	$0 \le f \le 127$ $d \in [0,1]$					
Operation:	See description below					
Status Affected:	С					
Description:	The contents of register 'f' are rotated one bit to the left through the Carry flag. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is stored back in register 'f'.					
Manda.	1					
Words:	1					
Cycles:	1					
Example:	RLF REG1,0					
	Before Instruction					
	REG1 = 1110 0110					
	C = 0					
	After Instruction					
	REG1 = 1110 0110					
	$W = 1100 \ 1100$ C = 1					
	C _ 1					

RRF Rotate Right f through Carry

Syntax: [label] RRF f,d

Operands: $0 \le f \le 127$

 $d \in [0,1]$

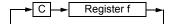
Operation: See description below

Status Affected: C

Description: The contents of register 'f' are rotated

one bit to the right through the Carry flag. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is

placed back in register 'f'.



SUBLW Subtract W from literal

Syntax: [label] SUBLW k

Operands: $0 \le k \le 255$ Operation: $k - (W) \rightarrow (W)$ Status Affected: C, DC, Z

Description: The W register is subtracted (2's

complement method) from the eight-bit literal 'k'. The result is placed in the W

register.

C = 0	W > k
C = 1	$W \leq k $
DC = 0	W<3:0> > k<3:0>
DC = 1	W<3:0> ≤ k<3:0>

SLEEP Enter Sleep mode

Syntax: [label] SLEEP

Operands: None

Operation: $00h \rightarrow WDT$,

 $0 \rightarrow WDT$ prescaler,

 $1 \to \frac{TO}{PD},$ $0 \to \overline{PD}$

Status Affected: TO, PD

Description: The power-down Status bit, \overline{PD} is

cleared. Time-out Status bit, $\overline{\text{TO}}$ is set. Watchdog Timer and its

prescaler are cleared.

The processor is put into Sleep mode

with the oscillator stopped.

SUBWF Subtract W from f

Syntax: [label] SUBWF f,d

Operands: $0 \le f \le 127$

 $d \in [0,1]$

Operation: (f) - (W) \rightarrow (destination)

Status Affected: C, DC, Z

Description: Subtract (2's complement method) W

register from register 'f'. If 'd' is '0', the

result is stored in the W

register. If 'd' is '1', the result is stored

back in register 'f.

C = 0	W > f
C = 1	$W \leq f$
DC = 0	W<3:0> > f<3:0>
DC = 1	W<3:0> ≤ f<3:0>

SUBWFB Subtract W from f with Borrow

Syntax: SUBWFB f {,d}

Operands: $0 \le f \le 127$

 $d \in [0,1]$

Operation: $(f) - (W) - (\overline{B}) \rightarrow dest$

Status Affected: C, DC, Z

Description: Subtract W and the BORROW flag

(CARRY) from register 'f' (2's complement method). If 'd' is '0', the result is stored in W. If 'd' is '1', the result is stored back in register 'f'.

SWAPF	Swap Nibbles in f					
Syntax:	[label] SWAPF f,d					
Operands:	$0 \le f \le 127$ $d \in [0,1]$					
Operation:	$(f<3:0>) \rightarrow (destination<7:4>),$ $(f<7:4>) \rightarrow (destination<3:0>)$					
Status Affected:	None					
Description:	None The upper and lower nibbles of register 'f' are exchanged. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed in register 'f'.					

TRIS	Load TRIS Register with W				
Syntax:	[label] TRIS f				
Operands:	$5 \leq f \leq 7$				
Operation:	$(W) \rightarrow TRIS register 'f'$				
Status Affected:	None				
Description:	Move data from W register to TRIS register. When 'f' = 5, TRISA is loaded. When 'f' = 6, TRISB is loaded. When 'f' = 7, TRISC is loaded.				

XORLW Exclusive OR literal with							
Syntax:	[label] XORLW k						
Operands:	$0 \leq k \leq 255$						
Operation:	(W) .XOR. $k \rightarrow (W)$						
Status Affected:	Z						
Description:	The contents of the W register are XOR'ed with the eight-bit literal 'k'. The result is placed in the W register.						

XORWF	Exclusive OR W with f					
Syntax:	[label] XORWF f,d					
Operands:	$0 \le f \le 127$ $d \in [0,1]$					
Operation:	(W) .XOR. (f) \rightarrow (destination)					
Status Affected:	Z					
Description:	Exclusive OR the contents of the W register with register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f'.					

30.0 ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings^(†)

Ambient temperature under bias40°C to +125°C				
Storage temperature65°C to +150°C				
Voltage on VDD with respect to Vss, PIC16F1825/18290.3V to +6.5V				
Voltage on VDD with respect to Vss, PIC16LF1825/18290.3V to +4.0V				
Voltage on MCLR with respect to Vss0.3V to +9.0V				
Voltage on all other pins with respect to Vss0.3V to (VDD + 0.3V)				
Total power dissipation ⁽¹⁾				
$\label{eq:maximum current out of Vss pin, -40°C lequal} Maximum current out of Vss pin, -40°C lequal +85°C for industrial. \\ 85 mA$				
Maximum current out of Vss pin, -40° C \leq TA \leq +125 $^{\circ}$ C for extended				
$\label{eq:maximum current into VDD pin, -40°C leq TA leq +85°C for industrial.} 80 \text{ mA}$				
Maximum current into VDD pin, -40° C \leq TA \leq +125 $^{\circ}$ C for extended				
Clamp current, IK (VPIN < 0 or VPIN > VDD) \pm 20 mA				
Maximum output current sunk by any I/O pin				
Maximum output current sourced by any I/O pin				
Note 1: Power dissipation is calculated as follows: PDIS = VDD x {IDD $- \sum IOH$ } + $\sum \{(VDD - VOH) \times IOH\} + \sum (VOI \times IOL)$.				

† NOTICE: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure above maximum rating conditions for extended periods may affect device reliability.

FIGURE 30-1: PIC16F1825/1829 VOLTAGE FREQUENCY GRAPH, -40°C ≤ TA ≤ +125°C

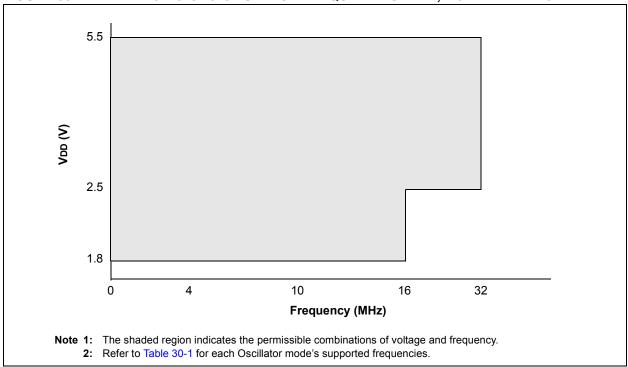
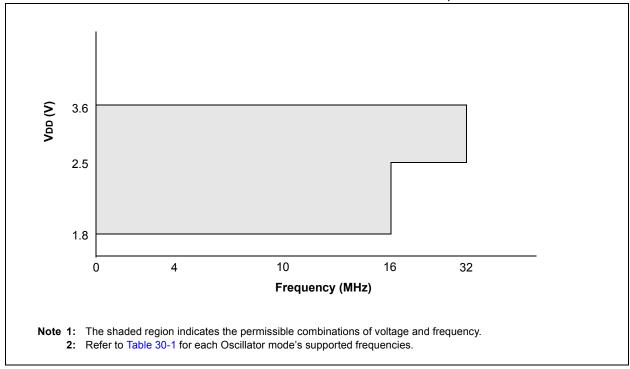
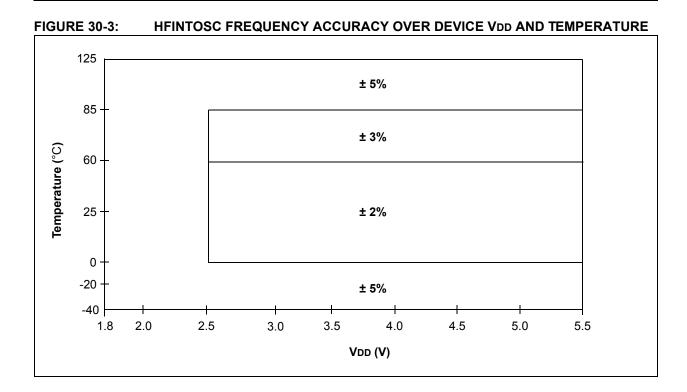


FIGURE 30-2: PIC16LF1825/1829 VOLTAGE FREQUENCY GRAPH, -40°C ≤ TA ≤ +125°C





30.1 DC Characteristics: PIC16(L)F1825/1829-I/E (Industrial, Extended)

PIC16LF1825/1829		Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{TA} \le +85^{\circ}\text{C}$ for industrial $-40^{\circ}\text{C} \le \text{TA} \le +125^{\circ}\text{C}$ for extended					
PIC16F1825/1829		Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{TA} \le +85^{\circ}\text{C}$ for industrial $-40^{\circ}\text{C} \le \text{TA} \le +125^{\circ}\text{C}$ for extended					
Param. No.	Sym.	Characteristic	Min.	Min. Typ† Max. Units Conditions			Conditions
D001	VDD	Supply Voltage (VDDMIN, VDDMAX)					
		PIC16LF1825/1829	1.8 2.5	_ _	3.6 3.6	V V	Fosc \leq 16 MHz: Fosc \leq 32 MHz (NOTE 2)
D001		PIC16F1825/1829	1.8 2.5		5.5 5.5	V V	FOSC \leq 16 MHz: FOSC \leq 32 MHz (NOTE 2)
D002*	VDR	RAM Data Retention Voltage ⁽¹⁾					
		PIC16LF1825/1829	1.5	—	_	V	Device in Sleep mode
D002*		PIC16F1825/1829	1.7	_	_	V	Device in Sleep mode
	VPOR*	Power-on Reset Release Voltage	_	1.6	_	V	
D002B*	VPORR*	Power-on Reset Rearm Voltage					
		PIC16LF1825/1829	_	8.0	_	V	Device in Sleep mode
		PIC16F1825/1829	_	1.7	_	V	Device in Sleep mode
D003	VADFVR	Fixed Voltage Reference Voltage for ADC	-8	_	6	%	$1.024V, VDD \ge 2.5V$ $2.048V, VDD \ge 2.5V$ $4.096V, VDD \ge 4.75V$
D003A	VCDAFVR	Fixed Voltage Reference Voltage for Comparator and DAC	-11	_	7	%	1.024V, VDD ≥ 2.5V 2.048V, VDD ≥ 2.5V 4.096V, VDD ≥ 4.75V
D003C*	TCVFVR	Temperature Coefficient, Fixed Voltage Reference		-130		ppm/°C	
D003D*	ΔVFVR/ ΔVIN	Line Regulation, Fixed Voltage Reference		0.270	_	%/V	
D004*	SVDD	VDD Rise Rate to ensure internal Power-on Reset signal	0.05	_	_	V/ms	See Section 7.1 "Power-on Reset (POR)" for details.

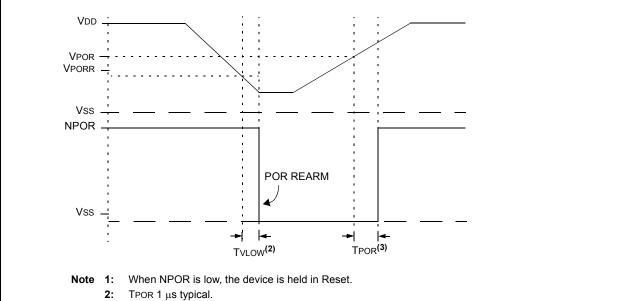
^{*} These parameters are characterized but not tested.

Note 1: This is the limit to which VDD can be lowered in Sleep mode without losing RAM data.

[†] Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

^{2:} PLL required for 32 MHz operation.





TVLOW 2.7 µs typical.

30.2 DC Characteristics: PIC16(L)F1825/1829-I/E (Industrial, Extended)

PIC16LF1	1825/1829	Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{Ta} \le +85^{\circ}\text{C}$ for industrial $-40^{\circ}\text{C} \le \text{Ta} \le +125^{\circ}\text{C}$ for extended							
PIC16F1825/1829				d Operati g tempera	ture -	less otherwise stated) A ≤ +85°C for industrial A ≤ +125°C for extended			
Param	Device	Min.	Tunt	Max.	Units	Conditions			
No.	Characteristics	IVIIII.	Typ†	IVIAX.	Units	VDD	Note		
	Supply Current (IDD) ^{(1,}	2)							
D010		_	5.5	15	μА	1.8	Fosc = 32 kHz		
		_	7.8	18	μА	3.0	LP Oscillator mode		
D010		_	20	55	μА	1.8	Fosc = 32 kHz		
		_	25	60	μΑ	3.0	LP Oscillator mode		
		_	27	65	μΑ	5.0			
D011		_	83	140	μΑ	1.8	Fosc = 1 MHz		
		_	130	230	μΑ	3.0	XT Oscillator mode		
D011		_	105	160	μА	1.8	Fosc = 1 MHz		
		_	160	250	μΑ	3.0	XT Oscillator mode		
		_	230	320	μΑ	5.0			
D012		_	220	310	μΑ	1.8	Fosc = 4 MHz		
		_	378	540	μΑ	3.0	XT Oscillator mode		
D012		_	240	300	μΑ	1.8	Fosc = 4 MHz		
		_	400	500	μΑ	3.0	XT Oscillator mode		
		_	500	760	μΑ	5.0			
D013		_	46	160	μΑ	1.8	FOSC = 1 MHz		
		_	90	230	μΑ	3.0	EC Oscillator mode, Medium-Power mode		
D013		_	70	180	μΑ	1.8	Fosc = 1 MHz		
		_	120	240	μΑ	3.0	EC Oscillator mode Medium-Power mode		
		_	190	320	μΑ	5.0			
D014		_	192	250	μΑ	1.8	FOSC = 4 MHz		
		_	336	430	μА	3.0	EC Oscillator mode, Medium-Power mode		
D014		_	210	275	μА	1.8	Fosc = 4 MHz		
			356	450	μА	3.0	EC Oscillator mode Medium-Power mode		
		_	430	650	μА	5.0			

- * These parameters are characterized but not tested.
- † Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.
- Note 1: The test conditions for all IDD measurements in active operation mode are: OSC1 = external square wave, from rail-to-rail; all I/O pins tri-stated, pulled to VDD; MCLR = VDD; WDT disabled.
 - 2: The supply current is mainly a function of the operating voltage and frequency. Other factors, such as I/O pin loading and switching rate, oscillator type, internal code execution pattern and temperature, also have an impact on the current consumption.
 - 3: 8 MHz internal RC oscillator with 4xPLL enabled.
 - **4:** 8 MHz crystal oscillator with 4xPLL enabled.
 - 5: For RC oscillator configurations, current through REXT is not included. The current through the resistor can be extended by the formula IR = VDD/2REXT (mA) with REXT in $k\Omega$..

30.2 DC Characteristics: PIC16(L)F1825/1829-I/E (Industrial, Extended) (Continued)

PIC16LF	1825/1829	Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{TA} \le +85^{\circ}\text{C}$ for industrial $-40^{\circ}\text{C} \le \text{TA} \le +125^{\circ}\text{C}$ for extended					
PIC16F18	PIC16F1825/1829			d Operati g tempera	iture -	40°C ≤ TA	less otherwise stated) $A \le +85^{\circ}\text{C for industrial}$ $A \le +125^{\circ}\text{C for extended}$
Param	Device	Min.	Tunt	Max.	1114-		Conditions
No.	Characteristics	IVIIII.	Тур†	IVIAX.	Units	VDD	Note
	Supply Current (IDD) ^{(1,}	2)					
D015		_	6.5	18	μΑ	1.8	Fosc = 31 kHz
		_	9.0	20	μΑ	3.0	LFINTOSC mode
D015		_	20	60	μΑ	1.8	Fosc = 31 kHz
		_	25	65	μΑ	3.0	LFINTOSC mode
		_	27	70	μΑ	5.0	
D016		_	110	170	μΑ	1.8	Fosc = 500 kHz
		_	130	200	μΑ	3.0	MFINTOSC mode
D016		_	125	180	μΑ	1.8	Fosc = 500 kHz
		_	155	250	μΑ	3.0	MFINTOSC mode
		_	160	280	μΑ	5.0	
D017*		_	0.6	0.85	mA	1.8	Fosc = 8 MHz
		_	0.9	1.25	mA	3.0	HFINTOSC mode
D017*		_	0.6	0.85	mA	1.8	Fosc = 8 MHz
		_	0.96	1.35	mA	3.0	HFINTOSC mode
		_	1.03	1.55	mA	5.0	
D018		_	0.9	1.2	mA	1.8	Fosc = 16 MHz
		_	1.4	1.95	mA	3.0	HFINTOSC mode
D018		_	0.92	1.2	mA	1.8	Fosc = 16 MHz
		_	1.49	1.9	mA	A 3.0	HFINTOSC mode
		_	1.58	2.4	mA	5.0	
D019		_	2.8	3.6	mA	3.0	Fosc = 32 MHz
		_	3.4	3.9	mA	3.6	HFINTOSC mode (Note 3)
D019		_	2.8	4.0	mA	3.0	Fosc = 32 MHz
			3.0	4.5	mA	5.0	HFINTOSC mode (Note 3)

- * These parameters are characterized but not tested.
- † Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: The test conditions for all IDD measurements in active operation mode are: OSC1 = external square wave, from rail-to-rail; all I/O pins tri-stated, pulled to VDD; MCLR = VDD; WDT disabled.

- 2: The supply current is mainly a function of the operating voltage and frequency. Other factors, such as I/O pin loading and switching rate, oscillator type, internal code execution pattern and temperature, also have an impact on the current consumption.
- 3: 8 MHz internal RC oscillator with 4xPLL enabled.
- 4: 8 MHz crystal oscillator with 4xPLL enabled.
- 5: For RC oscillator configurations, current through REXT is not included. The current through the resistor can be extended by the formula IR = VDD/2REXT (mA) with REXT in kΩ..

30.2 DC Characteristics: PIC16(L)F1825/1829-I/E (Industrial, Extended) (Continued)

PIC16LF1	1825/1829	Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{Ta} \le +85^{\circ}\text{C}$ for industrial $-40^{\circ}\text{C} \le \text{Ta} \le +125^{\circ}\text{C}$ for extended							
PIC16F18	325/1829		Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{TA} \le +85^{\circ}\text{C}$ for industrial $-40^{\circ}\text{C} \le \text{TA} \le +125^{\circ}\text{C}$ for extended						
Param	Device	N4:	T 4	Marr			Conditions		
No.	Characteristics	Min.	Typ†	Max.	Units	VDD	Note		
	Supply Current (IDD) ^(1, 2)								
D020	— 2.7 3.6 mA	mA	3.0	Fosc = 32 MHz					
		_	3.2	4.2	mA	3.6	HS Oscillator mode (Note 4)		
D020		_	2.7	4.0	mA	3.0	Fosc = 32 MHz		
		_	3.2	4.3	mA	5.0	HS Oscillator mode (Note 4)		
D021		_	222	350	μΑ	1.8	Fosc = 4 MHz		
		_	400	690	μΑ	3.0	EXTRC mode (Note 5)		
D021		_	240	500	μΑ	1.8	Fosc = 4 MHz		
		_	416	800	μΑ	3.0	EXTRC mode (Note 5)		
		_	497	900	μΑ	5.0			

^{*} These parameters are characterized but not tested.

- Note 1: The test conditions for all IDD measurements in active operation mode are: OSC1 = external square wave, from rail-to-rail; all I/O pins tri-stated, pulled to VDD; MCLR = VDD; WDT disabled.
 - 2: The supply current is mainly a function of the operating voltage and frequency. Other factors, such as I/O pin loading and switching rate, oscillator type, internal code execution pattern and temperature, also have an impact on the current consumption.
 - 3: 8 MHz internal RC oscillator with 4xPLL enabled.
 - 4: 8 MHz crystal oscillator with 4xPLL enabled.
 - 5: For RC oscillator configurations, current through Rext is not included. The current through the resistor can be extended by the formula IR = VDD/2REXT (mA) with REXT in $k\Omega$..

[†] Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested

30.3 DC Characteristics: PIC16(L)F1825/1829-I/E (Power-Down)

PIC16LF1825/1829				Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{Ta} \le +85^{\circ}\text{C}$ for industrial $-40^{\circ}\text{C} \le \text{Ta} \le +125^{\circ}\text{C}$ for extended						
PIC16F18	PIC16F1825/1829			Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{TA} \le +85^{\circ}\text{C}$ for industrial $-40^{\circ}\text{C} \le \text{TA} \le +125^{\circ}\text{C}$ for extended						
Param	Device Characteristics	Min.	Typ†	Max.	Max.	Units		Conditions		
No.	Dovido Gilaradionolido		.,,,,	+85°C	+125°C	011110	VDD	Note		
	Power-down Base Current	(IPD) ⁽²⁾								
D022		-	0.02	1.0	2.7	μΑ	1.8	WDT, BOR, FVR, and T1OSC		
		1	0.05	1.75	3.3	μΑ	3.0	disabled, all Peripherals Inactive		
D022		_	17.8	37	44	μА	1.8	WDT, BOR, FVR, and T1OSC		
			20.5	42	48	μА	3.0	disabled, all Peripherals Inactive		
		_	21.7	45	65	μА	5.0			
D023		1	0.3	1.5	3.0	μА	1.8	LPWDT Current (Note 1)		
		_	0.5	2.0	3.7	μΑ	3.0			
D023		1	18	38	44	μΑ	1.8	LPWDT Current (Note 1)		
		1	20.9	43	48	μΑ	3.0			
		_	22.1	48	65	μΑ	5.0			
D023A		_	12.6	22	25	μΑ	1.8	FVR current (Note 1)		
		_	12.8	24	27	μΑ	3.0			
D023A		_	32.7	62	65	μΑ	1.8	FVR current (Note 1)		
		-	39	72	75	μΑ	3.0			
		_	69	115	120	μΑ	5.0			
D024		_	7	14	16	μΑ	3.0	BOR Current (Note 1)		
D024		_	24	47	50	μА	3.0	BOR Current (Note 1)		
		_	27	55	70	μА	5.0			
D025		_	0.65	4.5	5	μА	1.8	T1OSC Current (Note 1)		
		_	1.3	5	5.5	μА	3.0			
D025		_	19	39	45	μА	1.8	T1OSC Current (Note 1)		
		_	21.6	42	65	μΑ	3.0			
		_	23	47	52	μА	5.0			
D026		_	0.05	1.5	3.0	μА	1.8	A/D Current (Note 1, 3), no		
		_	0.07	2.0	3.5	μА	3.0	conversion in progress		
D026			17.8	38	45	μА	1.8	A/D Current (Note 1, 3), no		
			21	45	50	μА	3.0	conversion in progress		
		_	22	50	65	μА	5.0			

^{*} These parameters are characterized but not tested.

- 2: The power-down current in Sleep mode does not depend on the oscillator type. Power-down current is measured with the part in Sleep mode, with all I/O pins in high-impedance state and tied to VDD.
- **3:** A/D oscillator source is FRC.

[†] Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested

Note 1: The peripheral current is the sum of the base IDD or IPD and the additional current consumed when this peripheral is enabled. The peripheral Δ current can be determined by subtracting the base IDD or IPD current from this limit. Max values should be used when calculating total current consumption.

