



Welcome to [E-XFL.COM](https://www.e-xfl.com)

What is "[Embedded - Microcontrollers](#)"?

"[Embedded - Microcontrollers](#)" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "[Embedded - Microcontrollers](#)"

Details

Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	32MHz
Connectivity	I ² C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	12
Program Memory Size	14KB (8K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 8x10b; D/A 1x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	16-VQFN Exposed Pad
Supplier Device Package	16-QFN (4x4)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf1615-e-ml

TABLE 4: 20-PIN ALLOCATION TABLE (PIC16(L)F1619)

I/O	20-Pin PDIP, SOIC, SSOP	20-Pin UQFN	A/D	Reference	Comparator	Timers	CCP	CWG	ZCD	CLC	EUSART	SMT	Angular Timer	MSSP	PWM	High Current I/O	Interrupt	Pull-up	Basic
RA0	19	16	AN0	DAC1OUT	C1IN+	—	—	—	—	—	—	—	—	—	—	—	IOC	Y	ICSPDAT
RA1	18	15	AN1	VREF+	C1IN0- C2IN0-	—	—	—	—	—	—	—	—	—	—	—	IOC	Y	ICSPCLK
RA2	17	14	AN2	—	—	T0CKI ⁽¹⁾	—	CWG1IN ⁽¹⁾	ZCD1IN	—	—	—	—	—	—	—	INT IOC	Y	—
RA3	4	1	—	—	—	T6IN ⁽¹⁾	—	—	—	—	—	SMTWIN2 ⁽¹⁾	—	—	—	—	IOC	Y	MCLR VPP
RA4	3	20	AN3	—	—	T1G ⁽¹⁾	—	—	—	—	—	SMTSIG1 ⁽¹⁾	—	—	—	—	IOC	Y	CLKOUT
RA5	2	19	—	—	—	T1CKI ⁽¹⁾ T2IN ⁽¹⁾	—	—	—	CLCIN3 ⁽¹⁾	—	SMTWIN1 ⁽¹⁾	—	—	—	—	IOC	Y	CLKIN
RB4	13	10	AN10	—	—	—	—	—	—	—	—	—	—	SDI ⁽¹⁾	—	—	IOC	Y	—
RB5	12	9	AN11	—	—	—	—	—	—	—	RX ^(1,3)	—	—	—	—	—	IOC	Y	—
RB6	11	8	—	—	—	—	—	—	—	—	—	—	—	SCK ^(1,3)	—	—	IOC	Y	—
RB7	10	7	—	—	—	—	—	—	—	—	CK ⁽¹⁾	—	—	—	—	—	IOC	Y	—
RC0	16	13	AN4	—	C2IN+	T5CKI ⁽¹⁾	—	—	—	—	—	—	—	—	—	—	IOC	Y	—
RC1	15	12	AN5	—	C1IN1- C2IN1-	T4IN ⁽¹⁾	—	—	—	CLCIN2 ⁽²⁾	—	SMTSIG2 ⁽¹⁾	—	—	—	—	IOC	Y	—
RC2	14	11	AN6	—	C1IN2- C2IN2-	—	—	—	—	—	—	—	—	—	—	—	IOC	Y	—
RC3	7	4	AN7	—	C1IN3- C2IN3-	T5G ⁽¹⁾	CCP2 ⁽¹⁾	—	—	CLCIN0 ⁽¹⁾	—	—	ATCC ⁽¹⁾	—	—	—	IOC	Y	—
RC4	6	3	—	—	—	T3G ⁽¹⁾	—	—	—	CLCIN1 ⁽¹⁾	—	—	—	—	—	HIC4	IOC	Y	—
RC5	5	2	—	—	—	T3CKI ⁽¹⁾	CCP1 ⁽¹⁾	—	—	—	—	—	ATIN ⁽¹⁾	—	—	HIC5	IOC	Y	—
RC6	8	5	AN8	—	—	—	—	—	—	—	—	—	—	SS ⁽¹⁾	—	—	IOC	Y	—
RC7	9	6	AN9	—	—	—	—	—	—	—	—	—	—	—	—	—	IOC	Y	—
VDD	1	18	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
VSS	20	17	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

- Note**
- 1: Default peripheral input. Input can be moved to any other pin with the PPS input selection registers.
 - 2: All pin outputs default to PORT latch data. Any pin can be selected as a digital peripheral output with the PPS output selection registers.
 - 3: These peripheral functions are bidirectional. The output pin selections must be the same as the input pin selections.