30.3 DC Characteristics: PIC16(L)F1825/1829-I/E (Power-Down) (Continued)

PIC16LF1825/1829				Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{TA} \le +85^{\circ}\text{C}$ for industrial $-40^{\circ}\text{C} \le \text{TA} \le +125^{\circ}\text{C}$ for extended						
PIC16F1825/1829				Standard Operating Cond Operating temperature			litions (unless otherwise stated) $-40^{\circ}\text{C} \le \text{TA} \le +85^{\circ}\text{C}$ for industrial $-40^{\circ}\text{C} \le \text{TA} \le +125^{\circ}\text{C}$ for extended			
Param No.	Device Characteristics	Min.	Typ†	Max. +85°C	Max. +125°C	Units	\	Conditions		
	Power-down Base Current	(Ipp)(2)		• • •	0		VDD	Note		
D026A*	Power-down base current	(140), ,	250	l _	_	μА	1.8	A/D Current (Note 1, 3),		
DOZOA			250			μΑ	3.0	conversion in progress		
D026A*			280			μΑ	1.8	A/D Current (Note 1, 3),		
DOZOA			280			μΑ	3.0	conversion in progress		
			280	_	_	μΑ	5.0			
D027		_	2.3	5	6	μΑ	1.8	Cap Sense Low-Power		
		_	3.5	7	9	μА	3.0	CPSRM = 0, CPSRNG = 01 (Note 1)		
D027		I	20	41	45	μА	1.8	Cap Sense Low-Power		
		-	23	47	55	μА	3.0	CPSRM = 0, CPSRNG = 01 (Note 1)		
		1	25	55	75	μА	5.0	(Note 1)		
D027A		-	5	8	10	μА	1.8	Cap Sense Medium-Power		
		1	6.5	13	14	μА	3.0	CPSRM = 0, CPSRNG = 10 (Note 1)		
D027A		ı	23	44	47	μА	1.8	Cap Sense Medium-Power		
		ı	26	53	60	μА	3.0	CPSRM = 0, CPSRNG = 10 (Note 1)		
		_	29	57	71	μΑ	5.0	(Note 1)		
D027B		_	13	22	24	μА	1.8	Cap Sense High-Power		
		1	35	45	47	μА	3.0	CPSRM = 0, CPSRNG = 11 (Note 1)		
D027B		1	30	58	65	μА	1.8	Cap Sense High-Power		
		_	55	84	90	μΑ	3.0	CPSRM = 0, CPSRNG = 11 (Note 1)		
		_	59	95	110	μΑ	5.0	(Note 1)		
D028		1	6.8	16	17	μΑ	1.8	Comparator Current		
		_	7.3	18	19	μА	3.0	Low-Power mode (Note 1)		
D028		_	24	45	50	μА	1.8	Comparator Current		
		_	27	56	61	μΑ	3.0	Low-Power mode (Note 1)		
		_	29	60	80	μА	5.0			
D028B		_	28	46	48	μА	1.8	Comparator Current		
		_	29	48	49	μА	3.0	High-Power mode (Note 1)		
D028B		_	60	80	85	μΑ	1.8	Comparator Current,		
		_	62	85	90	μΑ	3.0	High-Power mode (Note 1)		
		_	64	90	105	μΑ	5.0			

These parameters are characterized but not tested.

- Note 1: The peripheral current is the sum of the base IDD or IPD and the additional current consumed when this peripheral is enabled. The peripheral Δ current can be determined by subtracting the base IDD or IPD current from this limit. Max values should be used when calculating total current consumption.
 - 2: The power-down current in Sleep mode does not depend on the oscillator type. Power-down current is measured with the part in Sleep mode, with all I/O pins in high-impedance state and tied to VDD.
 - 3: A/D oscillator source is FRC.

[†] Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

30.4 DC Characteristics: PIC16(L)F1825/1829-I/E

30.4		HARACTERISTICS	Standard Operating Conditions (unless otherwise stated) Operating temperature -40° C \leq TA \leq +85° C for industrial -40° C \leq TA \leq +125° C for extended							
Param	Sym.	Characteristic	Min.	Тур†	-40°C ≤ TA :	≤ +125° Units	C for extended Conditions			
No.	.,									
	VIL	Input Low Voltage								
D000		I/O PORT:			1 00	I	145144165455			
D030		with TTL buffer	_		0.8	V	4.5V ≤ VDD ≤ 5.5V			
D030A			_		0.15 VDD	V	1.8V ≤ VDD ≤ 4.5V			
D031		with Schmitt Trigger buffer	_		0.2 VDD	V	2.0V ≤ VDD ≤ 5.5V			
		with I ² C™ levels	_	_	0.3 VDD	V				
		with SMBus levels	_	_	0.8	V	2.7V ≤ VDD ≤ 5.5V			
D032		MCLR, OSC1 (RC mode) ⁽¹⁾	_		0.2 VDD	V				
D033		OSC1 (HS mode)	_	_	0.3 VDD	V				
	VIH	Input High Voltage								
İ		I/O ports:			T					
D040		with TTL buffer	2.0	_	_	V	$4.5V \le VDD \le 5.5V$			
D040A			0.25 VDD + 0.8	_	_	V	1.8V ≤ VDD ≤ 4.5V			
D041		with Schmitt Trigger buffer	0.8 VDD	_	_	V	$2.0V \le VDD \le 5.5V$			
		with I ² C™ levels	0.7 VDD	_	_	V				
		with SMBus levels	2.1	_	_	V	$2.7V \le VDD \le 5.5V$			
D042		MCLR	0.8 VDD	_	_	V				
D043A		OSC1 (HS mode)	0.7 VDD	_	_	V				
D043B		OSC1 (RC mode)	0.9 VDD	_	_	V	VDD > 2.0V (Note 1)			
	lıL	Input Leakage Current ⁽²⁾				•				
D060		I/O ports	_	± 5	± 125	nA	Vss ≤ VPIN ≤ VDD, Pin at high- impedance at 85°C			
				± 5	± 1000	nA	125°C			
D061		MCLR ⁽³⁾	_	± 50	± 200	nA	Vss ≤ Vpin ≤ Vdd at 85°C			
	IPUR	Weak Pull-up Current			•					
D070*			25	100	200		VDD = 3.3V, VPIN = VSS			
			25	140	300	μΑ	VDD = 5.0V, VPIN = VSS			
	VOL	Output Low Voltage ⁽⁴⁾								
D080		I/O ports	_	_	0.6	V	IOL = 8mA, VDD = 5V IOL = 6mA, VDD = 3.3V			
	1/2	0-44 111-1- 14(4)					IOL = 1.8mA, VDD = 1.8V			
Door	Vон	Output High Voltage ⁽⁴⁾	 		T	1	Lieu O Sm. A. Mar. 534			
D090		I/O ports	VDD - 0.7	_	_	V	IOH = 3.5mA, VDD = 5V IOH = 3mA, VDD = 3.3V IOH = 1mA, VDD = 1.8V			
		Capacitive Loading Specs on	Output Pins		1	l .				
D101*	COSC2	OSC2 pin	_	_	15	pF	In XT, HS and LP modes when external clock is used to drive OSC1			
D101A*	Сю	All I/O pins	_	_	50	pF				
		- F	l			L.	<u> </u>			

^{*} These parameters are characterized but not tested.

[†] Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: In RC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended to use an external clock in RC mode.

^{2:} Negative current is defined as current sourced by the pin.

^{3:} 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.

^{4:} Including OSC2 in CLKOUT mode.

30.5 Memory Programming Requirements

DC CHA	RACTER	RISTICS	Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{Ta} \le +125^{\circ}\text{C}$							
Param No.	Sym.	Characteristic	Min.	Тур†	Max.	Units	Conditions			
		Program Memory Programming Specifications								
D110	VIHH	Voltage on MCLR/VPP/RA5 pin	8.0	_	9.0	V	(Note 3, 4)			
D111	IDDP	Supply Current during Programming	_	_	10	mA				
D112	VBE	VDD for Bulk Erase	2.7	_	VDDMAX	V				
D113	VPEW	VDD for Write or Row Erase	VDDMIN	_	VDDMAX	V				
D114	IPPPGM	Current on MCLR/VPP during Erase/ Write	_	1.0	_	mA				
D115	IDDPGM	Current on VDD during Erase/Write	_	5.0	_	mA				
		Data EEPROM Memory								
D116	ED	Byte Endurance	100K	_	_	E/W	-40°C to +85°C			
D117	VDRW	VDD for Read/Write	VDDMIN	_	VDDMAX	V				
D118	TDEW	Erase/Write Cycle Time	_	4.0	5.0	ms				
D119	TRETD	Characteristic Retention	_	40	_	Year	Provided no other specifications are violated			
D120	TREF	Number of Total Erase/Write Cycles before Refresh ⁽²⁾	1M	10M	_	E/W	-40°C to +85°C			
		Program Flash Memory								
D121	EP	Cell Endurance	10K	_	-	E/W	-40°C to +85°C (Note 1)			
D122	VPRW	VDD for Read/Write	VDDMIN	_	VDDMAX	V				
D123	Tıw	Self-timed Write Cycle Time	_	2	2.5	ms				
D124	TRETD	Characteristic Retention	_	40	_	Year	Provided no other specifications are violated			

[†] Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

- Note 1: Self-write and Block Erase.
 - 2: Refer to Section 11.2 "Using the Data EEPROM" for a more detailed discussion on data EEPROM endurance.
 - **3:** Required only if single-supply programming is disabled.
 - 4: The MPLAB ICD 2 does not support variable VPP output. Circuitry to limit the MPLAB ICD 2 VPP voltage must be placed between the MPLAB ICD 2 and target system when programming or debugging with the MPLAB ICD 2.

30.6 Thermal Considerations

Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}C \le TA \le +125^{\circ}C$

Param No.	Sym.	Characteristic	Тур.	Units	Conditions
TH01	θЈА	Thermal Resistance Junction to Ambient	70.0	°C/W	14-pin PDIP package
			95.3	°C/W	14-pin SOIC package
			100	°C/W	14-pin TSSOP package
			45.7	°C/W	16-pin QFN (4x4) package
			62.2	°C/W	20-pin PDIP package
			77.7	°C/W	20-pin SOIC package
			87.3	°C/W	20-pin SSOP package
			43.0	°C/W	20-pin QFN (4x4) package
TH02	θJC	Thermal Resistance Junction to Case	32.8	°C/W	14-pin PDIP package
			31.0	°C/W	14-pin SOIC package
			24.4	°C/W	14-pin TSSOP package
			6.3	°C/W	16-pin QFN (4x4) package
			27.5	°C/W	20-pin PDIP package
			23.1	°C/W	20-pin SOIC package
			31.1	°C/W	20-pin SSOP package
			5.3	°C/W	20-pin QFN (4x4) package
TH03	ТЈМАХ	Maximum Junction Temperature	150	°C	
TH04	PD	Power Dissipation	_	W	PD = PINTERNAL + PI/O
TH05	PINTERNAL	Internal Power Dissipation	_	W	PINTERNAL = IDD x VDD ⁽¹⁾
TH06	Pı/o	I/O Power Dissipation	_	W	$PI/O = \Sigma (IOL * VOL) + \Sigma (IOH * (VDD - VOH))$
TH07	PDER	Derated Power	_	W	PDER = PDMAX (TJ - TA)/θJA ^(2,3)

Note 1: IDD is current to run the chip alone without driving any load on the output pins.

- 2: TA = Ambient Temperature
- **3**: T_J = Junction Temperature

30.7 Timing Parameter Symbology

The timing parameter symbols have been created with one of the following formats:

1. TppS2ppS

2. TppS

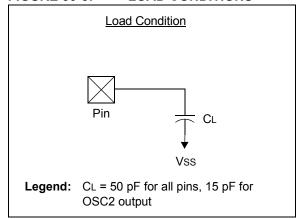
2. TPPO			
Т			
F	Frequency	T	Time
Lowerca	ase letters (pp) and their meanings:		
рр			
СС	CCP1	osc	OSC1
ck	CLKOUT	rd	RD
cs	CS	rw	RD or WR
di	SDIx	sc	SCKx
do	SDO1	SS	SS
dt	Data in	t0	T0CKI
io	I/O PORT	t1	T1CKI
mc	MCLR	wr	WR
Upperca	ase letters and their meanings:		
S			
F	Fall	Р	Period
Н	High	R	Rise
I	Invalid (High-impedance)	V	Valid

Ζ

High-impedance

FIGURE 30-5: LOAD CONDITIONS

Low



30.8 AC Characteristics: PIC16(L)F1825/1829-I/E

FIGURE 30-6: CLOCK TIMING

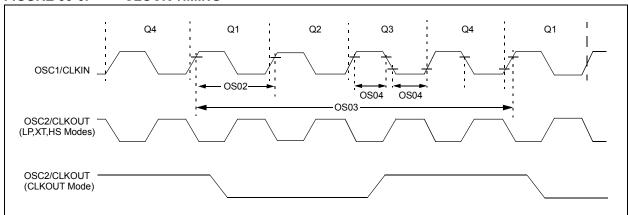


TABLE 30-1: CLOCK OSCILLATOR TIMING REQUIREMENTS

Standard Operating Conditions (unless otherwise stated)

Operatin	g tempera	ature -40°C ≤ Ta ≤ +125°C	1	1	1	1	
Param No.	Sym.	Characteristic	Min.	Тур†	Max.	Units	Conditions
OS01	Fosc	External CLKIN Frequency ⁽¹⁾	DC	_	0.5	MHz	EC Oscillator mode (low)
			DC	_	4	MHz	EC Oscillator mode (medium)
			DC	_	32	MHz	EC Oscillator mode (high)
		Oscillator Frequency ⁽¹⁾	_	32.768	_	kHz	LP Oscillator mode
			0.1	_	4	MHz	XT Oscillator mode
			1	_	4	MHz	HS Oscillator mode, VDD ≤ 2.7V
			1	_	20	MHz	HS Oscillator mode, VDD > 2.7V
			DC	_	4	MHz	RC Oscillator mode
OS02	Tosc	External CLKIN Period ⁽¹⁾	27	_	~	μS	LP Oscillator mode
			250	_	∞	ns	XT Oscillator mode
			50	_	∞	ns	HS Oscillator mode
			31.25	_	∞	ns	EC Oscillator mode
		Oscillator Period ⁽¹⁾	_	30.5	_	μS	LP Oscillator mode
			250	_	10,000	ns	XT Oscillator mode
			50	_	1,000	ns	HS Oscillator mode
			250	_	_	ns	RC Oscillator mode
OS03	TCY	Instruction Cycle Time ⁽¹⁾	125	_	DC	ns	Tcy = Fosc/4
OS04*	TosH,	External CLKIN High,	2	_	_	μS	LP oscillator
	TosL	External CLKIN Low	100	_	_	ns	XT oscillator
			20	_	_	ns	HS oscillator
OS05*	TosR,	External CLKIN Rise,	0	_	∞	ns	LP oscillator
	TosF	External CLKIN Fall	0	_	∞	ns	XT oscillator
			0	_	∞	ns	HS oscillator

These parameters are characterized but not tested.

[†] Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: Instruction cycle period (TcY) equals four 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 OSC1 pin. When an external clock input is used, the "max" cycle time limit is "DC" (no clock) for all devices.

TABLE 30-2: OSCILLATOR PARAMETERS

	d Operating Tempera	ng Conditions (unless otherwise stature $-40^{\circ}\text{C} \le \text{TA} \le +125^{\circ}\text{C}$	tated)					
Param No.	Sym.	Characteristic	Freq. Tolerance	Min.	Тур†	Max.	Units	Conditions
OS08	HFosc	Internal Calibrated HFINTOSC	±2%		16.0	_	MHz	$0^{\circ}C \le TA \le +60^{\circ}C, VDD \ge 2.5V$
		Frequency (Note 1)	±3%	_	16.0	_	MHz	$60^{\circ}C \leq TA \leq +85^{\circ}C, \ VDD \geq 2.5V$
			±5%	_	16.0	_	MHz	$-40^{\circ}C \le TA \le +125^{\circ}C$
OS08A	MFosc	Internal Calibrated MFINTOSC	±2%	_	500	_	kHz	$0^{\circ}C \leq TA \leq +60^{\circ}C, VDD \geq 2.5V$
		Frequency (Note 1)	±3%	_	500	_	kHz	$60^{\circ}C \le TA \le +85^{\circ}C, VDD \ge 2.5V$
			±5%	_	500	_	kHz	-40°C ≤ TA ≤ +125°C
OS09	LFosc	Internal LFINTOSC Frequency	±25%	_	31	_	kHz	-40°C ≤ TA ≤ +125°C
OS10*	Tiosc st	HFINTOSC Wake-up from Sleep Start-up Time	_	_	5	8	μS	
		MFINTOSC Wake-up from Sleep Start-up Time	_	_	20	30	μS	

^{*} These parameters are characterized but not tested.

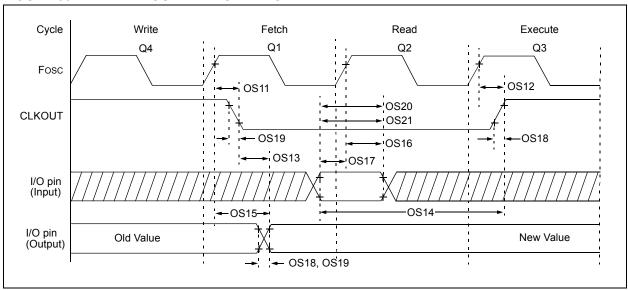
Note 1: To ensure these oscillator frequency tolerances, VDD and VSS must be capacitively decoupled as close to the device as possible. 0.1 μF and 0.01 μF values in parallel are recommended.

TABLE 30-3: PLL CLOCK TIMING SPECIFICATIONS (VDD = 2.7V TO 5.5V)

Param No.	Sym.	Characteristic	Min.	Typ†	Max.	Units	Conditions
F10	Fosc	Oscillator Frequency Range	4	_	8	MHz	
F11	Fsys	On-Chip VCO System Frequency	16	_	32	MHz	
F12	TRC	PLL Start-up Time (Lock Time)	_	_	2	ms	
F13*	Δ CLK	CLKOUT Stability (Jitter)	-0.25%	_	+0.25%	%	

^{*} These parameters are characterized but not tested.

FIGURE 30-7: CLKOUT AND I/O TIMING



[†] Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

[†] Data in "Typ" column is at 3V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

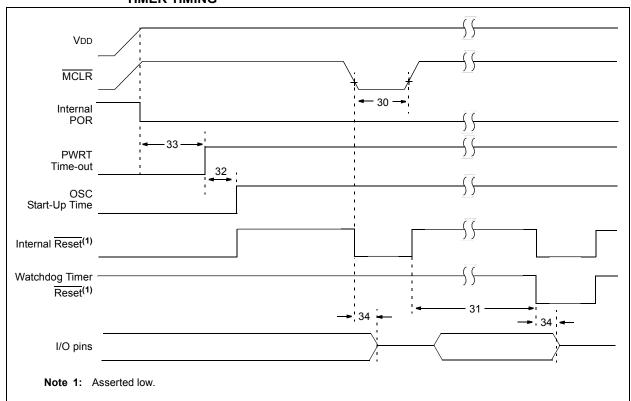
TABLE 30-4: CLKOUT AND I/O TIMING PARAMETERS

	•	g Conditions (unless otherwise stated) ure $-40^{\circ}C \le TA \le +125^{\circ}C$					
Param No.	Sym.	Characteristic	Min.	Тур†	Max.	Units	Conditions
OS11	TosH2ckL	Fosc↑ to CLKOUT↓ (1)	_	l	70	ns	VDD = 3.3-5.0V
OS12	TosH2ckH	Fosc↑ to CLKOUT↑ (1)	_		72	ns	VDD = 3.3-5.0V
OS13	TckL2ioV	CLKOUT↓ to Port out valid ⁽¹⁾	_		20	ns	
OS14	TioV2ckH	Port input valid before CLKOUT ⁽¹⁾	Tosc + 200 ns	_	_	ns	
OS15	TosH2ioV	Fosc↑ (Q1 cycle) to Port out valid	_	50	70*	ns	VDD = 3.3-5.0V
OS16	TosH2ioI	Fosc↑ (Q2 cycle) to Port input invalid (I/O in hold time)	50	_	_	ns	VDD = 3.3-5.0V
OS17	TioV2osH	Port input valid to Fosc↑ (Q2 cycle) (I/O in setup time)	20	_	_	ns	
OS18*	TioR	Port output rise time	_	40	72	ns	VDD = 1.8V
			_	15	32		VDD = 3.3-5.0V
OS19*	TioF	Port output fall time	_	28	55	ns	VDD = 1.8V
			_	15	30		VDD = 3.3-5.0V
OS20*	Tinp	INT pin input high or low time	25	1	_	ns	
OS21*	Tioc	Interrupt-on-change new input level time	25	_	_	ns	

^{*} These parameters are characterized but not tested.

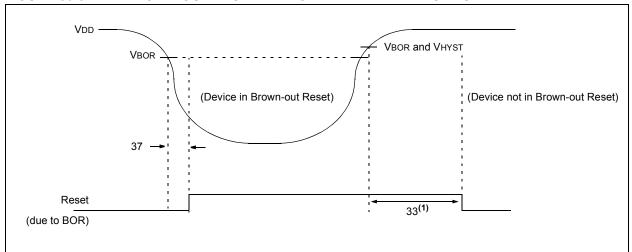
Note 1: Measurements are taken in RC mode where CLKOUT output is 4 x Tosc.

FIGURE 30-8: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING



[†] Data in "Typ" column is at 3.0V, 25°C unless otherwise stated.

FIGURE 30-9: BROWN-OUT RESET TIMING AND CHARACTERISTICS



Note 1: 64 ms delay only if PWRTE bit in the Configuration Word 1 is programmed to '0'.

2 ms delay if PWRTE = 0 and VREGEN = 1.

TABLE 30-5: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER, POWER-UP TIMER AND BROWN-OUT RESET PARAMETERS

Standard Operating Conditions (unless otherwise stated) Operating Temperature -40°C ≤ TA ≤ +125°C **Param** Sym. Characteristic Min. Typ† Max. Units Conditions No. 30 **TMCL** MCLR Pulse Width (low) 2 μS 31 **TWDTLP** Low-Power Watchdog Timer 12 16 20 ms VDD = 3.3V-5VTime-out Period 1:16 Prescaler used 32 Oscillator Start-up Timer Period⁽¹⁾ 1024 Tosc **Tost** 33* Power-up Timer Period, PWRTE = 0 **T**PWRT 40 65 140 ms 34* Tıoz I/O high-impedance from MCLR Low 2.0 μS or Watchdog Timer Reset **V**BOR Brown-out Reset Voltage⁽²⁾ 35 2.55 2.70 BORV = 0 2.85 ٧ BORV = 11.80 1.9 2.05 ٧ 36* **VHYST** Brown-out Reset Hysteresis 20 35 mV -40°C to +85°C 60 37* **TBORDC** Brown-out Reset DC Response 3 VDD ≤ VBOR 35 μS Time

- * These parameters are characterized but not tested.
- † Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.
- Note 1: By design, the Oscillator Start-up (OST) counts the first 1024 cycles, independent of frequency.
 - 2: To ensure these voltage tolerances, VDD and Vss must be capacitively decoupled as close to the device as possible. $0.1 \,\mu\text{F}$ and $0.01 \,\mu\text{F}$ values in parallel are recommended.

FIGURE 30-10: TIMERO AND TIMER1 EXTERNAL CLOCK TIMINGS

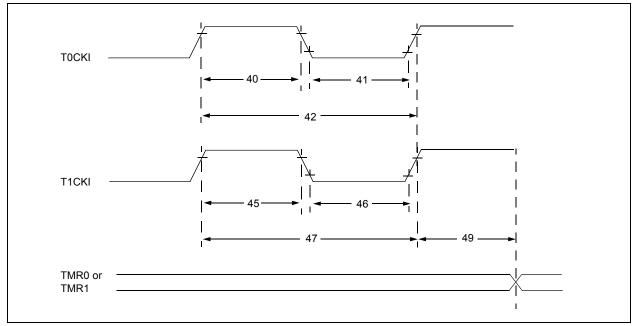


TABLE 30-6: TIMERO AND TIMER1 EXTERNAL CLOCK REQUIREMENTS

	rd Operating on the contract of the contract o		nless otherwis ≤ +125°C	e stated)					
Param No.	Sym.		Characteristic		Min.	Тур†	Max.	Units	Conditions
40*	Тт0Н	T0CKI High-F	Pulse Width	No Prescaler	0.5 Tcy + 20	_		ns	
				With Prescaler	10	_	_	ns	
41*	TT0L	T0CKI Low-F	ulse Width	No Prescaler	0.5 Tcy + 20	_	_	ns	
				With Prescaler	10	_	_	ns	
42*	Тт0Р	T0CKI Period	i		Greater of: 20 or TCY + 40 N		_	ns	N = prescale value (2, 4,, 256)
45*	TT1H	T1CKI High	Synchronous, N	No Prescaler	0.5 Tcy + 20	_	_	ns	
		Time	Synchronous, with Prescaler		15	_	_	ns	
			Asynchronous		30	_		ns	
46*	TT1L	T1CKI Low	Synchronous, N	No Prescaler	0.5 Tcy + 20	_	_	ns	
		Time	Synchronous, v	vith Prescaler	15	_	_	ns	
			Asynchronous		30	_	1	ns	
47*	Тт1Р	T1CKI Input Period	Synchronous		Greater of: 30 or TCY + 40 N			ns	N = prescale value (1, 2, 4, 8)
			Asynchronous		60	_		ns	
48	FT1		lator Input Frequency Range abled by setting bit T1OSCEN)		32.4	32.768	33.1	kHz	
49*	TCKEZTMR1	Delay from E Increment	xternal Clock Ed	dge to Timer	2 Tosc	_	7 Tosc	_	Timers in Sync mode

^{*} These parameters are characterized but not tested.

FIGURE 30-11: CAPTURE/COMPARE/PWM TIMINGS (CCP)

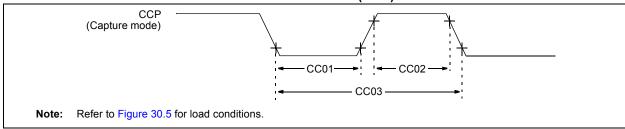


TABLE 30-7: CAPTURE/COMPARE/PWM REQUIREMENTS (CCP)

	Standard Operating Conditions (unless otherwise stated) Operating Temperature $-40^{\circ}\text{C} \le \text{TA} \le +125^{\circ}\text{C}$												
Param No.	Sym Characteristic Min Lynt Max Linits Conditions												
CC01*	* TccL CCP Input Low Time No Prescaler 0.5Tcy + 20 — ns												
			With Prescaler	20	_	_	ns						
CC02*	TccH	CCP Input High Time	No Prescaler	0.5Tcy + 20	_	_	ns						
	With Prescaler 20 — ns												
CC03*													

^{*} These parameters are characterized but not tested.

[†] Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

[†] Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

TABLE 30-8: PIC16(L)F1825/1829 A/D CONVERTER (ADC) CHARACTERISTICS^(1,2,3)

	Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{TA} \le +125^{\circ}\text{C}$											
Param No.	Sym.	Characteristic	Min.	Typ†	Max.	Units	Conditions					
AD01	NR	Resolution	_	_	10	bit						
AD02	EIL	Integral Error	_	_	±1.7	LSb	VREF = 3.0V					
AD03	EDL	Differential Error	_	_	±1	LSb	No missing codes VREF = 3.0V					
AD04	Eoff	Offset Error	_	_	±2.5	LSb	VREF = 3.0V					
AD05	Egn	Gain Error	_	_	±2.0	LSb	VREF = 3.0V					
AD06	VREF	Reference Voltage ⁽⁴⁾	1.8	_	VDD	V	VREF = (VREF+ minus VREF-)					
AD07	VAIN	Full-Scale Range	Vss	_	VREF	V						
AD08	ZAIN	Recommended Impedance of Analog Voltage Source		-	10	kΩ	Can go higher if external 0.01μF capacitor is present on input pin.					

[†] Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

- Note 1: Total Absolute Error includes integral, differential, offset and gain errors.
 - 2: The A/D conversion result never decreases with an increase in the input voltage and has no missing codes.
 - 3: ADC VREF is from external VREF, VDD pin or FVR, whichever is selected as reference input.
 - **4:** ADC Reference Voltage (REF+) is the selected input, VREF+ pin, VDD pin or the FVR Buffer1. When the FVR is selected as the reference input, the FVR Buffer1 output selection must be 2.048 or 4.096V (ADFVR<1:0> = 1x).

TABLE 30-9: PIC16(L)F1825/1829 A/D CONVERSION REQUIREMENTS

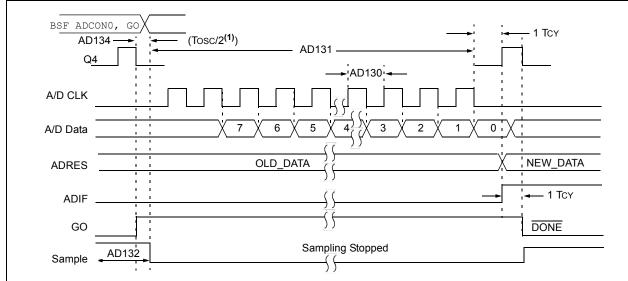
	Standard Operating Conditions (unless otherwise stated) Departing temperature -40°C ≤ TA ≤ +125°C										
Param No.	Sym.	Characteristic	Min.	Typ†	Max.	Units	Conditions				
AD130*	TAD	A/D Clock Period	1.0		9.0	μS	Tosc-based				
		A/D Internal RC Oscillator Period	1.0	2.5	6.0	μS	ADCS<1:0> = 11 (ADRC mode)				
AD131	TCNV	Conversion Time (not including Acquisition Time) ⁽¹⁾	_	11	_	TAD	Set GO/DONE bit to conversion complete				
AD132*	TACQ	Acquisition Time	_	5.0	_	μS					

^{*} These parameters are characterized but not tested.

Note 1: The ADRES register may be read on the following TcY cycle.

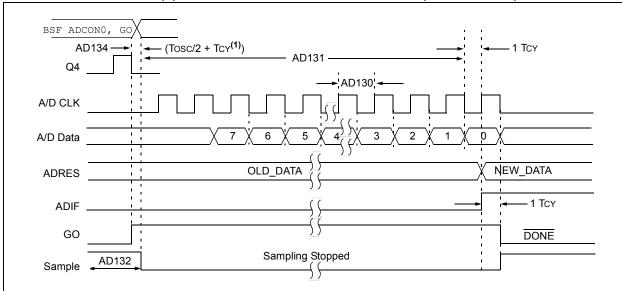
[†] Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

FIGURE 30-12: PIC16(L)F1825/1829 A/D CONVERSION TIMING (NORMAL MODE)



Note 1: If the A/D clock source is selected as RC, a time of Tcy is added before the A/D clock starts. This allows the SLEEP instruction to be executed.

FIGURE 30-13: PIC16(L)F1825/1829 A/D CONVERSION TIMING (SLEEP MODE)



Note 1: If the A/D clock source is selected as RC, a time of Tcy is added before the A/D clock starts. This allows the SLEEP instruction to be executed.

TABLE 30-10: COMPARATOR SPECIFICATIONS

Operating	Operating Conditions: 1.8V < VDD < 5.5V, -40°C < TA < +125°C (unless otherwise stated).									
Param No.	Sym.	Characteristics	Min.	Тур.	Max.	Units	Comments			
CM01	VIOFF	Input Offset Voltage ⁽¹⁾	_	±7.5	±60	mV	High-Power mode VICM = VDD/2			
CM02	VICM	Input Common Mode Voltage	0	_	VDD	V				
CM03	CMRR	Common Mode Rejection Ratio	_	50	_	dB				
CM04A		Response Time Rising Edge	_	400	800	ns	High-Power mode			
CM04B	TRESP ⁽¹⁾	Response Time Falling Edge	_	200	400	ns	High-Power mode			
CM04C	I RESP.	Response Time Rising Edge	_	1200	_	ns	Low-Power mode			
CM04D		Response Time Falling Edge	_	550	_	ns	Low-Power mode			
CM05	Тмс2о∨	Comparator Mode Change to Output Valid*	_	_	10	μS				
CM06	CHYSTER	Comparator Hysteresis ⁽²⁾	_	65		mV	CxHYS = 1			

Note 1: High-Power mode only.

TABLE 30-11: DIGITAL-TO-ANALOG CONVERTER (DAC) SPECIFICATIONS

Operating	Operating Conditions: 1.8V < VDD < 5.5V, -40°C < TA < +125°C (unless otherwise stated).										
Param No.	Sym. Characteristics Min. Typ. Max. Units Comments										
DAC01*	CLSB	Step Size ⁽²⁾	_	VDD/32	_	V					
DAC02*	CACC	Absolute Accuracy	_	_	± 1/2	LSb					
DAC03*	CR	Unit Resistor Value (R)	_	5K	_	Ω					
DAC04*	CsT	Settling Time ⁽¹⁾	_	_	10	μS					

^{*} These parameters are characterized but not tested.

Note 1: Settling time measured while DACR<4:0> transitions from '0000' to '1111'.

TABLE 30-12: PIC16(L)F1825/1829 LOW DROPOUT (LDO) REGULATOR CHARACTERISTICS:

	Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \le \text{TA} \le +125^{\circ}\text{C}$										
Param No.	Sym Characteristic Min Ivn+ May Hnits Conditions										
LD001	LD001 LD0 Regulation Voltage — 3.2 — V										
LD002	LD002 LDO External Capacitor 0.1 — 1 μF										

[†] Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

^{2:} Comparator Hysteresis is available when the CxHYS bit of the CMxCON0 register is enabled.

FIGURE 30-14: USART SYNCHRONOUS TRANSMISSION (MASTER/SLAVE) TIMING

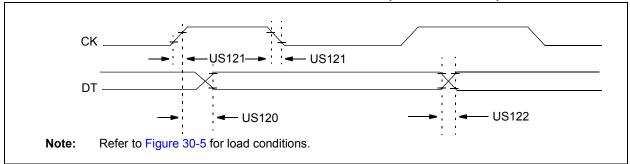


TABLE 30-13: USART SYNCHRONOUS TRANSMISSION REQUIREMENTS

	Standard Operating Conditions (unless otherwise stated) Operating Temperature $-40^{\circ}\text{C} \le \text{TA} \le +125^{\circ}\text{C}$											
Param. No. Symbol Characteristic Min. Max. Units Condition												
US120	TCKH2DTV	SYNC XMIT (Master and Slave)	3.0-5.5V	_	80	ns						
		Clock high to data-out valid	1.8-5.5V	_	100	ns						
US121	TCKRF	Clock out rise time and fall time	3.0-5.5V	_	45	ns						
		(Master mode)	1.8-5.5V	_	50	ns						
US122	TDTRF	Data-out rise time and fall time	3.0-5.5V	_	45	ns						
			1.8-5.5V	_	50	ns						

FIGURE 30-15: USART SYNCHRONOUS RECEIVE (MASTER/SLAVE) TIMING

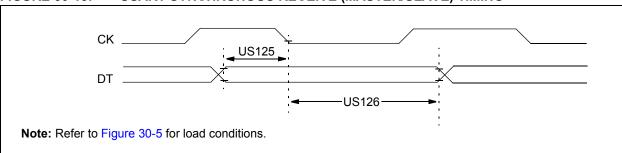


TABLE 30-14: USART SYNCHRONOUS RECEIVE REQUIREMENTS

Standard Operating Conditions (unless otherwise stated) Operating Temperature $-40^{\circ}\text{C} \le \text{TA} \le +125^{\circ}\text{C}$									
Param. No.	Symbol	Characteristic	Min.	Max.	Units	Conditions			
US125	TDTV2CKL	SYNC RCV (Master and Slave) Data-hold before CK ↓ (DT hold time)	10	_	ns				
US126	TCKL2DTL	Data-hold after CK ↓ (DT hold time)	15	_	ns				

FIGURE 30-16: SPI MASTER MODE TIMING (CKE = 0, SMP = 0)

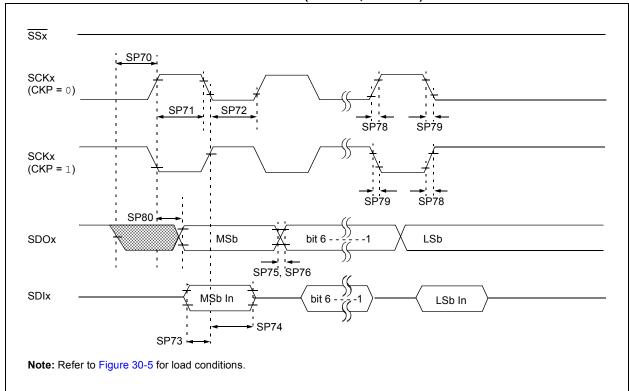


FIGURE 30-17: SPI MASTER MODE TIMING (CKE = 1, SMP = 1)

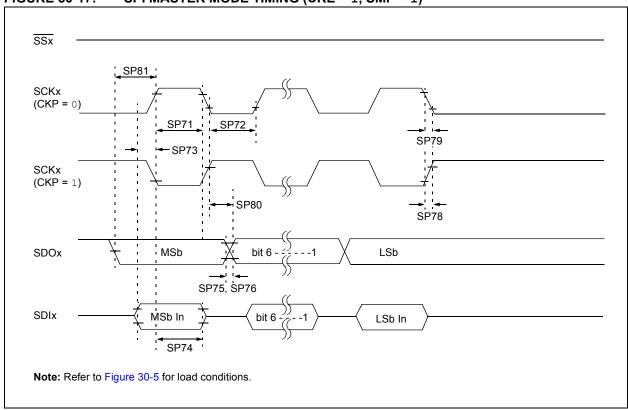


FIGURE 30-18: SPI SLAVE MODE TIMING (CKE = 0)

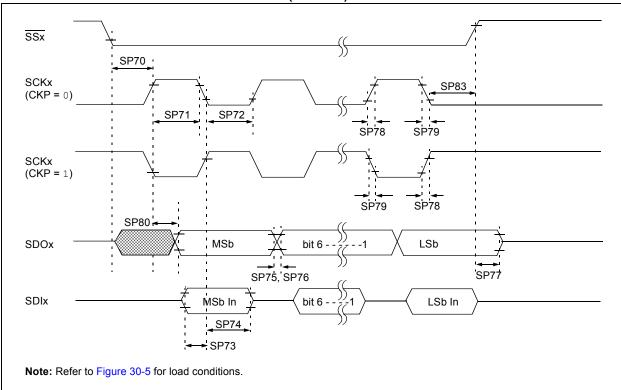


FIGURE 30-19: SPI SLAVE MODE TIMING (CKE = 1)

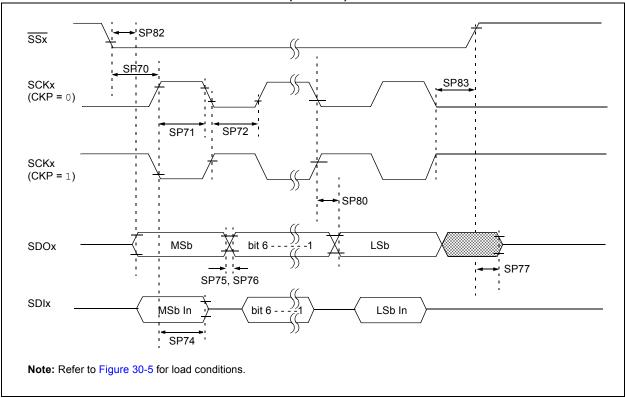


TABLE 30-15: SPI MODE REQUIREMENTS

Param No.	Symbol	Characteristic	Min.	Тур†	Max.	Units	Conditions	
SP70*	TssL2scH, TssL2scL	SSx↓ to SCKx↓ or SCKx↑ input	Tcy	_	_	ns		
SP71*	TscH	SCKx input high time (Slave mod	de)	Tcy + 20	_	_	ns	
SP72*	TscL	SCKx input low time (Slave mod	e)	Tcy + 20	_	_	ns	
SP73*	TDIV2scH, TDIV2scL	Setup time of SDIx data input to	100	_	_	ns		
SP74*	TscH2DIL, TscL2DIL	Hold time of SDIx data input to S	100	_	_	ns		
SP75*	TDOR	SDOx data output rise time	3.0-5.5V	_	10	25	ns	
			1.8-5.5V	_	25	50	ns	
SP76*	TDOF	SDOx data output fall time		_	10	25	ns	
SP77*	TssH2DoZ	SSx↑ to SDOx output high-impe	10	_	50	ns		
SP78*	TscR	SCKx output rise time	3.0-5.5V	_	10	25	ns	
		(Master mode)	1.8-5.5V	_	25	50	ns	
SP79*	TscF	SCKx output fall time (Master mo	ode)	_	10	25	ns	
SP80*	TscH2DoV,	SDOx data output valid after	3.0-5.5V	_	_	50	ns	
	TscL2doV	SCKx edge 1.8-5.5\		_	_	145	ns	
SP81*	TDOV2scH, TDOV2scL	SDOx data output setup to SCKx edge		Tcy	_	_	ns	
SP82*	TssL2DoV	SDOx data output valid after SS	_	_	50	ns		
SP83*	TscH2ssH, TscL2ssH	SSx ↑ after SCKx edge	1.5Tcy + 40		_	ns		

These parameters are characterized but not tested.

[†] Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

FIGURE 30-20: I²C™ BUS START/STOP BITS TIMING

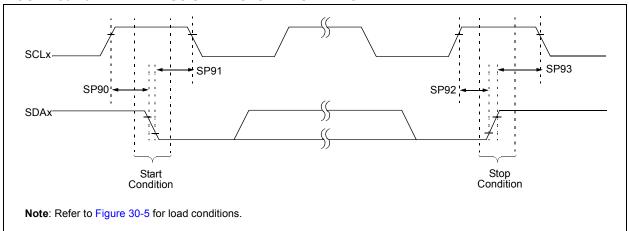


TABLE 30-16: I²C™ BUS START/STOP BITS REQUIREMENTS

Param No.	Symbol	Characteristic		Min.	Тур	Max.	Units	Conditions	
SP90*	Tsu:sta	Start condition	100 kHz mode	4700	_	_	ns	Only relevant for Repeated	
		Setup time	400 kHz mode	600	_	_		Start condition	
SP91*	THD:STA	Start condition	100 kHz mode	4000	_	_	ns	After this period, the first	
		Hold time	400 kHz mode	600	_	_		clock pulse is generated	
SP92*	Tsu:sto	Stop condition	100 kHz mode	4700	_	_	ns		
		Setup time	400 kHz mode	600	_	_			
SP93	THD:STO	Stop condition	100 kHz mode	4000		_	ns		
		Hold time	400 kHz mode	600		_			

^{*} These parameters are characterized but not tested.

FIGURE 30-21: I²C™ BUS DATA TIMING

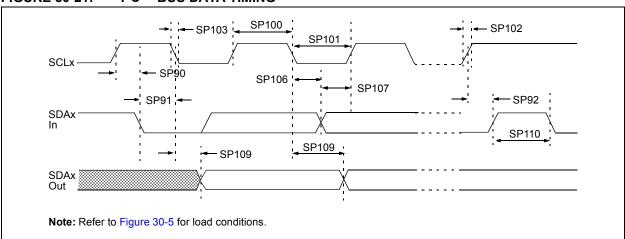


TABLE 30-17: I²C™ BUS DATA REQUIREMENTS

Param. No.	Symbol	Characteristic		Min.	Max.	Units	Conditions
SP100*	THIGH	Clock high time	100 kHz mode	4.0	1	μ\$	Device must operate at a minimum of 1.5 MHz
			400 kHz mode	0.6	1	μS	Device must operate at a minimum of 10 MHz
			SSPx module	1.5Tcy	I		
SP101*	TLOW	Clock low time	100 kHz mode	4.7	1	μ\$	Device must operate at a minimum of 1.5 MHz
			400 kHz mode	1.3	1	μ\$	Device must operate at a minimum of 10 MHz
			SSPx module	1.5Tcy		_	
SP102*	TR	SDAx and SCLx	100 kHz mode	_	1000	ns	
		rise time	400 kHz mode	20 + 0.1CB	300	ns	CB is specified to be from 10-400 pF
SP103*	TF	SDAx and SCLx fall	100 kHz mode	_	250	ns	
		time	400 kHz mode	20 + 0.1CB	250	ns	CB is specified to be from 10-400 pF
SP106*	THD:DAT	Data input hold time	100 kHz mode	0	_	ns	
			400 kHz mode	0	0.9	μS	
SP107*	TSU:DAT	Data input setup	100 kHz mode	250	_	ns	(Note 2)
		time	400 kHz mode	100	_	ns	
SP109*	TAA	Output valid from	100 kHz mode	_	3500	ns	(Note 1)
		clock	400 kHz mode	_	_	ns	
SP110*	TBUF	Bus free time	100 kHz mode	4.7		μS	Time the bus must be free
			400 kHz mode	1.3	_	μS	before a new transmission can start
SP111	Св	Bus capacitive loadir	ng	_	400	pF	

^{*} These parameters are characterized but not tested.

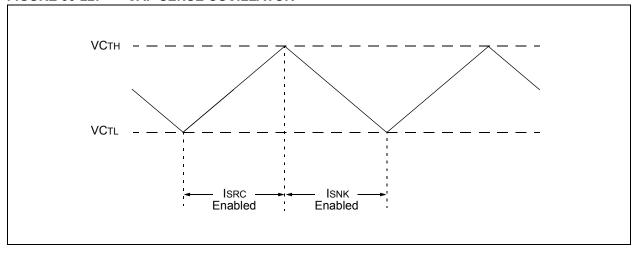
- **Note 1:** As a transmitter, the device must provide this internal minimum delay time to bridge the undefined region (min. 300 ns) of the falling edge of SCLx to avoid unintended generation of Start or Stop conditions.
 - 2: A Fast mode (400 kHz) I²C™ bus device can be used in a Standard mode (100 kHz) I²C bus system, but the requirement Tsu:DAT ≥ 250 ns must then be met. This will automatically be the case if the device does not stretch the low period of the SCLx signal. If such a device does stretch the low period of the SCLx signal, it must output the next data bit to the SDAx line TR max. + Tsu:DAT = 1000 + 250 = 1250 ns (according to the Standard mode I²C bus specification), before the SCLx line is released.

TABLE 30-18: CAP SENSE OSCILLATOR SPECIFICATIONS

Param. No.	Symbol	Characteristic		Min.	Typ†	Max.	Units	Conditions
CS01	ISRC	Current Source	High	_	-8	_	μΑ	
			Medium	_	-1.5	_	μА	
			Low	_	-0.3	_	μА	
CS02	ISNK	Current Sink	High	_	7.5	_	μА	
			Medium	_	1.5	_	μА	
			Low	_	0.25	_	μА	
CS03	VСтн	Cap Threshold		_	0.8	_	V	
CS04	VCTL	Cap Threshold		_	0.4	_	V	
CS05	VCHYST	CAP HYSTERESIS	High	_	525	_	mV	
		(VCTH-VCTL)	Medium	_	375	_	mV	
			Low	_	300	_	mV	

[†] Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

FIGURE 30-22: CAP SENSE OSCILLATOR



31.0 DC AND AC CHARACTERISTICS GRAPHS AND CHARTS

The graphs and tables provided in this section are for design guidance and are not tested.

In some graphs or tables, the data presented are **outside specified operating range** (i.e., outside specified VDD range). This is for **information only** and devices are ensured to operate properly only within the specified range.

Note: The graphs and tables provided following this note are a statistical summary based on a limited number of samples and are provided for informational purposes only. The performance characteristics listed herein are not tested or guaranteed. In some graphs or tables, the data presented may be outside the specified operating range (e.g., outside specified power supply range) and therefore, outside the warranted range.

"Typical" represents the mean of the distribution at 25°C. "MAXIMUM", "Max.", "MINIMUM" or "Min." represents (mean + 3σ) or (mean - 3σ) respectively, where σ is a standard deviation, over each temperature range.

FIGURE 31-1: IDD, LP OSCILLATOR MODE (Fosc = 32 kHz), PIC16LF1825/1829 ONLY

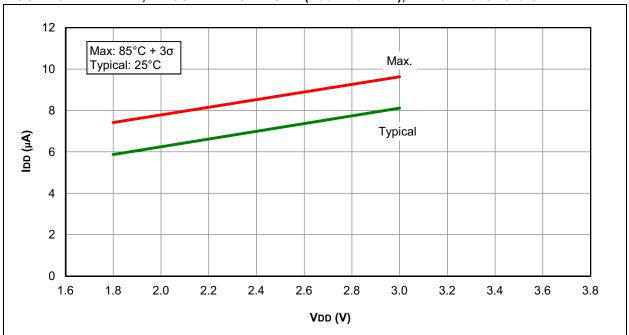
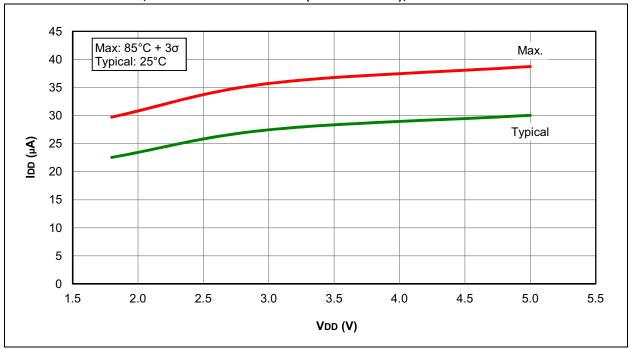
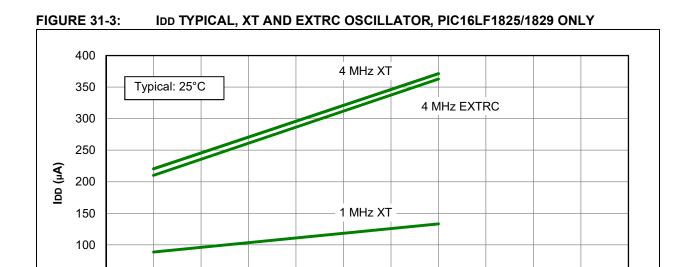


FIGURE 31-2: IDD, LP OSCILLATOR MODE (Fosc = 32 kHz), PIC16F1825/1829 ONLY





2.6

2.8

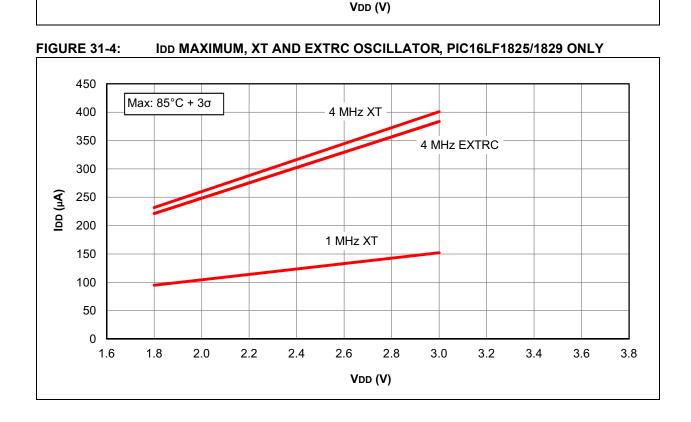
3.0

3.2

3.4

3.6

3.8



50

1.6

1.8

2.0

2.2

2.4

FIGURE 31-5: IDD TYPICAL, XT AND EXTRC OSCILLATOR, PIC16F1825/1829 ONLY

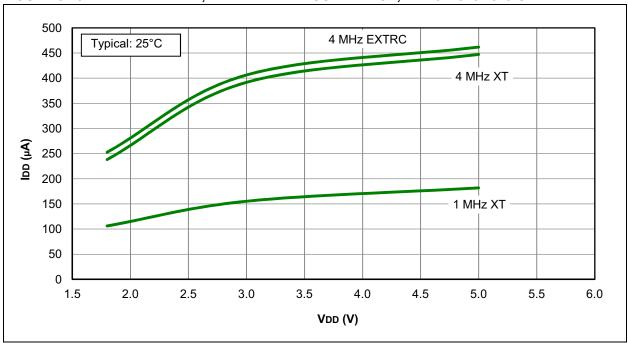


FIGURE 31-6: IDD MAXIMUM, XT AND EXTRC OSCILLATOR, PIC16F1825/1829 ONLY

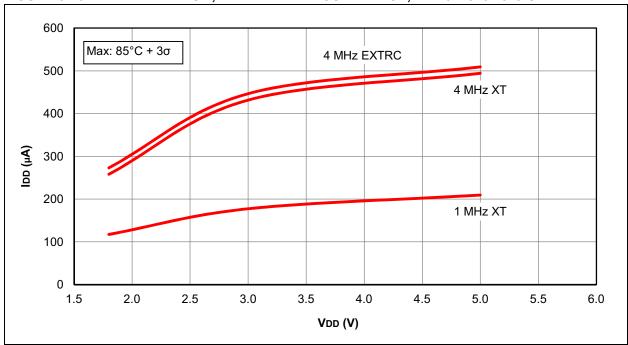


FIGURE 31-7: IDD TYPICAL, EC OSCILLATOR, MEDIUM-POWER MODE, PIC16LF1825/1829 ONLY

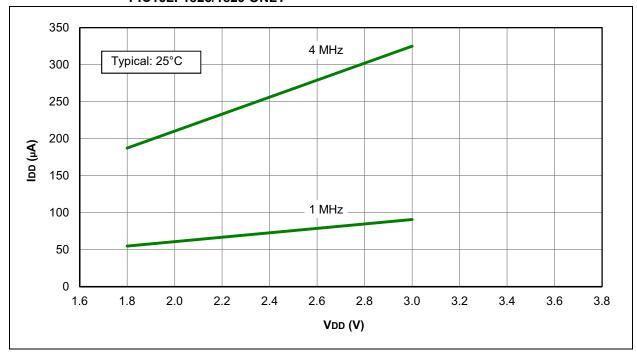


FIGURE 31-8: IDD MAXIMUM, EC OSCILLATOR, MEDIUM-POWER MODE, PIC16LF1825/1829 ONLY

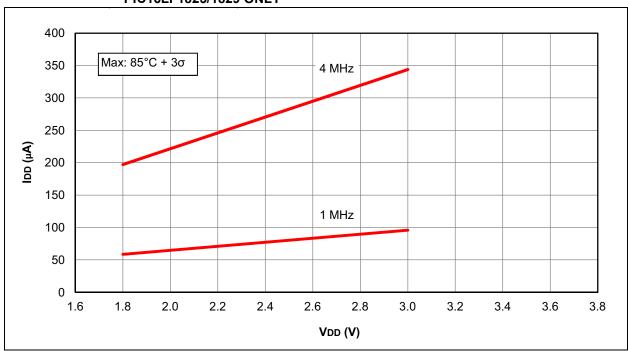


FIGURE 31-9: IDD TYPICAL, EC OSCILLATOR, MEDIUM-POWER MODE, PIC16F1825/1829 ONLY

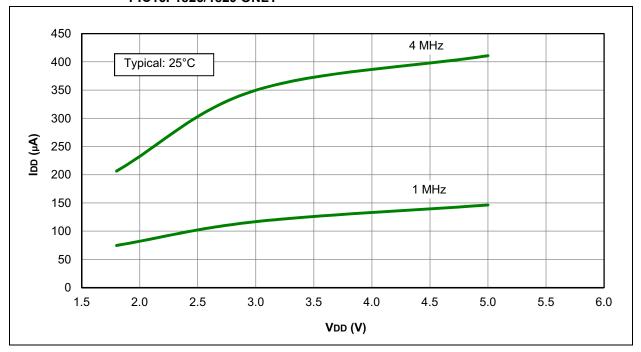
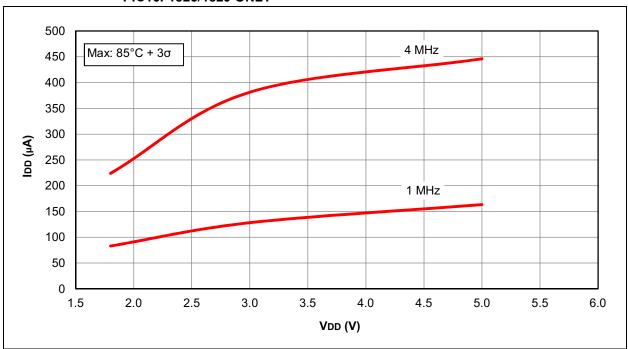
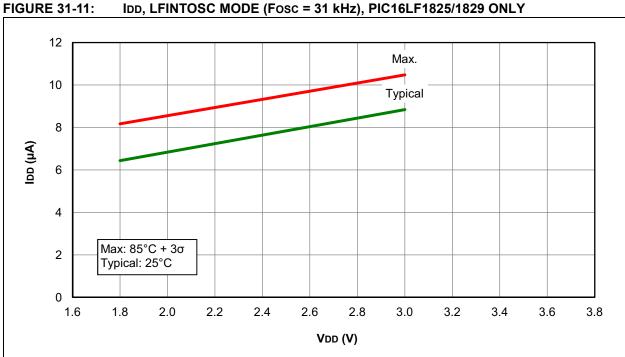
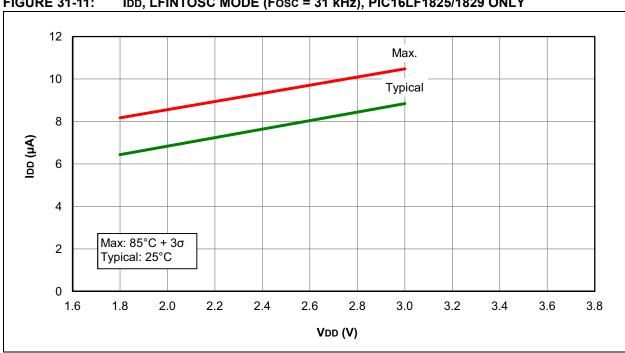


FIGURE 31-10: IDD MAXIMUM, EC OSCILLATOR, MEDIUM-POWER MODE, PIC16F1825/1829 ONLY







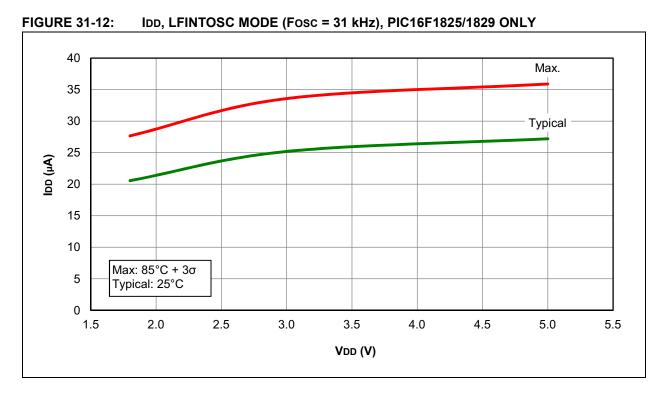


FIGURE 31-13: IDD, MFINTOSC MODE (Fosc = 500 kHz), PIC16LF1825/1829 ONLY

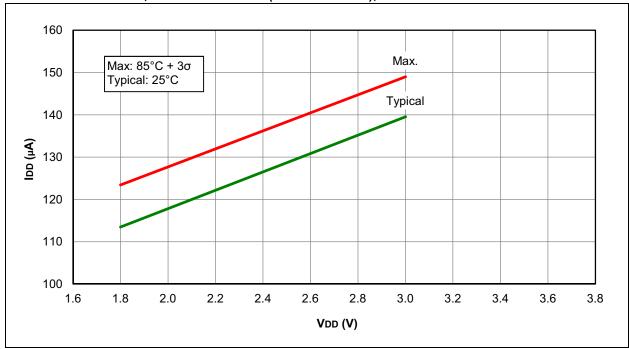
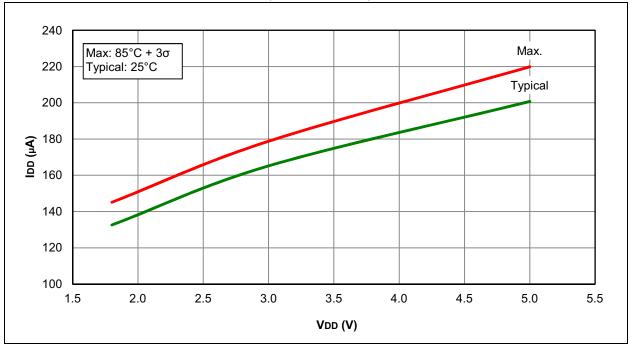


FIGURE 31-14: IDD, MFINTOSC MODE (Fosc = 500 kHz), PIC16F1825/1829 ONLY





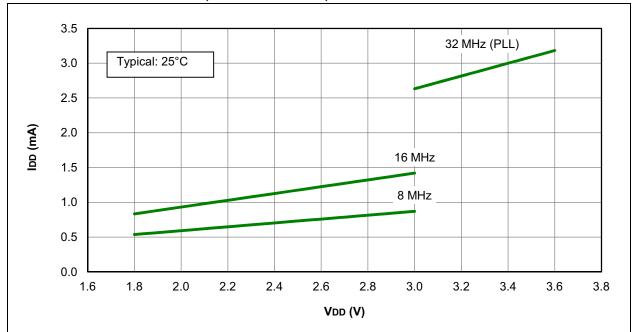


FIGURE 31-16: IDD MAXIMUM, HFINTOSC MODE, PIC16LF1825/1829 ONLY

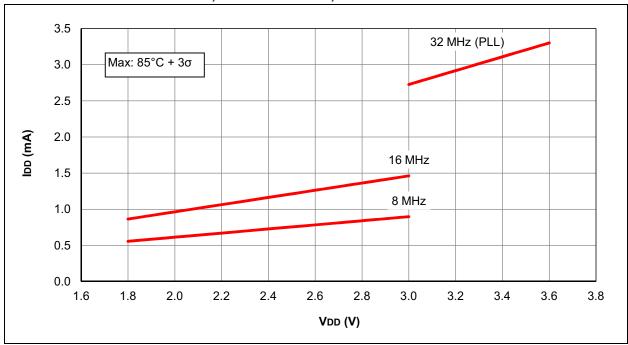


FIGURE 31-17: IDD TYPICAL, HFINTOSC MODE, PIC16F1825/1829 ONLY

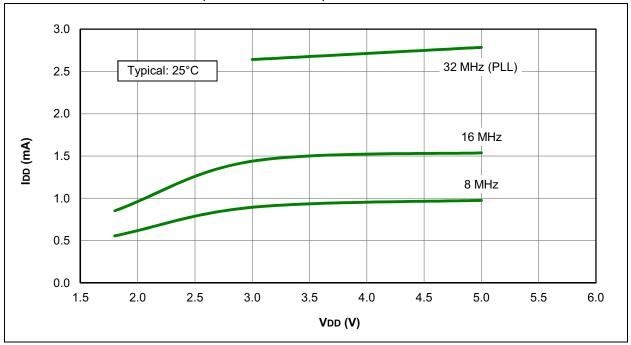
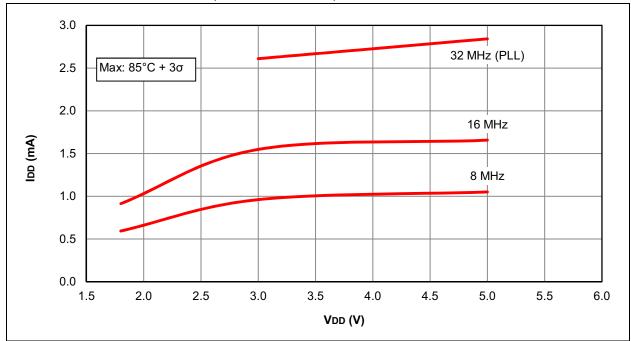
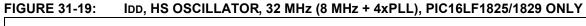


FIGURE 31-18: IDD MAXIMUM, HFINTOSC MODE, PIC16F1825/1829 ONLY





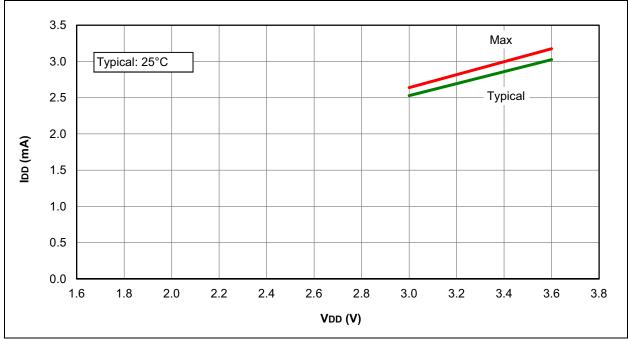


FIGURE 31-20: IDD, HS OSCILLATOR, 32 MHz (8 MHz + 4xPLL), PIC16F1825/1829 ONLY

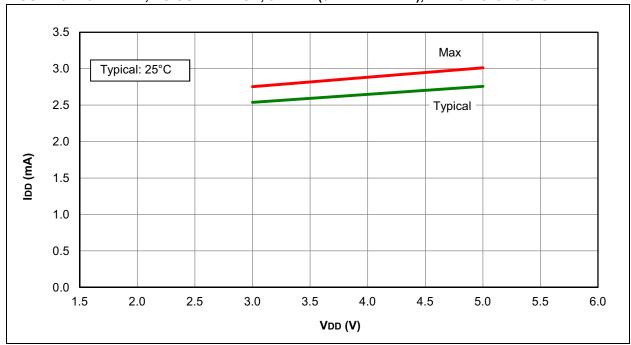


FIGURE 31-21: IPD BASE, LOW-POWER SLEEP MODE, PIC16LF1825/1829 ONLY

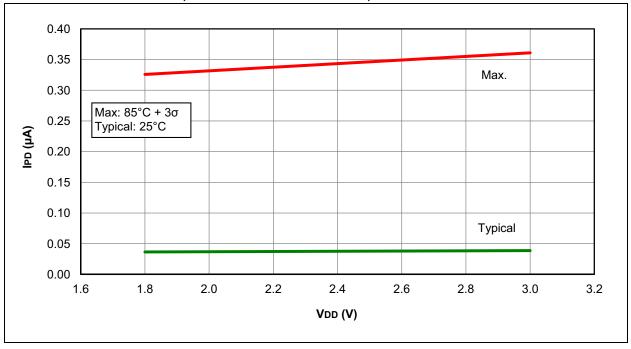
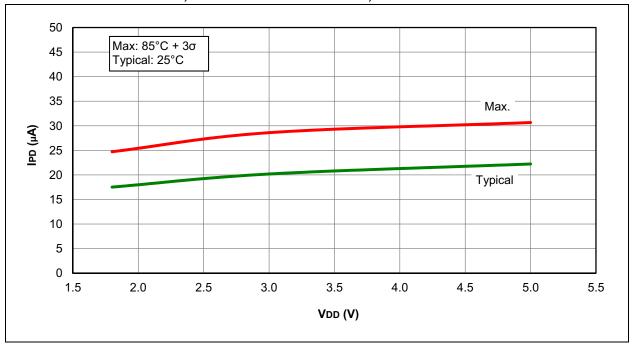
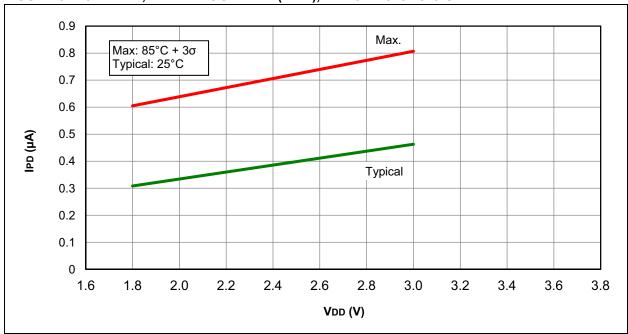


FIGURE 31-22: IPD BASE, LOW-POWER SLEEP MODE, PIC16F1825/1829 ONLY









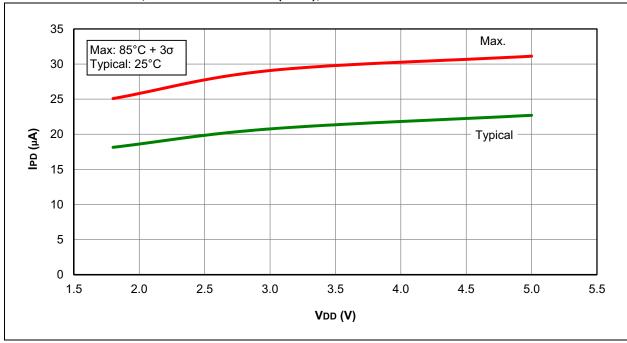


FIGURE 31-25: IPD, FIXED VOLTAGE REFERENCE (FVR), PIC16LF1825/1829 ONLY

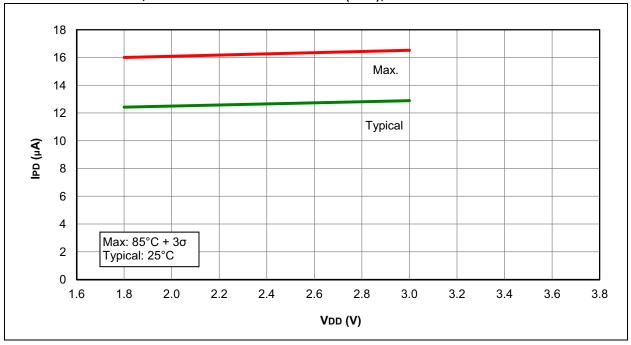
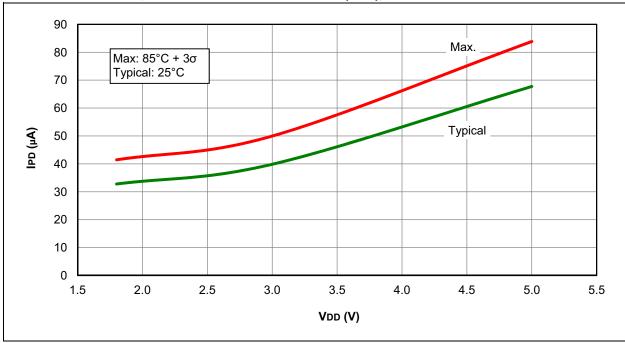


FIGURE 31-26: IPD, FIXED VOLTAGE REFERENCE (FVR), PIC16F1825/1829 ONLY





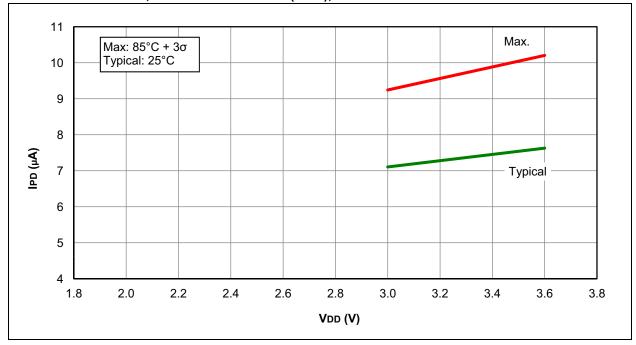


FIGURE 31-28: IPD, BROWN-OUT RESET (BOR), PIC16F1825/1829 ONLY

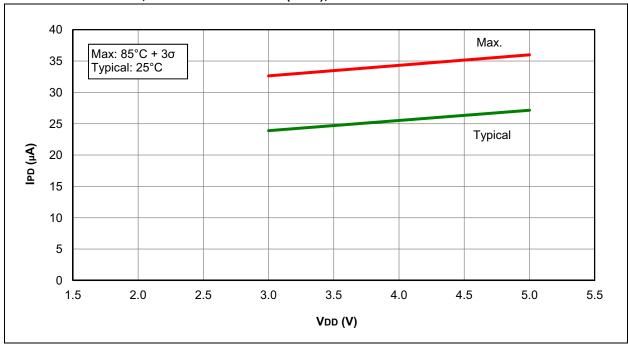


FIGURE 31-29: IPD, TIMER1 OSCILLATOR (Fosc = 32 kHz), PIC16LF1825/1829 ONLY

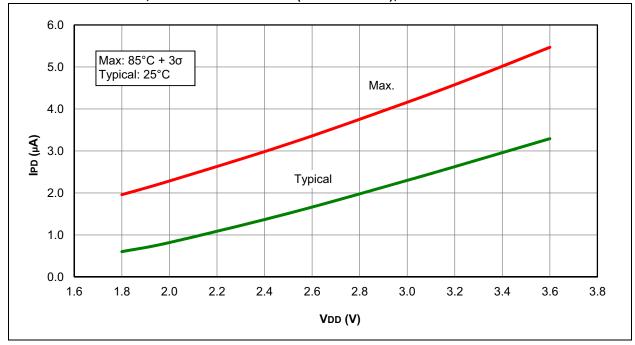


FIGURE 31-30: IPD, TIMER1 OSCILLATOR (Fosc = 32 kHz), PIC16F1825/1829 ONLY

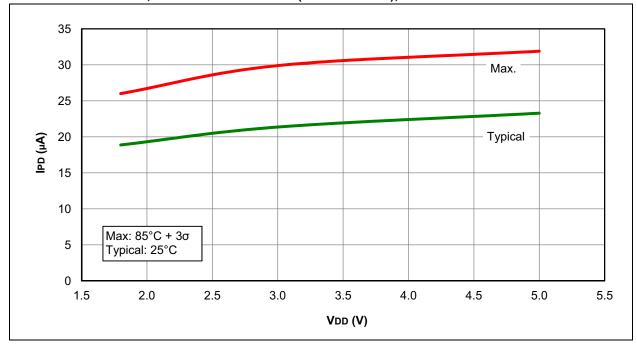


FIGURE 31-31: IPD, CAPACITIVE SENSING (CPS) MODULE, LOW-CURRENT RANGE, CPSRM = 0, PIC16LF1825/1829 ONLY

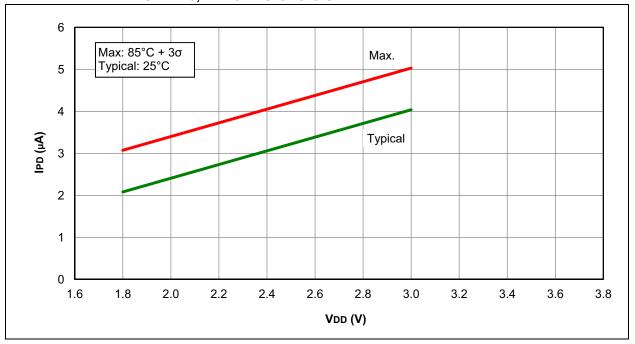


FIGURE 31-32: IPD, CAPACITIVE SENSING (CPS) MODULE, LOW-CURRENT RANGE, CPSRM = 0, PIC16F1825/1829 ONLY

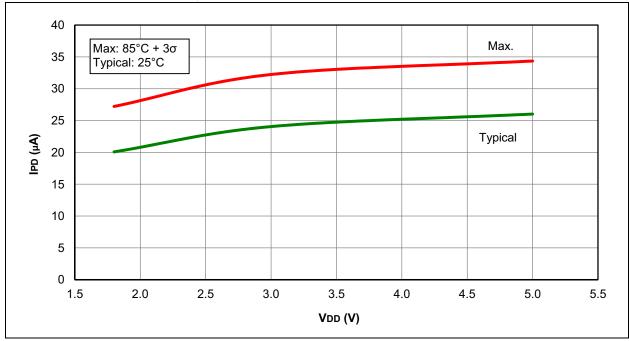


FIGURE 31-33: IPD, CAPACITIVE SENSING (CPS) MODULE, MEDIUM-CURRENT RANGE, CPSRM = 0, PIC16LF1825/1829 ONLY

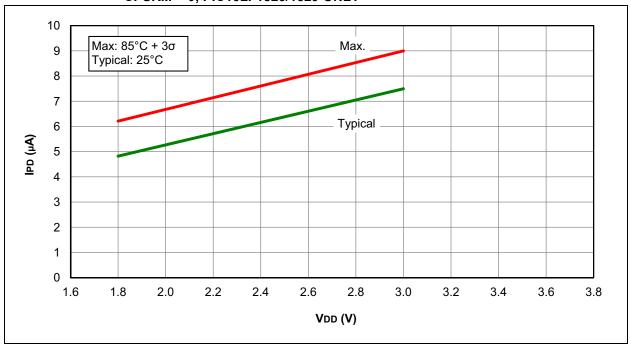


FIGURE 31-34: IPD, CAPACITIVE SENSING (CPS) MODULE, MEDIUM-CURRENT RANGE, CPSRM = 0, PIC16F1825/1829 ONLY

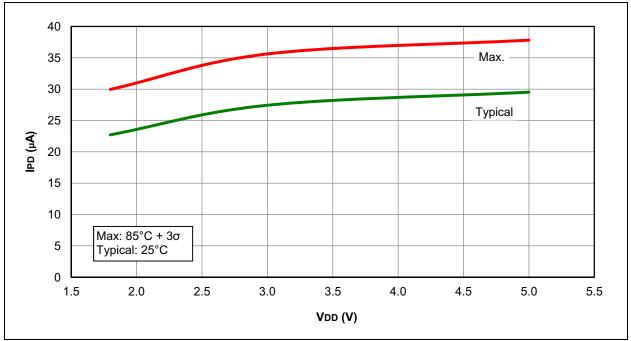


FIGURE 31-35: IPD, CAPACITIVE SENSING (CPS) MODULE, HIGH-CURRENT RANGE, CPSRM = 0, PIC16LF1825/1829 ONLY

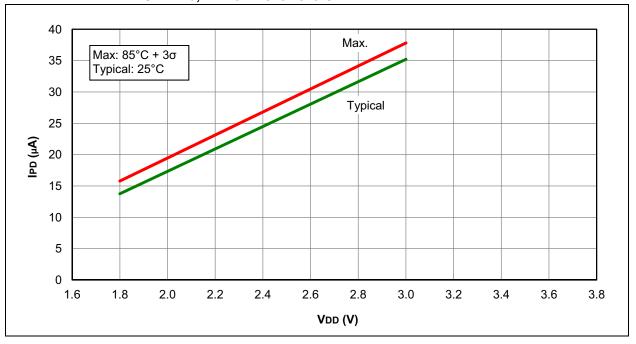
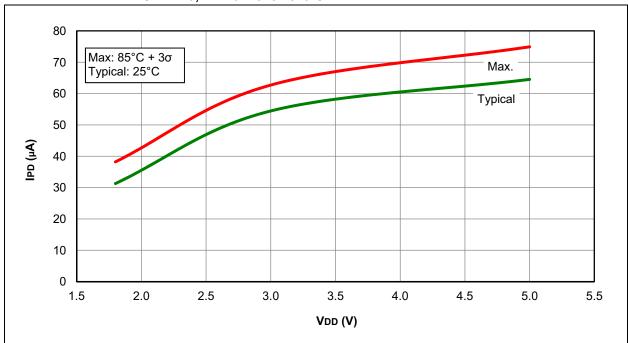


FIGURE 31-36: IPD, CAPACITIVE SENSING (CPS) MODULE, HIGH-CURRENT RANGE, CPSRM = 0, PIC16F1825/1829 ONLY





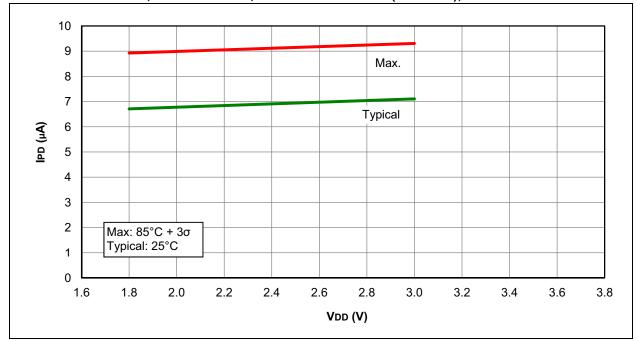


FIGURE 31-38: IPD, COMPARATOR, LOW-POWER MODE (CxSP = 0), PIC16F1825/1829 ONLY

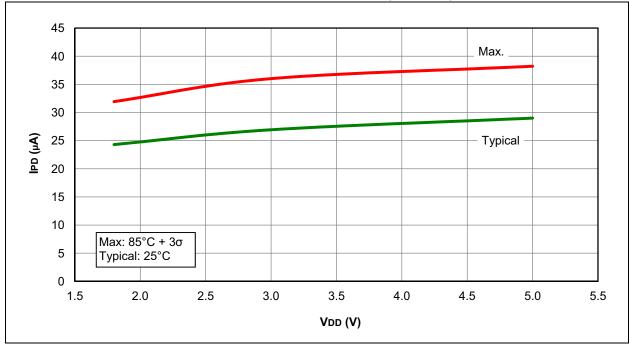


FIGURE 31-39: IPD, COMPARATOR, NORMAL-POWER MODE (CxSP = 1), PIC16LF1825/1829 ONLY

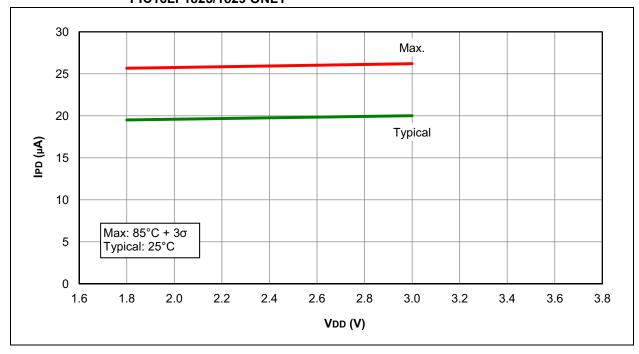


FIGURE 31-40: IPD, COMPARATOR, NORMAL-POWER MODE (CxSP = 1), PIC16F1825/1829 ONLY

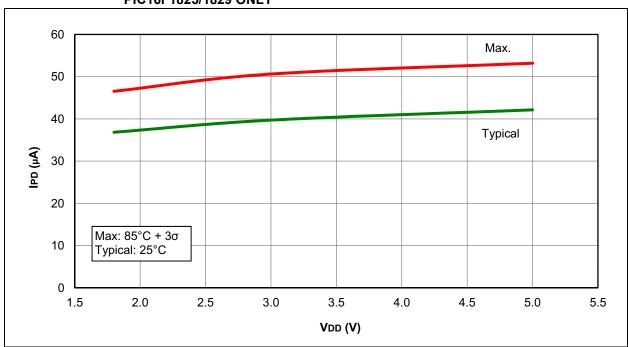


FIGURE 31-41: VOH vs. IOH OVER TEMPERATURE (VDD = 5.0V), PIC16F1825/1829 ONLY

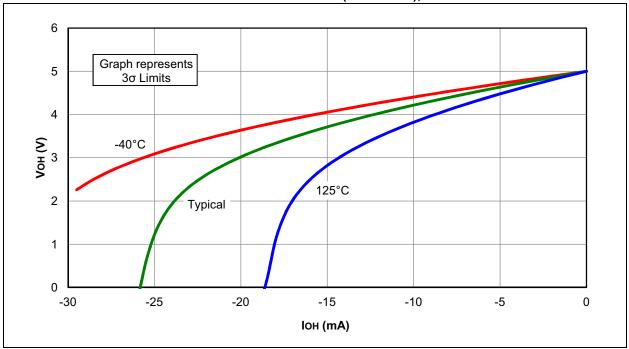
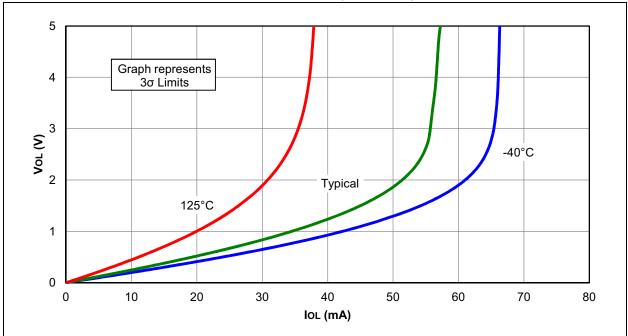


FIGURE 31-42: Vol vs. Iol OVER TEMPERATURE (VDD = 5.0V), PIC16F1825/1829 ONLY





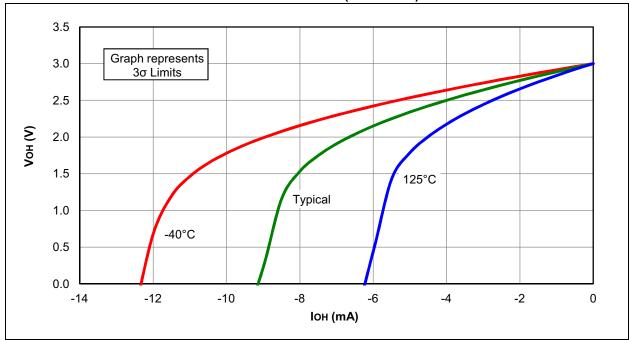


FIGURE 31-44: Vol vs. Iol OVER TEMPERATURE (VDD = 3.0V)

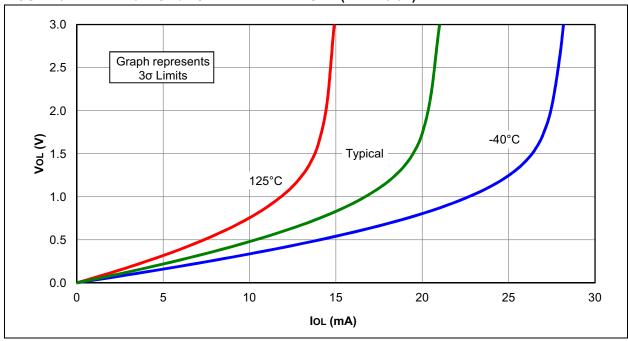


FIGURE 31-45: VOH vs. IOH OVER TEMPERATURE (VDD = 1.8V)

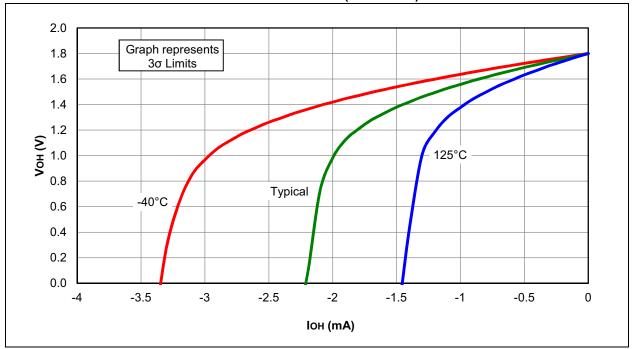


FIGURE 31-46: Vol vs. Iol OVER TEMPERATURE (VDD = 1.8V)

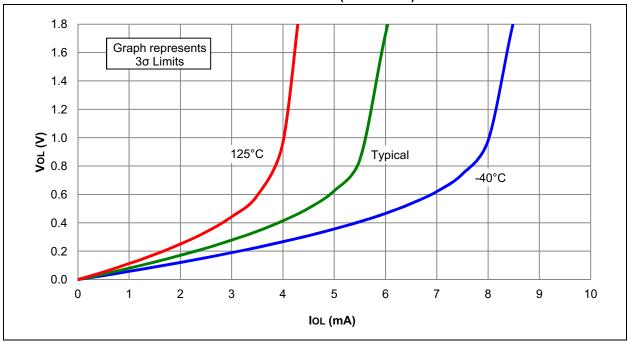


FIGURE 31-47: POR RELEASE VOLTAGE

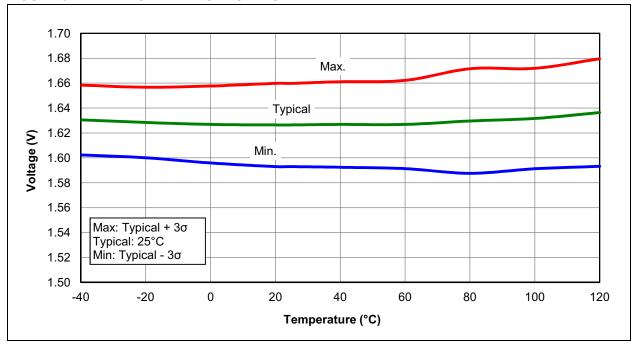


FIGURE 31-48: POR REARM VOLTAGE, PIC16F1825/1829 ONLY

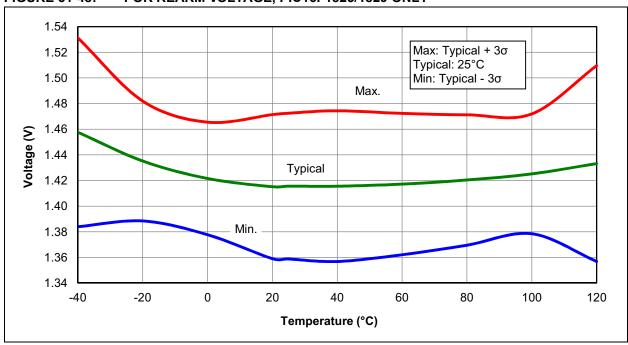


FIGURE 31-49: BROWN-OUT RESET VOLTAGE, BORV = 1

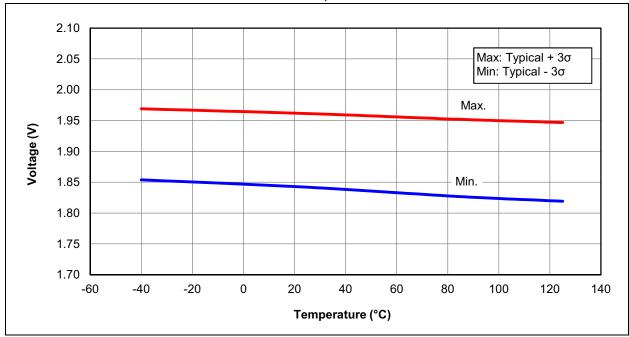


FIGURE 31-50: BROWN-OUT RESET HYSTERESIS, BORV = 1

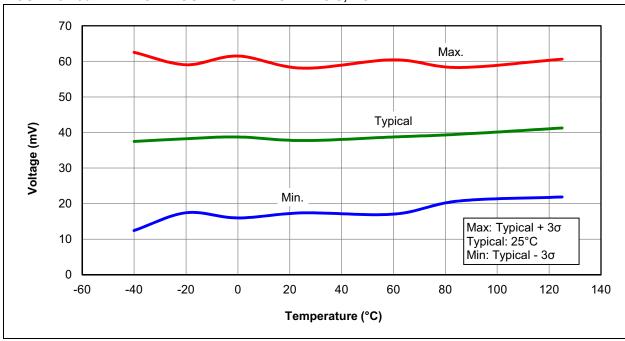


FIGURE 31-51: BROWN-OUT RESET VOLTAGE, BORV = 0

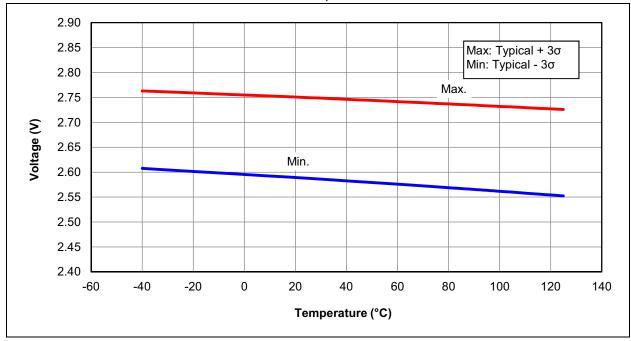


FIGURE 31-52: BROWN-OUT RESET HYSTERESIS, BORV = 0

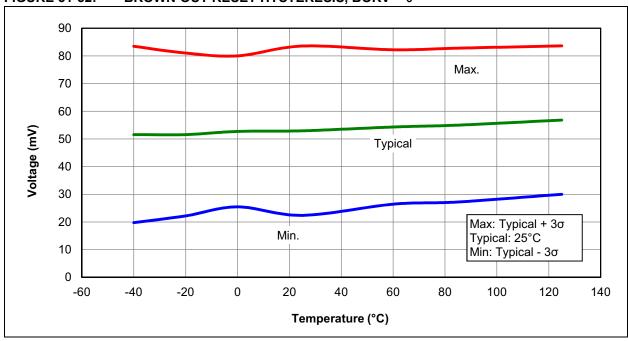


FIGURE 31-53: WDT TIME-OUT PERIOD

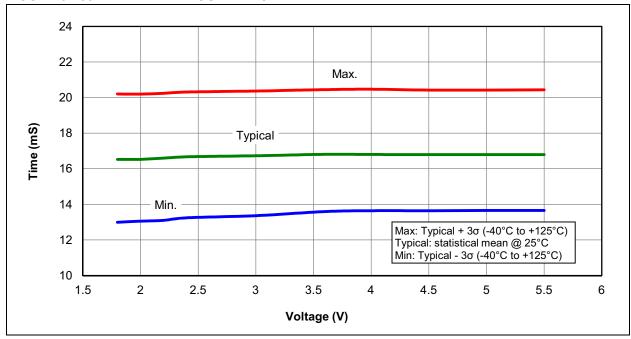
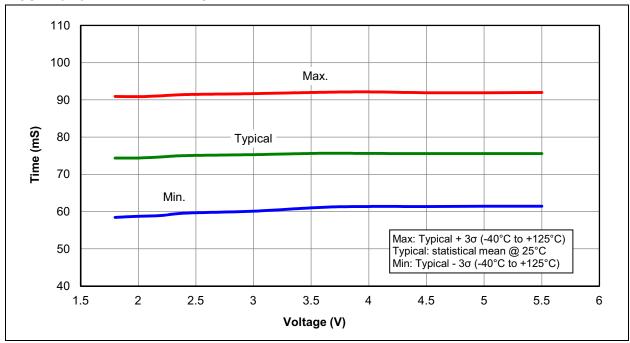
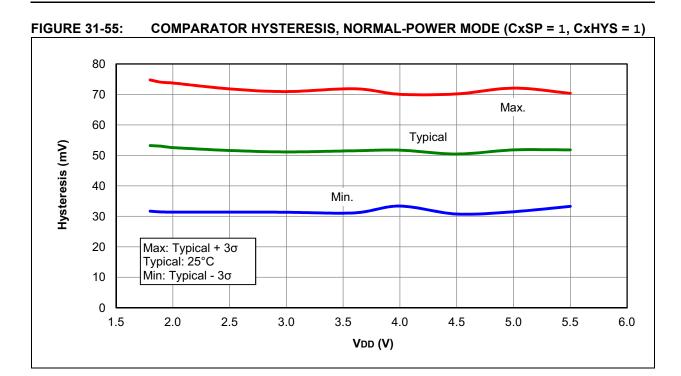
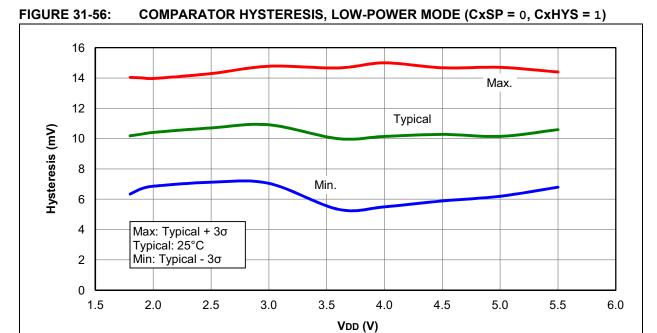


FIGURE 31-54: PWRT PERIOD







© 2010-2012 Microchip Technology Inc.

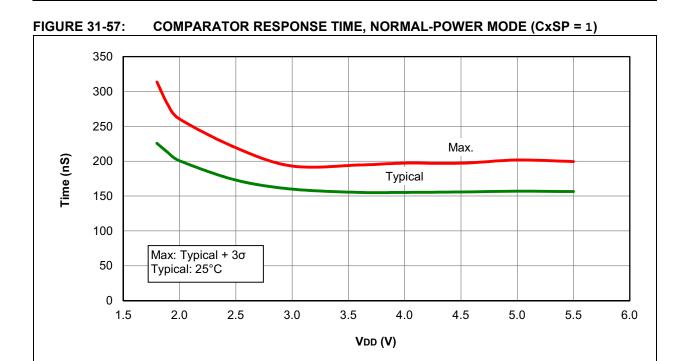


FIGURE 31-58: COMPARATOR RESPONSE TIME OVER TEMPERATURE, NORMAL-POWER MODE (CxSP = 1)

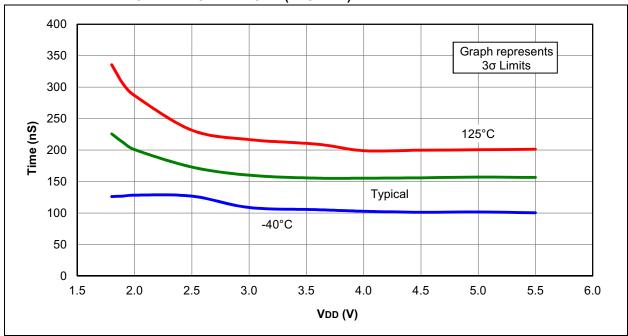
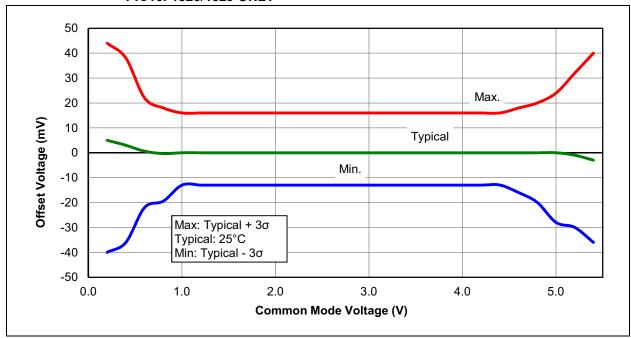


FIGURE 31-59: COMPARATOR INPUT OFFSET AT 25°C, NORMAL-POWER MODE (CxSP = 1), PIC16F1825/1829 ONLY



NOTES:

32.0 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
 - MPLINKTM Object Linker/ MPLIBTM 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

32.1 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.

32.2 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.

32.3 HI-TECH C for Various Device Families

The HI-TECH C Compiler code development systems are complete ANSI C compilers for Microchip's PIC family of microcontrollers and the dsPIC family of digital signal controllers. These compilers provide powerful integration capabilities, omniscient code generation and ease of use.

For easy source level debugging, the compilers provide symbol information that is optimized to the MPLAB IDE debugger.

The compilers include a macro assembler, linker, preprocessor, and one-step driver, and can run on multiple platforms.

32.4 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

32.5 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

32.6 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

32.7 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.

32.8 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 incircuit debugger systems (RJ11) or with the new high-speed, noise tolerant, Low-Voltage Differential Signal (LVDS) interconnection (CAT5).

The emulator is field upgradable through future firmware downloads in MPLAB IDE. In upcoming releases of MPLAB IDE, new devices will be supported, and new features will be added. MPLAB REAL ICE offers significant advantages over competitive emulators including low-cost, full-speed emulation, run-time variable watches, trace analysis, complex breakpoints, a ruggedized probe interface and long (up to three meters) interconnection cables.

32.9 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.

32.10 PICkit 3 In-Circuit Debugger/ Programmer and PICkit 3 Debug Express

The MPLAB PICkit 3 allows debugging and programming of PIC® and dsPIC® Flash microcontrollers at a most affordable price point using the powerful graphical user interface of the MPLAB Integrated Development Environment (IDE). The MPLAB PICkit 3 is connected to the design engineer's PC using a full speed USB interface and can be connected to the target via an Microchip debug (RJ-11) connector (compatible with MPLAB ICD 3 and MPLAB REAL ICE). The connector uses two device I/O pins and the reset line to implement in-circuit debugging and In-Circuit Serial Programming $^{\rm TM}$.

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.

32.11 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.

32.12 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.

32.13 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 PICDEMTM and dsPICDEMTM demonstration/development board series of circuits, Microchip has a line of evaluation kits and demonstration software for analog filter design, Keeloq® security ICs, CAN, IrDA®, PowerSmart battery management, Seevaluation system, Sigma-Delta ADC, flow rate sensing, plus many more.

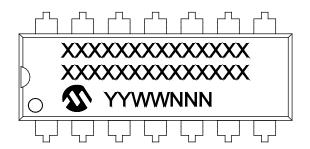
Also available are starter kits that contain everything needed to experience the specified device. This usually includes a single application and debug capability, all on one board.

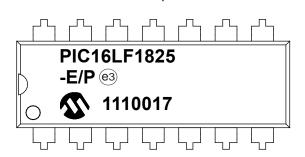
Check the Microchip web page (www.microchip.com) for the complete list of demonstration, development and evaluation kits.

33.0 PACKAGING INFORMATION

33.1 Package Marking Information

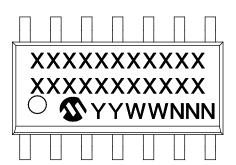
14-Lead PDIP (300 mil)

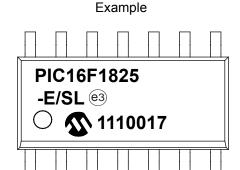




Example

14-Lead SOIC (3.90 mm)





Legend: XX...X Customer-specific information

Y Year code (last digit of calendar year)
YY Year code (last 2 digits of calendar year)
WW Week code (week of January 1 is week '01')

NNN Alphanumeric traceability code

(e3) Pb-free JEDEC designator for Matte Tin (Sn)

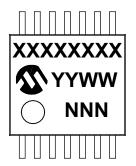
* This package is Pb-free. The Pb-free JEDEC designator (e3)

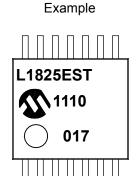
can be found on the outer packaging for this package.

Note: In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line, thus limiting the number of available characters for customer-specific information.

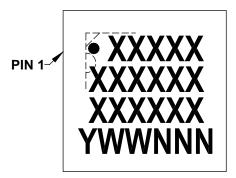
33.1 Package Marking Information

14-Lead TSSOP (4.4 mm)

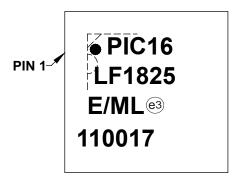




16-Lead QFN (4x4x0.9 mm)



Example



Legend: XX...X Customer-specific information

Y Year code (last digit of calendar year)
YY Year code (last 2 digits of calendar year)
WW Week code (week of January 1 is week '01')

NNN Alphanumeric traceability code

(e3) Pb-free JEDEC designator for Matte Tin (Sn)

This package is Pb-free. The Pb-free JEDEC designator (@3) can be found on the outer packaging for this package.

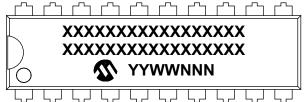
Note: In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line, thus limiting the number of available

characters for customer-specific information.

33.1 Package Marking Information

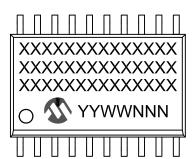
20-Lead PDIP (300 mil)

Example

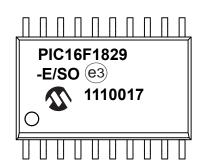




20-Lead SOIC (7.50 mm)



Example



Legend: XX...X Customer-specific information

Y Year code (last digit of calendar year)
YY Year code (last 2 digits of calendar year)
WW Week code (week of January 1 is week '01')

NNN Alphanumeric traceability code

(e3) Pb-free JEDEC designator for Matte Tin (Sn)

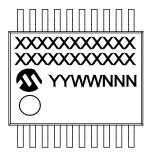
This package is Pb-free. The Pb-free JEDEC designator (e3)

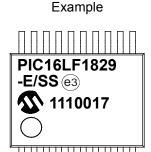
can be found on the outer packaging for this package.

Note: In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line, thus limiting the number of available characters for customer-specific information.

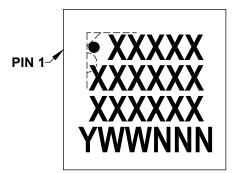
33.1 Package Marking Information

20-Lead SSOP (5.30 mm)

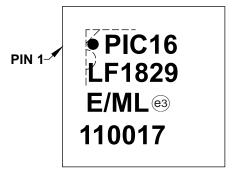




20-Lead QFN (4x4x0.9 mm)



Example



Legend: XX...X Customer-specific information

Y Year code (last digit of calendar year)
YY Year code (last 2 digits of calendar year)
WW Week code (week of January 1 is week '01')

NNN Alphanumeric traceability code

(e3) Pb-free JEDEC designator for Matte Tin (Sn)

This package is Pb-free. The Pb-free JEDEC designator (@3) can be found on the outer packaging for this package.

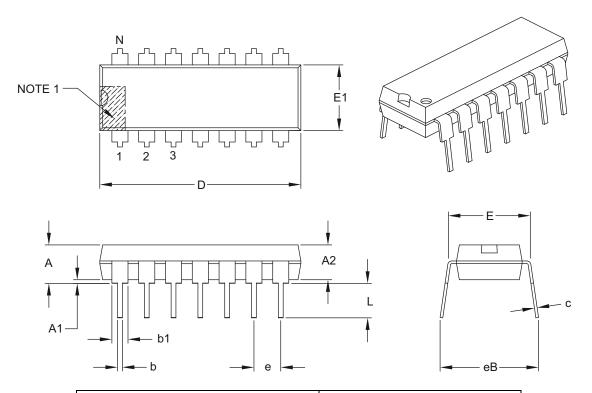
Note: In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line, thus limiting the number of available characters for customer-specific information.

33.2 Package Details

The following sections give the technical details of the packages.

14-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



	Units		INCHES	
Dimen:	sion Limits	MIN	NOM	MAX
Number of Pins	N		14	
Pitch	е		.100 BSC	
Top to Seating Plane	Α	_	_	.210
Molded Package Thickness	A2	.115	.130	.195
Base to Seating Plane	A1	.015	_	_
Shoulder to Shoulder Width	E	.290	.310	.325
Molded Package Width	E1	.240	.250	.280
Overall Length	D	.735	.750	.775
Tip to Seating Plane	L	.115	.130	.150
Lead Thickness	С	.008	.010	.015
Upper Lead Width	b1	.045	.060	.070
Lower Lead Width	b	.014	.018	.022
Overall Row Spacing §	eB	_	_	.430

Notes.

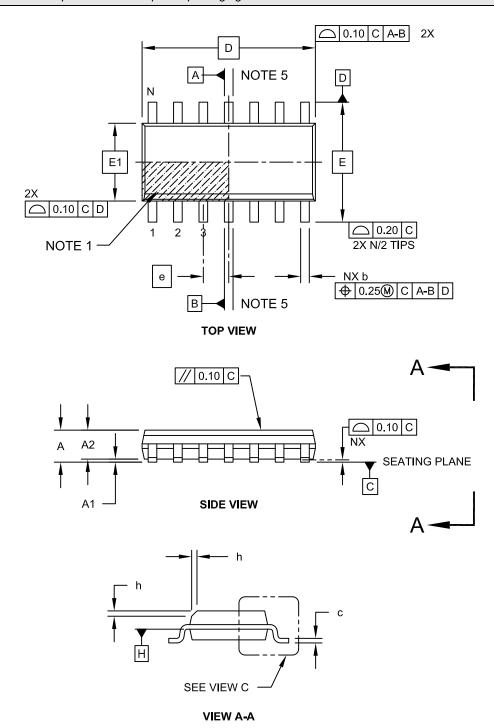
- 1. Pin 1 visual index feature may vary, but must be located with the hatched area.
- 2. § Significant Characteristic.
- 3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" per side.
- 4. Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-005B

14-Lead Plastic Small Outline (SL) - Narrow, 3.90 mm Body [SOIC]

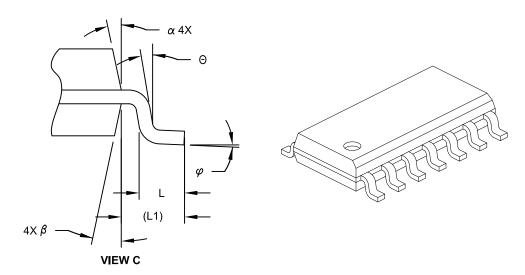
Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Microchip Technology Drawing No. C04-065C Sheet 1 of 2

14-Lead Plastic Small Outline (SL) - Narrow, 3.90 mm Body [SOIC]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



	N	/ILLIMETER	s	
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		14	
Pitch	е		1.27 BSC	
Overall Height	Α	ı	-	1.75
Molded Package Thickness	A2	1.25	-	-
Standoff §	A1	0.10	-	0.25
Overall Width	E		6.00 BSC	
Molded Package Width	E1		3.90 BSC	
Overall Length	D		8.65 BSC	
Chamfer (Optional)	h	0.25	-	0.50
Foot Length	L	0.40	-	1.27
Footprint	L1		1.04 REF	
Lead Angle	Θ	0°	-	-
Foot Angle	φ	0°	-	8°
Lead Thickness	С	0.10	-	0.25
Lead Width	b	0.31	-	0.51
Mold Draft Angle Top	α	5°	-	15°
Mold Draft Angle Bottom	β	5°	-	15°

Notes:

- 1. Pin 1 visual index feature may vary, but must be located within the hatched area.
- 2. § Significant Characteristic
- Dimension D does not include mold flash, protrusions or gate burrs, which shall not exceed 0.15 mm per end. Dimension E1 does not include interlead flash or protrusion, which shall not exceed 0.25 mm per side.
- 4. Dimensioning and tolerancing per ASME Y14.5M

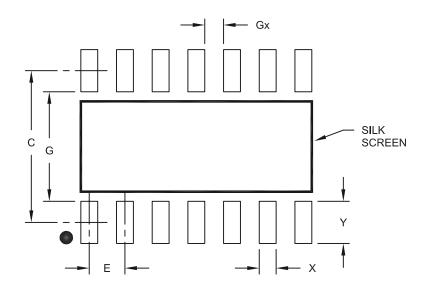
BSC: Basic Dimension. Theoretically exact value shown without tolerances. REF: Reference Dimension, usually without tolerance, for information purposes only.

5. Datums A & B to be determined at Datum H.

Microchip Technology Drawing No. C04-065C Sheet 2 of 2

14-Lead Plastic Small Outline (SL) - Narrow, 3.90 mm Body [SOIC]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



RECOMMENDED LAND PATTERN

	Units		MILLIMETERS		
Dimension	Limits	MIN	NOM	MAX	
Contact Pitch	Е	1.27 BSC			
Contact Pad Spacing	С		5.40		
Contact Pad Width	Х			0.60	
Contact Pad Length	Υ			1.50	
Distance Between Pads	Gx	0.67			
Distance Between Pads	G	3.90			

Notes:

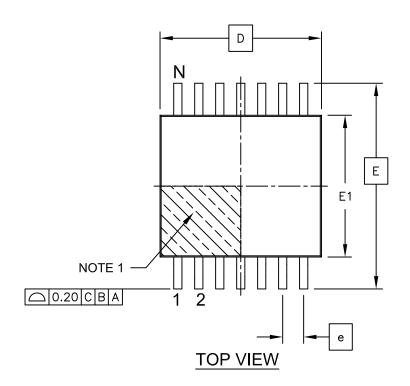
1. Dimensioning and tolerancing per ASME Y14.5M

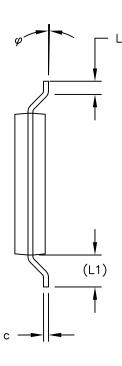
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

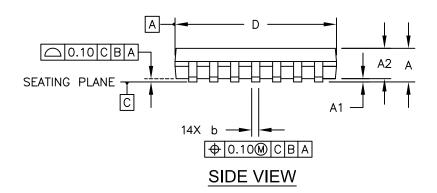
Microchip Technology Drawing No. C04-2065A

14-Lead Plastic Thin Shrink Small Outline (ST) - 4.4 mm Body [TSSOP]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



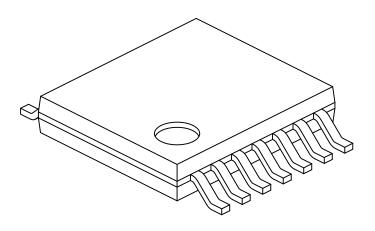




Microchip Technology Drawing C04-087C Sheet 1 of 2

14-Lead Plastic Thin Shrink Small Outline (ST) - 4.4 mm Body [TSSOP]

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	Z		14	
Pitch	е		0.65 BSC	
Overall Height	Α	=	-	1.20
Molded Package Thickness	A2	0.80	1.00	1.05
Standoff	A1	0.05	-	0.15
Overall Width	Е	6.40 BSC		
Molded Package Width	E1	4.30	4.40	4.50
Molded Package Length	D	4.90	5.00	5.10
Foot Length	Г	0.45	0.60	0.75
Footprint	(L1)	1.00 REF		
Foot Angle	φ	0°	-	8°
Lead Thickness	С	0.09	-	0.20
Lead Width	b	0.19	-	0.30

Notes:

- 1. Pin 1 visual index feature may vary, but must be located within the hatched area.
- 2. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.15mm per side.
- 3. Dimensioning and tolerancing per ASME Y14.5M

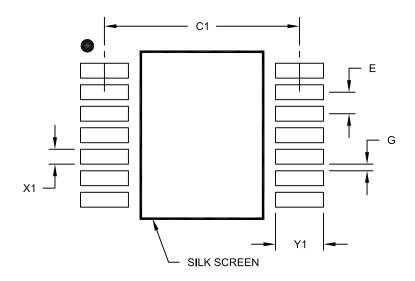
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing No. C04-087C Sheet 2 of 2

14-Lead Plastic Thin Shrink Small Outline (ST) - 4.4 mm Body [TSSOP]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E		0.65 BSC	
Contact Pad Spacing	C1		5.90	
Contact Pad Width (X14)	X1			0.45
Contact Pad Length (X14)	Y1			1.45
Distance Between Pads	G	0.20		

Notes:

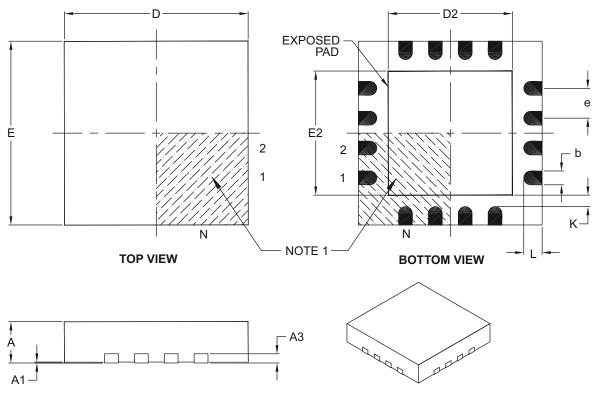
1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2087A

16-Lead Plastic Quad Flat, No Lead Package (ML) – 4x4x0.9 mm Body [QFN]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



	Units	its MILLIMETERS		3
	Dimension Limits	MIN	NOM	MAX
Number of Pins	N		16	
Pitch	е		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		4.00 BSC	
Exposed Pad Width	E2	2.50	2.65	2.80
Overall Length	D		4.00 BSC	
Exposed Pad Length	D2	2.50	2.65	2.80
Contact Width	b	0.25	0.30	0.35
Contact Length	L	0.30	0.40	0.50
Contact-to-Exposed Pad	K	0.20	_	_

Notes:

- 1. Pin 1 visual index feature may vary, but must be located within the hatched area.
- 2. Package is saw singulated.
- 3. Dimensioning and tolerancing per ASME Y14.5M.

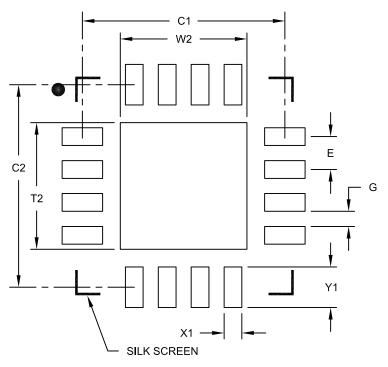
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-127B

16-Lead Plastic Quad Flat, No Lead Package (ML) - 4x4x0.9mm Body [QFN]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension	Limits	MIN	NOM	MAX
Contact Pitch	Е		0.65 BSC	
Optional Center Pad Width	W2			2.50
Optional Center Pad Length	T2			2.50
Contact Pad Spacing	C1		4.00	
Contact Pad Spacing	C2		4.00	
Contact Pad Width (X16)	X1			0.35
Contact Pad Length (X16)	Y1			0.80
Distance Between Pads	G	0.30		

Notes:

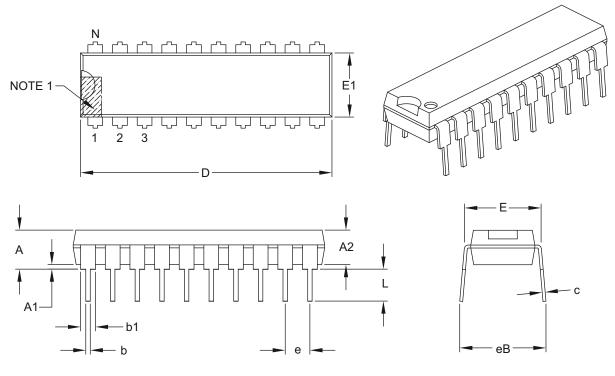
1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2127A

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



	Units		INCHES	
Dimension	n Limits	MIN	NOM	MAX
Number of Pins	N		20	
Pitch	е		.100 BSC	
Top to Seating Plane	Α	_	_	.210
Molded Package Thickness	A2	.115	.130	.195
Base to Seating Plane	A1	.015	_	_
Shoulder to Shoulder Width	Е	.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	С	.008	.010	.015
Upper Lead Width	b1	.045	.060	.070
Lower Lead Width	b	.014	.018	.022
Overall Row Spacing §	eB	_	_	.430

Notes:

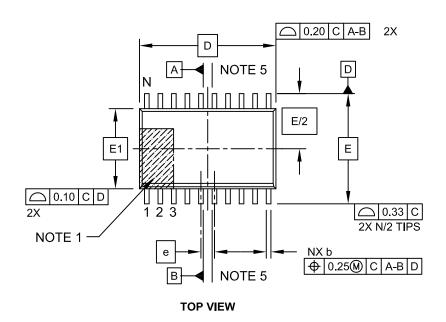
- 1. Pin 1 visual index feature may vary, but must be located within the hatched area.
- 2. § Significant Characteristic.
- 3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" per side.
- 4. Dimensioning and tolerancing per ASME Y14.5M.

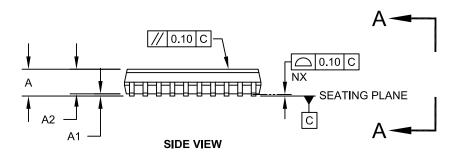
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

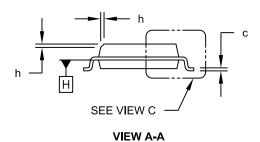
Microchip Technology Drawing C04-019B

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



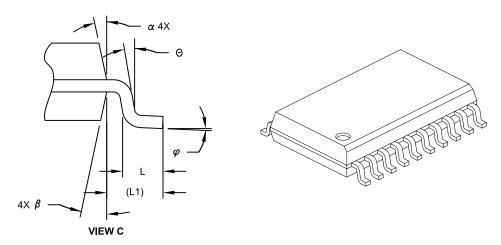




Microchip Technology Drawing C04-094C Sheet 1 of 2

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



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		20	
Pitch	е		1.27 BSC	
Overall Height	Α	-	-	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	Θ	0°	-	-
Foot Angle	φ	0°	_	8°
Lead Thickness	С	0.20	-	0.33
Lead Width	b	0.31	_	0.51
Mold Draft Angle Top	α	5°	-	15°
Mold Draft Angle Bottom	β	5°	-	15°

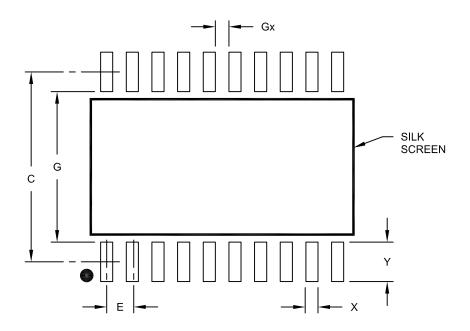
Notes:

- 1. Pin 1 visual index feature may vary, but must be located within the hatched area.
- 2. § Significant Characteristic
- 3. Dimension D does not include mold flash, protrusions or gate burrs, which shall not exceed 0.15 mm per end. Dimension E1 does not include interlead flash or protrusion, which shall not exceed 0.25 mm per side.
- 4. Dimensioning and tolerancing per ASME Y14.5M
 - BSC: Basic Dimension. Theoretically exact value shown without tolerances.
 - REF: Reference Dimension, usually without tolerance, for information purposes only.
- 5. Datums A & B to be determined at Datum H.

Microchip Technology Drawing No. C04-094C Sheet 2 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



RECOMMENDED LAND PATTERN

	MILLIMETERS			
Dimension Limits		MIN	NOM	MAX
Contact Pitch	Е	1.27 BSC		
Contact Pad Spacing	С		9.40	
Contact Pad Width (X20)	Х			0.60
Contact Pad Length (X20)	Υ			1.95
Distance Between Pads	Gx	0.67		
Distance Between Pads	G	7.45		

Notes:

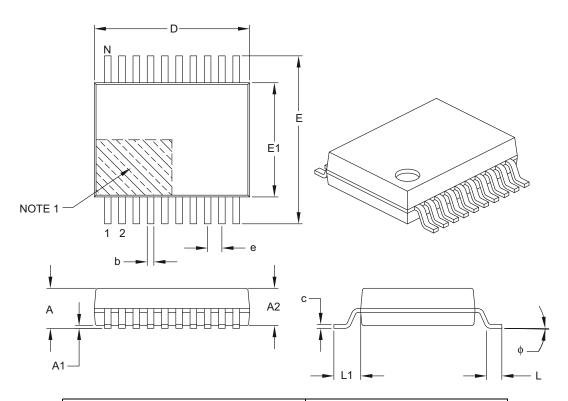
1. Dimensioning and tolerancing per ASME Y14.5M

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



	Units		MILLIMETERS		
Din	nension Limits	MIN	NOM	MAX	
Number of Pins	N		20		
Pitch	е		0.65 BSC		
Overall Height	Α	_	_	2.00	
Molded Package Thickness	A2	1.65	1.75	1.85	
Standoff	A1	0.05	_	_	
Overall Width	E	7.40	7.80	8.20	
Molded Package Width	E1	5.00	5.30	5.60	
Overall Length	D	6.90	7.20	7.50	
Foot Length	L	0.55	0.75	0.95	
Footprint	L1		1.25 REF		
Lead Thickness	С	0.09	_	0.25	
Foot Angle	ф	0°	4°	8°	
Lead Width	b	0.22	_	0.38	

Notes:

- 1. Pin 1 visual index feature may vary, but must be located within the hatched area.
- 2. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.20 mm per side.
- 3. Dimensioning and tolerancing per ASME Y14.5M.

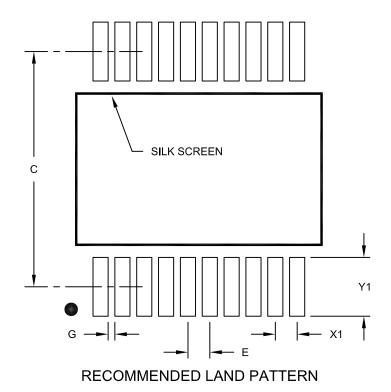
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-072B

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



	Units			S
Dimension Limits		MIN	NOM	MAX
Contact Pitch	Е	0.65 BSC		
Contact Pad Spacing	С		7.20	
Contact Pad Width (X20)	X1			0.45
Contact Pad Length (X20)	Y1			1.75
Distance Between Pads	G	0.20		

Notes:

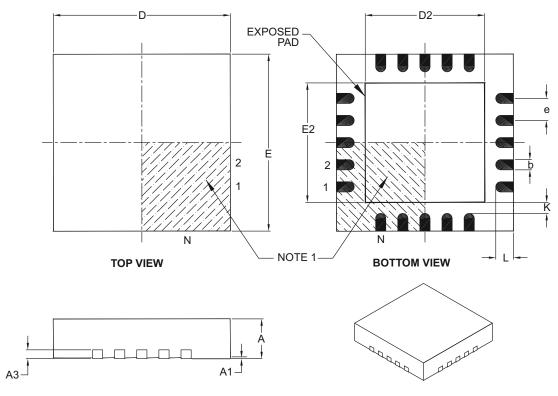
1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2072A

20-Lead Plastic Quad Flat, No Lead Package (ML) – 4x4x0.9 mm Body [QFN]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



	Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX	
Number of Pins	N	20			
Pitch	е	0.50 BSC			
Overall Height	Α	0.80	0.90	1.00	
Standoff	A1	0.00	0.02	0.05	
Contact Thickness	А3	0.20 REF			
Overall Width	Е	4.00 BSC			
Exposed Pad Width	E2	2.60	2.70	2.80	
Overall Length	D	4.00 BSC			
Exposed Pad Length	D2	2.60	2.70	2.80	
Contact Width	b	0.18	0.25	0.30	
Contact Length	L	0.30	0.40	0.50	
Contact-to-Exposed Pad	K	0.20	_	_	

Notes:

- 1. Pin 1 visual index feature may vary, but must be located within the hatched area.
- 2. Package is saw singulated.
- 3. Dimensioning and tolerancing per ASME Y14.5M.

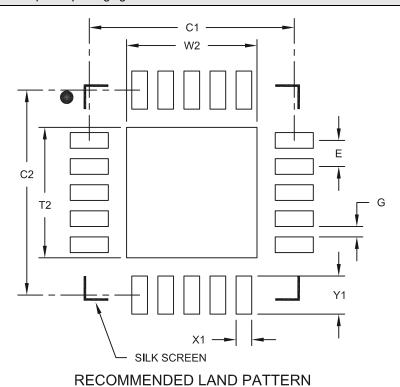
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-126B

20-Lead Plastic Quad Flat, No Lead Package (ML) - 4x4 mm Body [QFN] With 0.40 mm Contact Length

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Units **MILLIMETERS Dimension Limits** MIN MOM MAX Contact Pitch Ε 0.50 BSC Optional Center Pad Width W2 2.50 Optional Center Pad Length T2 2.50 Contact Pad Spacing C1 3.93 Contact Pad Spacing C2 3.93 Contact Pad Width Χ1 0.30 Contact Pad Length <u>Y1</u> 0.73 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. C04-2126A

NOTES:

APPENDIX A: DATA SHEET REVISION HISTORY

Revision A (08/2010)

Original release.

Revision B (05/2011)

Revised Electrical Specifications.

Revision C (06/2012)

Updated the Family Types table; Updated Figures 1, 2 and 3; Updated Table 3-3; Changed all instances of SDO into SDO1, SDOSEL into SDO1SEL and SSSEL into SS1SEL; Added PIR3, PIR4, PIE3 and PIE4 to Table 3-3; Updated Register 4-2; Updated Sections 5.2.2.5 and 5.5.3; Added Note 1 to Table 11-3; Updated Figure 13-1 and Equation 16-1; Updated Section 19.9; Added charts to the DC and AC Characteristics Graphs section; Revised the Electrical Specifications section; Updated the Product Identification System section; Other minor corrections.

APPENDIX B: MIGRATING FROM OTHER PIC® DEVICES

This shows a comparison of features in the migration from the PIC16F648 device to the PIC16(L)F1825/1829 family of devices.

This section provides comparisons when migrating from other similar PIC[®] devices to the PIC16(L)F1825/1829 family of devices.

B.1 PIC16F648A to PIC16F1825/1829 TABLE B-1: FEATURE COMPARISON

Feature	PIC16F648A	PIC16F1825/1829
Max. Operating Speed	20 MHz	32 MHz
Max. Program Memory (Words)	4K	8K
Max. SRAM (Bytes)	256	1024
Max. EEPROM (Bytes)	256	256
A/D Resolution	10-bit	10-bit
Timers (8/16-bit)	2/1	4/1
Brown-out Reset	Y	Y
Internal Pull-ups	RB<7:0>	PIC16F1825: RA<5:0>, RC<5:0> DSTEMP: RA<5:0>, RB<7:4>, RC<7:0>
Interrupt-on-change	RB<7:4>	PIC16F1825: RA<5:0>, Edge Selectable DSTEMP: RA<5:0>, RB<7:4>, Edge Selectable
Comparator	2	2
AUSART/EUSART	1/0	0/1
Extended WDT	N	Y
Software Control Option of WDT/BOR	N	Y
INTOSC Frequencies	48 kHz or 4 MHz	31 kHz - 32 MHz
Clock Switching	Υ	Y
Capacitive Sensing	N	Y
CCP/ECCP	2/0	2/2
Enhanced PIC16 CPU	N	Y
MSSPx/SSPx	0	2/0
Reference Clock	N	Y
Data Signal Modulator	N	Y
SR Latch	N	Y
Voltage Reference	N	Y
DAC	Υ	Y

NOTES:

INDEX

Α		EUSART Receive	
A/D		EUSART Transmit	293
Specifications	368	External RC Mode	60
Absolute Maximum Ratings		Fail-Safe Clock Monitor (FSCM)	68
AC Characteristics		Generic I/O Port	
Industrial and Extended	362	Interrupt Logic	85
Load Conditions		On-Chip Reset Circuit	77
ACKSTAT		PIC16F/LF1825/1829	20
ACKSTAT Status Flag		PIC16F/LF1825/29	
ADC		PWM (Enhanced)	222
Acquisition Requirements		Resonator Operation	58
Associated registers		Timer0	183
Block Diagram		Timer1	187
Calculating Acquisition Time		Timer1 Gate	193, 194
Channel Selection		Timer2/4/6	
Configuration		Voltage Reference	
Configuring Interrupt		Voltage Reference Output Buffer Example	
Conversion Clock		BORCON Register	
Conversion Procedure		BRA	338
Internal Sampling Switch (Rss) Impedance		Break Character (12-bit) Transmit and Receive	
Interrupts		Brown-out Reset (BOR)	
Operation		Specifications	
Operation During Sleep		Timing and Characteristics	365
Port Configuration		С	
Reference Voltage (VREF)		_	
Source Impedance		C Compilers	
Special Event Trigger		MPLAB C18	
Starting an A/D Conversion		CALL	
ADCON0 Register		CALLW	
ADCON1 Register		Capacitive Sensing	
ADDFSR	·	Associated registers w/ Capacitive Sensing	
ADDWFC		Specifications	
ADRESH Register		Capture Module. See Enhanced Capture/Compare/F	'WM
ADRESH Register (ADFM = 0)		(ECCP)	
ADRESH Register (ADFM = 1)		Capture/Compare/PWM	213
ADRESL Register (ADFM = 0)		Capture/Compare/PWM (CCP)	
ADRESL Register (ADFM = 1)		Associated Registers w/ Capture	
Alternate Pin Function		Associated Registers w/ Compare	
Analog-to-Digital Converter. See ADC		Associated Registers w/ PWM	
ANSELA Register	126	Capture Mode	
ANSELB Register		CCPx Pin Configuration	
ANSELC Register		Compare Mode	
APFCON0 Register		CCPx Pin Configuration	
APFCON1 Register		Software Interrupt Mode	
Assembler		Special Event Trigger	
MPASM Assembler	412	Timer1 Mode Resource	
_		Prescaler	214
В		PWM Mode	
BAUDCON Register	304	Duty Cycle	
BF		Effects of Reset	221
BF Status Flag	·	Example PWM Frequencies and	
Block Diagram	•	Resolutions, 20 MHZ	220
Capacitive Sensing	321, 322	Example PWM Frequencies and	
Block Diagrams	•	Resolutions, 32 MHZ	220
(CCP) Capture Mode Operation	214	Example PWM Frequencies and	
ADC		Resolutions, 8 MHz	
ADC Transfer Function		Operation in Sleep Mode	
Analog Input Model		Resolution	
CCP PWM		System Clock Frequency Changes	
Clock Source		PWM Operation	
Compare		PWM Overview	
Crystal Operation		PWM Period	219
Digital-to-Analog Converter (DAC)		PWM Setup	
2.g 10 . 1.1.10g Controller (D/10/		CCP1CON Register	36, 37

CCPR1H Register	36, 37	Configuration Word	49
CCPR1L Register	36, 37	User ID	53, 54
CCPTMRS Register	237	Device Overview	11, 101
CCPxAS Register	238	Digital-to-Analog Converter (DAC)	163
CCPxCON (ECCPx) Register	236	Associated Registers	167
CLKRCON Register	74	Specifications	370
Clock Accuracy with Asynchronous Operation	302	· -	
Clock Sources		E	
External Modes	57	ECCP/CCP. See Enhanced Capture/Compare/PWM	Л
EC	57	EEADR Registers	105
HS	57	EEADRH Registers	105
LP		EEADRL Register	
OST		EEADRL Registers	
RC		EECON1 Register	
XT		EECON2 Register	
Internal Modes		EEDATH Register	
HFINTOSC		EEDATL Register	
Internal Oscillator Clock Switch Timing		EEPROM Data Memory	
LFINTOSC		Avoiding Spurious Write	106
MFINTOSC		Write Verify	
		Effects of Reset	110
Clock Switching		PWM mode	221
CMOUT Register		Electrical Specifications	
CMxCON0 Register			
CMxCON1 Register	180	Enhanced Capture/Compare/PWM (ECCP)	
Code Examples		Enhanced PWM Mode	
A/D Conversion		Auto-Restart	
Changing Between Capture Prescalers		Auto-shutdown	
Initializing PORTA		Direction Change in Full-Bridge Output M	
Write Verify		Full-Bridge Application	
Writing to Flash Program Memory	113	Full-Bridge Mode	
Comparator		Half-Bridge Application	
Associated Registers	181	Half-Bridge Application Examples	
Operation	175	Half-Bridge Mode	225
Comparator Module	175	Output Relationships (Active-High and	
Cx Output State Versus Input Conditions	176	Active-Low)	
Comparator Specifications	370	Output Relationships Diagram	
Comparators		Programmable Dead Band Delay	
C2OUT as T1 Gate	189	Shoot-through Current	232
Compare Module. See Enhanced Capture/Com	pare/PWM	Start-up Considerations	234
(ECCP)		Specifications	367
CONFIG1 Register	50	Enhanced Mid-range CPU	19
CONFIG2 Register		Enhanced Universal Synchronous Asynchronous	
CPSCON0 Register	327	Receiver Transmitter (EUSART)	293
CPSCON1 Register		Errata	10
Customer Change Notification Service		EUSART	293
Customer Notification Service		Associated Registers	
Customer Support		Baud Rate Generator	306
		Asynchronous Mode	
D		12-bit Break Transmit and Receive	
DACCON0 (Digital-to-Analog Converter Control 0)		Associated Registers	
Register	166	Receive	301
DACCON1 (Digital-to-Analog Converter Control 1)		Transmit	
Register		Auto-Wake-up on Break	
Data EEPROM Memory		Baud Rate Generator (BRG)	
Associated Registers		Clock Accuracy	
Code Protection		Receiver	
Reading		Setting up 9-bit Mode with Address Detec	
Writing		Transmitter(PDO)	290
Data Memory		Baud Rate Generator (BRG)	040
DC and AC Characteristics	379	Auto Baud Rate Detect	
DC Characteristics		Baud Rate Error, Calculating	
Extended and Industrial		Baud Rates, Asynchronous Modes	
Industrial and Extended		Formulas	
Development Support		High Baud Rate Select (BRGH Bit)	
Device Configuration		Synchronous Master Mode	. 314, 318
Code Protection	53	Associated Registers	

Receive317	MOVF	341
Transmit315	MOVIW	342
Reception316	MOVLB	342
Transmission	MOVWI	343
Synchronous Slave Mode	OPTION	343
Associated Registers	RESET	343
Receive319	SUBWFB	345
Transmit318	TRIS	346
Reception319	BCF	338
Transmission	BSF	338
Extended Instruction Set	BTFSC	338
ADDFSR337	BTFSS	338
	CALL	339
F	CLRF	339
Fail-Safe Clock Monitor	CLRW	339
Fail-Safe Condition Clearing68	CLRWDT	339
Fail-Safe Detection68	COMF	339
Fail-Safe Operation68	DECF	
Reset or Wake-up from Sleep68	DECFSZ	
Firmware Instructions	GOTO	
Fixed Voltage Reference (FVR)145	INCF	
Associated Registers	INCFSZ	
Flash Program Memory	IORLW	
Erasing110	IORWF	
Modifying114	MOVLW	
Writing110	MOVWF	
FSR Register 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41	NOP	
FVRCON (Fixed Voltage Reference Control) Register 146	RETFIE	
1 VICON (1 Ixed Voltage Neleterice Control) Negister 140	RETLW	
1	RETURN	
I ² C Mode (MSSPx)	RLF	
Acknowledge Sequence Timing	RRF	
Bus Collision		
During a Repeated Start Condition284	SLEEP	
During a Stop Condition	SUBLW	
Effects of a Reset281	SUBWF	
I ² C Clock Rate w/BRG287	SWAPF	
Master Mode	XORLW	
Operation272	XORWF	
Reception	INTCON Register	90
Start Condition Timing274, 275	Internal Oscillator Block	
Transmission	INTOSC	
Multi-Master Communication, Bus Collision and	Specifications	
Arbitration	Internal Sampling Switch (Rss) Impedance	
Multi-Master Mode	Internet Address	
	Interrupt-On-Change	
Read/Write Bit Information (R/W Bit)257	Associated Registers	
Slave Mode	Interrupts	
Transmission	ADC	
Sleep Operation	Associated registers w/ Interrupts	
Stop Condition Timing	TMR1	
INDF Register 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41	INTOSC Specifications	
Indirect Addressing45	IOCAF Register	14′
INLVLA Register	IOCAN Register	141
INLVLB Register	IOCAP Register	140
Instruction Format	IOCBF Register	142
Instruction Set	IOCBN Register	142
ADDLW 337	IOCBP Register	141
ADDWF337	1	
ADDWFC	L	
ANDLW337	LATA Register	. 126, 136
ANDWF337	LATB Register	131
BRA338	Load Conditions	361
CALL339	LSLF	341
CALLW339	LSRF	341
LSLF341		
LSRF341		

M		PIE2 Register	32, 92
Master Synchronous Serial Port. See MSSPx		PIE3 Register	32, 93
MCLR	80	PIE4 Register	32, 94
Internal		Pinout Descriptions	
MDCARH Register		PIC16F/LF1826/27	
MDCARL Register		PIR1 Register	31, 95
MDCON Register		PIR2 Register	31, 96
MDSRC Register		PIR3 Register	31, 97
Memory Organization		PIR4 Register	31, 98
Data		PORTA	123
		ANSELA Register	123
Program Microchip Internet Web Site		Associated Registers	128
		Configuration Word w/ PORTA	128
Migrating from other PIC Micro Devices		LATA Register	
MOVIW		PORTA Register	
MOVLB		Specifications	
MOVWI		PORTA Register	
MPLAB ASM30 Assembler, Linker, Librarian		PORTB	
MPLAB Integrated Development Environment Software Company of the C		ANSELB Register	129
MPLAB PM3 Device Programmer		Associated Registers	
MPLAB REAL ICE In-Circuit Emulator System		LATB Register	
MPLINK Object Linker/MPLIB Object Librarian		Pin Descriptions and Diagrams	
MSSPx		PORTB Register	
I ² C Mode	252		
I ² C Mode Operation	254	PORTB Register	13
SPI Mode	244	PORTC	407
SSPxBUF Register	247	ANSELC Register	
SSPxSR Register	247	Associated Registers	
•		LATC Register	
0		Pin Descriptions and Diagrams	
OPCODE Field Descriptions	333	PORTC Register	
OPTION	343	PORTC Register	
OPTION_REG Register	185	Power-Down Mode (Sleep)	
OSCCON Register		Associated Registers100	ე, 211
Oscillator		Power-on Reset	
Associated Registers	72	Power-up Time-out Sequence	80
Oscillator Module		Power-up Timer (PWRT)	78
ECH		Specifications	366
ECL		PR2 Register	3′
ECM		PR4 Register	39
HS		PR6 Register	39
INTOSC		Precision Internal Oscillator Parameters	
LP		Program Memory	
RC		Map and Stack (PIC16F/LF1826)	
XT		Map and Stack (PIC16F/LF1826/27)	
		Programming, Device Instructions	
Oscillator Parameters		PSTRxCON Register	
Oscillator Specifications	362	PWM (ECCP Module)	
Oscillator Start-up Timer (OST)		PWM Steering	233
Specifications	366	Steering Synchronization	
Oscillator Switching		PWM Mode. See Enhanced Capture/Compare/PWM	
Fail-Safe Clock Monitor		PWM Steering	
Two-Speed Clock Start-up		PWMxCON Register	
OSCSTAT Register		F WIMACON Register	238
OSCTUNE Register	72	R	
P		RCREG	200
P1A/P1B/P1C/P1D.See Enhanced Capture/Compa	ire/	RCREG Register	
PWM (ECCP)	222	RCSTA Register	
Packaging		Reader Response	
Marking415, 416	, 417, 418	Read-Modify-Write Operations	
PDIP Details	419	Reference Clock	
PCL and PCLATH		Associated Registers	75
PCL Register31, 32, 33, 34, 35, 36, 37, 38,		Registers	
PCLATH Register 31, 32, 33, 34, 35, 36, 37, 38,		ADCON0 (ADC Control 0)	155
PCON Register		ADCON1 (ADC Control 1)	
PIE1 Register		ADRESH (ADC Result High) with ADFM = 0)	
	02, 01	ADRESH (ADC Result High) with ADEM = 1)	

	. 157	PORTB	131
ADRESL (ADC Result Low) with ADFM = 1)	. 158	PORTC	136
ANSELA (PORTA Analog Select)	. 126	PSTRxCON (PWM Steering Control)	240
ANSELB (PORTB Analog Select)	. 132	PWMxCON (Enhanced PWM Control)	239
ANSELC (PORTC Analog Select)	. 137	RCREG	310
APFCON0 (Alternate Pin Function Control 0)	. 121	RCSTA (Receive Status and Control)	303
APFCON1 (Alternate Pin Function Control 1)		SPBRGH	
BAUDCON (Baud Rate Control)		SPBRGL	
BORCON Brown-out Reset Control)		Special Function, Summary	
CCPTMRS (PWM Timer Selection Control)		SRCON0 (SR Latch Control 0)	
CCPxAS (CCPx Auto-Shutdown Control)		SRCON1 (SR Latch Control 1)	
CCPxCON (ECCPx Control)		SSPxADD (MSSPx Address and Baud Rate,	
CLKRCON (Reference Clock Control)		I ² C Mode)	292
CMOUT (Comparator Output)		SSPxCON1 (MSSPx Control 1)	
CMxCON0 (Cx Control)		SSPxCON2 (SSPx Control 2)	
CMxCON1 (Cx Control 1)		SSPxCON3 (SSPx Control 3)	
,			
Configuration Word 1		SSPxMSK (SSPx Mask)	
Configuration Word 2		SSPxSTAT (SSPx Status)	
CPSCON0 (Capacitive Sensing Control Register 0)		STATUS	
CPSCON1 (Capacitive Sensing Control Register 1)		T1CON (Timer1 Control)	
DACCON0		T1GCON (Timer1 Gate Control)	
DACCON1		TRISA (Tri-State PORTA)	
EEADRL (EEPROM Address)		TRISB (Tri-State PORTB)	
EECON1 (EEPROM Control 1)		TRISC (Tri-State PORTC)	136
EECON2 (EEPROM Control 2)		TXCON	201
EEDATH (EEPROM Data)	. 116	TXSTA (Transmit Status and Control)	302
EEDATL (EEPROM Data)	. 116	WDTCON (Watchdog Timer Control)	103
FVRCON	. 146	WPUA (Weak Pull-up PORTA)	127
INLVLA (Input Level Control PORTA)	. 127	WPUB (Weak Pull-up PORTB)	132
INLVLB (Input Level Control PORTB)		WPUC (Weak Pull-up PORTC)	137
INLVLC (Input Level Control PORTC)	. 138	RESET	
INTCON (Interrupt Control)		Reset	77
IOCAF (Interrupt-on-Change PORTA Flag)		Reset Instruction	
IOCAN (Interrupt-on-Change PORTA		Resets	
• •			
Nedalive Edde)	. 141	Associated Registers	84
Negative Edge)	. 141	Associated Registers	
IOCAP (Interrupt-on-Change PORTA		Revision History	
IOCAP (Interrupt-on-Change PORTA Positive Edge)	. 140		
IOCAP (Interrupt-on-Change PORTA Positive Edge) IOCBF (Interrupt-on-Change PORTB Flag)	. 140	Revision HistoryS	437
IOCAP (Interrupt-on-Change PORTA Positive Edge) IOCBF (Interrupt-on-Change PORTB Flag) IOCBN (Interrupt-on-Change PORTB	. 140 . 142	S Shoot-through Current	437
IOCAP (Interrupt-on-Change PORTA Positive Edge) IOCBF (Interrupt-on-Change PORTB Flag) IOCBN (Interrupt-on-Change PORTB Negative Edge)	. 140 . 142	S Shoot-through Current	437 232 413
IOCAP (Interrupt-on-Change PORTA Positive Edge) IOCBF (Interrupt-on-Change PORTB Flag) IOCBN (Interrupt-on-Change PORTB Negative Edge) IOCBP (Interrupt-on-Change PORTB	. 140 . 142 . 142	Shoot-through Current	437 232 413
IOCAP (Interrupt-on-Change PORTA Positive Edge) IOCBF (Interrupt-on-Change PORTB Flag) IOCBN (Interrupt-on-Change PORTB Negative Edge) IOCBP (Interrupt-on-Change PORTB Positive Edge)	. 140 . 142 . 142 . 141	Revision History S Shoot-through Current Software Simulator (MPLAB SIM) SPBRG Register SPBRGH Register	437 232 413 34
IOCAP (Interrupt-on-Change PORTA Positive Edge) IOCBF (Interrupt-on-Change PORTB Flag) IOCBN (Interrupt-on-Change PORTB Negative Edge) IOCBP (Interrupt-on-Change PORTB Positive Edge) LATA (Data Latch PORTA)	. 140 . 142 . 142 . 141 . 126	Revision History S Shoot-through Current	437 232 413 34 305
IOCAP (Interrupt-on-Change PORTA Positive Edge)	. 140 . 142 . 142 . 141 . 126 . 131	Revision History	437 232 413 34 305 153
IOCAP (Interrupt-on-Change PORTA Positive Edge)	. 140 . 142 . 142 . 141 . 126 . 131	Revision History	437 232 413 34 305 153
IOCAP (Interrupt-on-Change PORTA Positive Edge)	. 140 . 142 . 142 . 141 . 126 . 131 . 136	Revision History	437 232 413 305 305 153
IOCAP (Interrupt-on-Change PORTA Positive Edge)	. 140 . 142 . 142 . 141 . 126 . 131 . 136	Revision History	437 232 413 305 305 153 251
IOCAP (Interrupt-on-Change PORTA Positive Edge)	. 140 . 142 . 142 . 141 . 126 . 131 . 136	Revision History	437 232 413 305 305 153 251 247
IOCAP (Interrupt-on-Change PORTA Positive Edge)	. 140 . 142 . 142 . 141 . 126 . 131 . 136 . 210)211 . 208	Revision History	437 232 413 345 305 153 251 247 169
IOCAP (Interrupt-on-Change PORTA Positive Edge)	. 140 . 142 . 142 . 141 . 126 . 131 . 136 . 210)211 . 208 . 209	Revision History	437 232 413 345 305 153 251 247 169 173
IOCAP (Interrupt-on-Change PORTA Positive Edge)	. 140 . 142 . 142 . 141 . 126 . 131 . 136 . 210)211 . 208 . 209 . 185	Revision History	437 232 413 305 305 153 251 247 169 173
IOCAP (Interrupt-on-Change PORTA Positive Edge)	. 140 . 142 . 142 . 141 . 126 . 131 . 136 . 210)211 . 208 . 209 . 185 70	Revision History	437 232 413 305 305 153 251 247 169 173 171
IOCAP (Interrupt-on-Change PORTA Positive Edge)	. 140 . 142 . 142 . 141 . 126 . 131 . 136 . 210)211 . 208 . 209 . 185 70 71	Revision History	437 232 413 305 305 153 251 247 169 173 171 172
IOCAP (Interrupt-on-Change PORTA Positive Edge)	. 140 . 142 . 142 . 141 . 126 . 131 . 136 . 210)211 . 208 . 209 . 185 70 71	Revision History	437 232 413 305 305 153 251 247 169 173 171 172 35
IOCAP (Interrupt-on-Change PORTA Positive Edge)	. 140 . 142 . 142 . 141 . 126 . 131 . 136 . 210)211 . 208 . 209 . 185 70 71 72 83	Revision History	437 232 413 305 305 153 251 247 169 173 171 172 35
IOCAP (Interrupt-on-Change PORTA Positive Edge)	. 140 . 142 . 142 . 141 . 126 . 131 . 136 . 210)211 . 208 . 209 . 185 70 71 72 83 83	Revision History	437 232 413 305 305 153 251 247 169 171 172 35 35
IOCAP (Interrupt-on-Change PORTA Positive Edge)	. 140 . 142 . 142 . 141 . 126 . 131 . 136 . 210)211 . 208 . 209 . 185 70 71 72 83 83	Revision History	437 232 413 305 305 31 251 251 163 171 172 35 35
IOCAP (Interrupt-on-Change PORTA Positive Edge)	. 140 . 142 . 142 . 141 . 126 . 131 . 136 . 210)211 . 208 . 209 . 185 70 71 72 83 83 91	Revision History	437 232 413 305 305 31 251 251 163 171 172 35 35 35
IOCAP (Interrupt-on-Change PORTA Positive Edge)	. 140 . 142 . 142 . 141 . 126 . 131 . 136 . 210)211 . 208 . 209 . 185 70 71 72 83 83 91 92	Revision History	437 232 413 305 305 31 251 251 172 172 35 35 35 35
IOCAP (Interrupt-on-Change PORTA Positive Edge)	. 140 . 142 . 142 . 141 . 126 . 131 . 136 . 210)211 . 208 . 209 . 185 70 71 72 83 83 91 92 93	Revision History	437 232 413 305 305 31 251 251 172 172 35 35 35 35
IOCAP (Interrupt-on-Change PORTA Positive Edge)	. 140 . 142 . 142 . 141 . 126 . 131 . 136 . 210 .)211 . 208 . 209 . 185 70 71 72 83 83 91 92 93 94	Revision History	437 232 413 305 305 31 251 247 169 172 35 35 35 35
IOCAP (Interrupt-on-Change PORTA Positive Edge)	. 140 . 142 . 142 . 141 . 126 . 131 . 136 . 210 .)211 . 208 . 209 . 185 70 71 72 83 83 91 92 93 94 95	Revision History	437 232 413 305 305 31 251 247 169 173 35 35 35 35 35
IOCAP (Interrupt-on-Change PORTA Positive Edge)	. 140 . 142 . 142 . 141 . 126 . 131 . 136 . 210 .)211 . 208 . 209 . 185 70 71 72 83 83 91 92 93 94 95 96	Revision History	437 232 413 305 305 315 251 247 173 171 172 35 35 35 35 35 35
IOCAP (Interrupt-on-Change PORTA Positive Edge)	. 140 . 142 . 142 . 141 . 126 . 131 . 136 . 210)211 . 208 . 209 . 185 70 71 72 83 83 91 92 93 94 95 96 97	S Shoot-through Current Software Simulator (MPLAB SIM) SPBRG Register SPBRGH Register SPBRGL Register Special Event Trigger Special Function Registers (SFRs) SPI Mode (MSSPx) Associated Registers SPI Clock SR Latch Associated registers w/ SR Latch SRCON0 Register SRCON1 Register SSP1ADD Register SSP1ADD Register SSP1CON1 Register SSP1CON2 Register SSP1CON3 Register SSP1CON3 Register SSP1STAT Register SSP2ADD Register SSP2ADD Register SSP2BUF Register SSP2BUF Register SSP2BUF Register SSP2BUF Register SSP2BUF Register	437 232 413 305 305 315 251 247 173 171 172 35 35 35 35 35 35 35
IOCAP (Interrupt-on-Change PORTA Positive Edge)	. 140 . 142 . 142 . 141 . 126 . 131 . 136 . 210)211 . 208 . 209 . 185 70 71 72 83 83 91 92 93 94 95 96 97 98	Revision History	437 232 413 305 305 315 251 247 173 171 172 35 35 35 35 35 35 35

SSPOV Status Flag	. 278	Brown-out Reset Situations	79
SSPxADD Register	. 292	Bus Collision During a Repeated Start Condition	
SSPxCON1 Register	. 289	(Case 1)	284
SSPxCON2 Register	. 290	Bus Collision During a Repeated Start Condition	
SSPxCON3 Register	. 291	(Case 2)	284
SSPxMSK Register	. 292	Bus Collision During a Start Condition (SCL = 0)	283
SSPxSTAT Register	. 288	Bus Collision During a Stop Condition (Case 1)	285
R/W Bit	. 257	Bus Collision During a Stop Condition (Case 2)	285
Stack	43	Bus Collision During Start Condition (SDA only)	
Accessing	43	Bus Collision for Transmit and Acknowledge	
Reset	45	CLKOUT and I/O	
Stack Overflow/Underflow	80	Clock Synchronization	270
STATUS Register	24	Clock Timing	
SUBWFB	. 345	Comparator Output	
_		Enhanced Capture/Compare/PWM (ECCP)	367
Т		Fail-Safe Clock Monitor (FSCM)	
T1CON Register31	, 195	First Start Bit Timing	274
T1GCON Register	. 196	Full-Bridge PWM Output	
T2CON Register	31	Half-Bridge PWM Output225,	
T4CON Register	39	I ² C Bus Data	
T6CON Register	39	I ² C Bus Start/Stop Bits	
Temperature Indicator Module	. 147	I ² C Master Mode (7 or 10-Bit Transmission)	
Thermal Considerations	. 360	I ² C Master Mode (7-Bit Reception)	
Timer0	. 183	I ² C Stop Condition Receive or Transmit Mode	
Associated Registers		INT Pin Interrupt	
Operation		Internal Oscillator Switch Timing	
Specifications		PWM Auto-shutdown	
Timer1	. 187	Firmware Restart	
Associated registers	. 197	PWM Direction Change	228
Asynchronous Counter Mode		PWM Direction Change at Near 100% Duty Cycle	
Reading and Writing		PWM Output (Active-High)	
Clock Source Selection		PWM Output (Active-Low)	
Interrupt		Repeat Start Condition	
Operation		Reset Start-up Sequence	
Operation During Sleep		Reset, WDT, OST and Power-up Timer	
Oscillator		Send Break Character Sequence	
Prescaler		SPI Master Mode (CKE = 1, SMP = 1)	
Specifications		SPI Mode (Master Mode)	
Timer1 Gate		SPI Slave Mode (CKE = 0)	
Selecting Source	. 189	SPI Slave Mode (CKE = 1)	
TMR1H Register		Synchronous Reception (Master Mode, SREN)	
TMR1L Register		Synchronous Transmission	
Timer2		Synchronous Transmission (Through TXEN)	
Associated registers	.202	Timer0 and Timer1 External Clock	
Timer2/4/6		Timer1 Incrementing Edge	
Associated registers		Two Speed Start-up	
Timers		USART Synchronous Receive (Master/Slave)	
Timer1		USART Synchronous Transmission (Master/Slave).	
T1CON	195	Wake-up from Interrupt	
T1GCON		iming Diagrams and Specifications	100
Timer2/4/6		PLL Clock	363
TXCON	201 т	iming Parameter Symbology	
Timing Diagrams		iming Requirements	501
A/D Conversion		I ² C Bus Data	376
A/D Conversion (Sleep Mode)		SPI Mode	
Acknowledge Sequence		INLVLC Register	
Asynchronous Reception		MR0 Register	
Asynchronous Transmission		•	
Asynchronous Transmission (Back to Back)	· · · · · · · · · · · · · · · · · ·	MR1H Register MR1L Register	
Auto Wake-up Bit (WUE) During Normal Operation			
Auto Wake-up Bit (WUE) During Normal Operation Auto Wake-up Bit (WUE) During Sleep		MR2 Register	
Auto wake-up bit (WOE) builing sleep		MR4 Register	
Baud Rate Generator with Clock Arbitration	273 +	MR6 Register	
BRG Reset Due to SDA Arbitration During Start		RIS	
Condition		RISA Register	
Brown-out Reset (BOR)	365	NOD	123

TRISB Register	32, 131
TRISC	134
TRISC Register3	
Two-Speed Clock Start-up Mode	66
TXCON (Timer2/4/6) Register	
TXREG	
TXREG Register	34
TXSTA Register3	
BRGH Bit	305
U	
USART	
Synchronous Master Mode	
Requirements, Synchronous Receive	371
Requirements, Synchronous Transmission	
Timing Diagram, Synchronous Receive	
Timing Diagram, Synchronous Transmission	
V	
•	
VREF. SEE ADC Reference Voltage	
W	
Wake-up on Break	311
Wake-up Using Interrupts	
Watchdog Timer (WDT)	
Associated Registers	104
Modes	102
Specifications	366
WCOL273, 276, 27	'8, 280
WCOL Status Flag273, 276, 27	'8, 280
WDTCON Register	103
WPUA Register	127
WPUB Register	
WPUC Register	
Write Protection	
WWW Address	
M/M/M/ On Line Support	10

NOTES:

THE MICROCHIP WEB SITE

Microchip provides online support via our WWW site at www.microchip.com. This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- Product Support Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- General Technical Support Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- Business of Microchip Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at www.microchip.com. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- · Distributor or Representative
- · Local Sales Office
- Field Application Engineer (FAE)
- · Technical Support
- · Development Systems Information Line

Customers should contact their distributor, representative or field application engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the web site at: http://microchip.com/support

READER RESPONSE

It is our intention to provide you with the best documentation possible to ensure successful use of your Microchip product. If you wish to provide your comments on organization, clarity, subject matter, and ways in which our documentation can better serve you, please FAX your comments to the Technical Publications Manager at (480) 792-4150.

Please list the following information, and use this outline to provide us with your comments about this document.

TO: RE:	Technical Publications Manager Reader Response	Total Pages Sent
	•	
FIOIII.	Name	
	CompanyAddress	
	City / State / ZIP / Country	
	Telephone: (FAX: ()
Applic	ation (optional):	
Would	l you like a reply? Y N	
Device	e: PIC16(L)F1825/1829	Literature Number: DS41440C
Quest	ions:	
1. W	hat are the best features of this document?	
_		
2. H	ow does this document meet your hardware and softw	vare development needs?
_		
3. D	o you find the organization of this document easy to fo	ollow? If not, why?
_		
4. W	hat additions to the document do you think would enh	ance the structure and subject?
_		
5. W	that deletions from the decument could be made within	out offseting the everall usefulness?
S. VV	hat deletions from the document could be made witho	out affecting the overall usefulless?
_		
6. Is	there any incorrect or misleading information (what a	nd where)?
_		
7. H	ow 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.

PART NO. Device	X	attern	Examples: a) PIC16F1825 - E/SL 301 = Extended temp., SOIC package, QTP pattern #301. b) PIC16LF1829 - E/SS = Extended temp., SSOP package.
Device:	PIC16F1825, PIC16LF1825 PIC16F1829T, PIC16LF1829		c) PIC16LF1829 - E/ML= Extended temp., QFN package.
Tape and Reel Option:	Blank = Standard packaging (tube or tray) T = Tape and Reel ⁽¹⁾		
Temperature Range:	I = -40° C to $+85^{\circ}$ C (Industrial) E = -40° C to $+125^{\circ}$ C (Extended)		
Package: ⁽²⁾	ML = QFN P = PDIP SL = SOIC SO = SOIC SS = SSOP ST = TSSOP		Note 1: Tape and Reel identifier only appears in the catalog part number description. This identifier is used for ordering purposes and is not printed on the device package. Check with your Microchip Sales Office for package availability with the Tape and Reel option. 2: For other small form-factor package
Pattern:	QTP, SQTP, Code or Special Requirements (blank otherwise)		availability and marking information, please visit www.microchip.com/packaging or contact your local sales office.



Worldwide Sales and Service

AMERICAS

Corporate Office

2355 West Chandler Blvd. Chandler, AZ 85224-6199 Tel: 480-792-7200 Fax: 480-792-7277

Technical Support:

http://www.microchip.com/

support

Web Address:

www.microchip.com

Atlanta

Duluth, GA Tel: 678-957-9614

Fax: 678-957-1455

Boston

Westborough, MA Tel: 774-760-0087 Fax: 774-760-0088

Chicago Itasca, IL

Tel: 630-285-0071 Fax: 630-285-0075

Cleveland

Independence, OH Tel: 216-447-0464 Fax: 216-447-0643

Dallas

Addison, TX Tel: 972-818-7423 Fax: 972-818-2924

Detroit

Farmington Hills, MI Tel: 248-538-2250 Fax: 248-538-2260

Indianapolis

Noblesville, IN Tel: 317-773-8323 Fax: 317-773-5453

Los Angeles

Mission Viejo, CA Tel: 949-462-9523 Fax: 949-462-9608

Santa Clara

Santa Clara, CA Tel: 408-961-6444 Fax: 408-961-6445

Toronto

Mississauga, Ontario,

Canada

Tel: 905-673-0699 Fax: 905-673-6509

ASIA/PACIFIC

Asia Pacific Office

Suites 3707-14, 37th Floor Tower 6, The Gateway Harbour City, Kowloon

Hong Kong Tel: 852-2401-1200

Fax: 852-2401-3431

Australia - Sydney

Tel: 61-2-9868-6733 Fax: 61-2-9868-6755

China - Beijing

Tel: 86-10-8569-7000 Fax: 86-10-8528-2104

China - Chengdu

Tel: 86-28-8665-5511 Fax: 86-28-8665-7889

China - Chongqing

Tel: 86-23-8980-9588 Fax: 86-23-8980-9500

China - Hangzhou

Tel: 86-571-2819-3187 Fax: 86-571-2819-3189

China - Hong Kong SAR

Tel: 852-2401-1200 Fax: 852-2401-3431

China - Nanjing

Tel: 86-25-8473-2460 Fax: 86-25-8473-2470

China - Qingdao

Tel: 86-532-8502-7355 Fax: 86-532-8502-7205

China - Shanghai

Tel: 86-21-5407-5533 Fax: 86-21-5407-5066

China - Shenyang

Tel: 86-24-2334-2829 Fax: 86-24-2334-2393

China - Shenzhen

Tel: 86-755-8203-2660 Fax: 86-755-8203-1760

China - Wuhan

Tel: 86-27-5980-5300 Fax: 86-27-5980-5118

China - Xian

Tel: 86-29-8833-7252 Fax: 86-29-8833-7256

China - Xiamen

Tel: 86-592-2388138 Fax: 86-592-2388130

China - Zhuhai

Tel: 86-756-3210040 Fax: 86-756-3210049

ASIA/PACIFIC

India - Bangalore

Tel: 91-80-3090-4444 Fax: 91-80-3090-4123

India - New Delhi

Tel: 91-11-4160-8631 Fax: 91-11-4160-8632

India - Pune

Tel: 91-20-2566-1512 Fax: 91-20-2566-1513

Japan - Osaka

Tel: 81-66-152-7160 Fax: 81-66-152-9310

Japan - Yokohama

Tel: 81-45-471- 6166 Fax: 81-45-471-6122

Korea - Daegu

Tel: 82-53-744-4301 Fax: 82-53-744-4302

Korea - Seoul

Tel: 82-2-554-7200 Fax: 82-2-558-5932 or 82-2-558-5934

Malaysia - Kuala Lumpur

Tel: 60-3-6201-9857 Fax: 60-3-6201-9859

Malaysia - Penang

Tel: 60-4-227-8870 Fax: 60-4-227-4068

Philippines - Manila

Tel: 63-2-634-9065 Fax: 63-2-634-9069

Singapore

Tel: 65-6334-8870 Fax: 65-6334-8850

Taiwan - Hsin Chu

Tel: 886-3-5778-366 Fax: 886-3-5770-955

Taiwan - Kaohsiung

Tel: 886-7-536-4818 Fax: 886-7-330-9305

Taiwan - Taipei

Tel: 886-2-2500-6610 Fax: 886-2-2508-0102

Thailand - Bangkok Tel: 66-2-694-1351

Fax: 66-2-694-1350

EUROPE

Austria - Wels

Tel: 43-7242-2244-39 Fax: 43-7242-2244-393

Denmark - Copenhagen

Tel: 45-4450-2828 Fax: 45-4485-2829

France - Paris

Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79

Germany - Munich

Tel: 49-89-627-144-0 Fax: 49-89-627-144-44

Italy - Milan

Tel: 39-0331-742611 Fax: 39-0331-466781

Netherlands - Drunen

Tel: 31-416-690399 Fax: 31-416-690340

Spain - Madrid

Tel: 34-91-708-08-90 Fax: 34-91-708-08-91

UK - Wokingham Tel: 44-118-921-5869 Fax: 44-118-921-5820

11/29/11



Мы молодая и активно развивающаяся компания в области поставок электронных компонентов. Мы поставляем электронные компоненты отечественного и импортного производства напрямую от производителей и с крупнейших складов мира.

Благодаря сотрудничеству с мировыми поставщиками мы осуществляем комплексные и плановые поставки широчайшего спектра электронных компонентов.

Собственная эффективная логистика и склад в обеспечивает надежную поставку продукции в точно указанные сроки по всей России.

Мы осуществляем техническую поддержку нашим клиентам и предпродажную проверку качества продукции. На все поставляемые продукты мы предоставляем гарантию .

Осуществляем поставки продукции под контролем ВП МО РФ на предприятия военно-промышленного комплекса России, а также работаем в рамках 275 ФЗ с открытием отдельных счетов в уполномоченном банке. Система менеджмента качества компании соответствует требованиям ГОСТ ISO 9001.

Минимальные сроки поставки, гибкие цены, неограниченный ассортимент и индивидуальный подход к клиентам являются основой для выстраивания долгосрочного и эффективного сотрудничества с предприятиями радиоэлектронной промышленности, предприятиями ВПК и научноисследовательскими институтами России.

С нами вы становитесь еще успешнее!

Наши контакты:

Телефон: +7 812 627 14 35

Электронная почта: sales@st-electron.ru

Адрес: 198099, Санкт-Петербург,

Промышленная ул, дом № 19, литера Н,

помещение 100-Н Офис 331