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

FIGURE 2-1: CORE BLOCK DIAGRAM

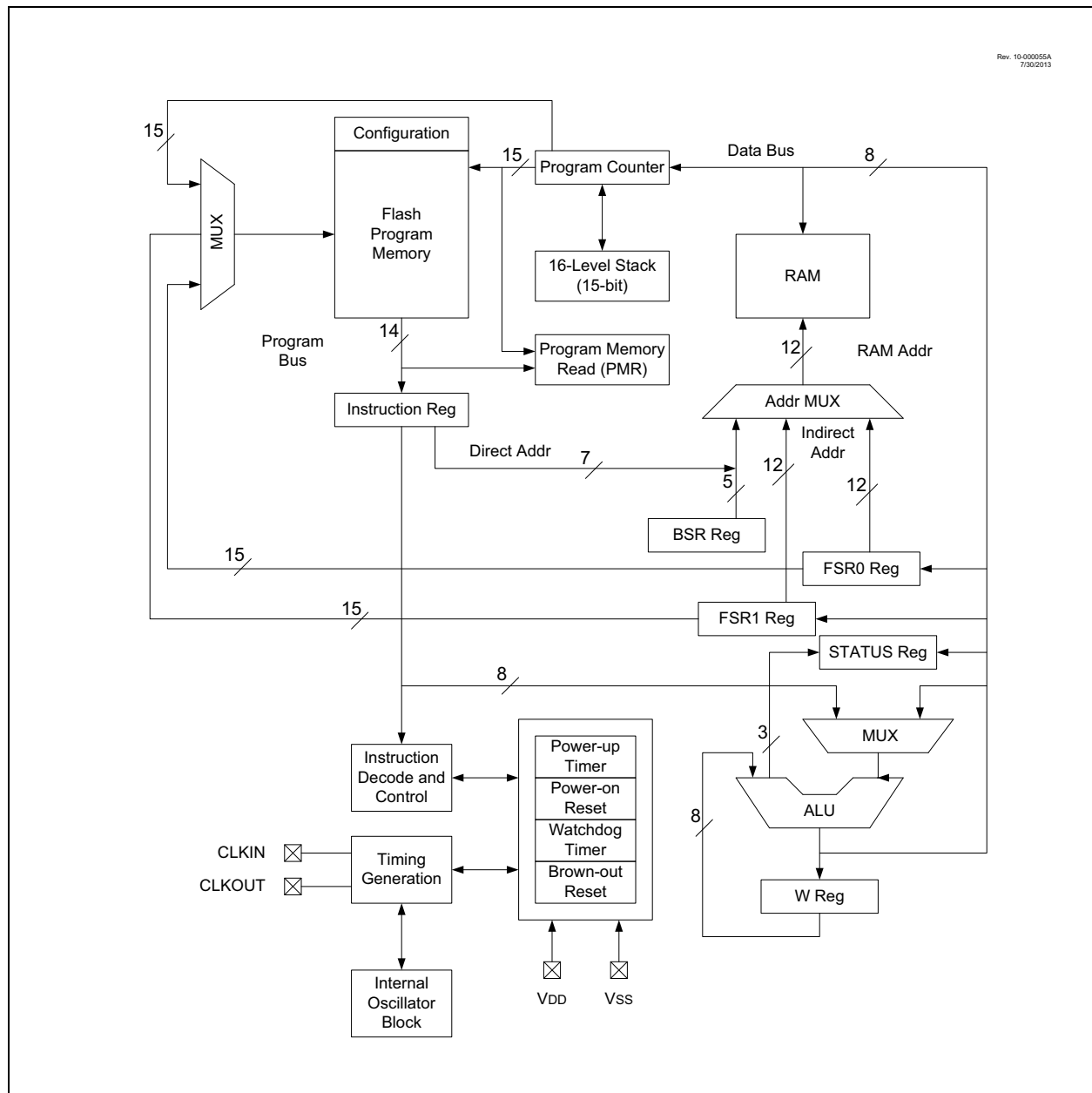


TABLE 3-14: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Addr	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 1											
08Ch	TRISA	—	—	TRISA5	TRISA4	— ⁽²⁾	TRISA2	TRISA1	TRISA0	--11 1111	--11 1111
08Dh	TRISB ⁽⁴⁾	TRISB7	TRISB6	TRISB5	TRISB4	—	—	—	—	1111 ----	1111 ----
08Eh	TRISC	TRISC7 ⁽⁴⁾	TRISC6 ⁽⁴⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111
08Fh	—	Unimplemented								—	—
090h	—	Unimplemented								—	—
090h	PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
091h	PIE2	OSFIE	C2IE	C1IE	—	BCL1IE	TMR6IE	TMR4IE	CCP2IE	-00- 0000	-00- 0000
092h	PIE3	—	—	CWGIE	ZCDIE	CLC4IE	CLC3IE	CLC2IE	CLC1IE	--00 0000	--00 0000
093h	PIE4	SCANIE	CRCIE	SMT2PWAIE	SMT2PRAIE	SMT2IE	SMT1PWAIE	SMT1PRAIE	SMT1IE	0000 0000	0000 0000
094h	PIE5	TMR3GIE	TMR3IE	TMR5GIE	TMR5IE	—	AT1IE	PID1EIE	PID1DIE	0000 -000	0000 -000
095h	OPTION_REG	WPUEN	INTEDG	TMR0CS	TMR0SE	PSA	PS<2:0>			1111 1111	1111 1111
096h	PCON	STKOVF	STKUNF	WDTWV	RWD \overline{T}	RMCLR	R \overline{I}	POR	BOR	00-1 11qq	qq-q qqqu
097h	—	Unimplemented								—	—
098h	OSCTUNE	—	—	TUN<5:0>						--00 0000	--00 0000
099h	OSCCON	SPLLEN	IRCF<3:0>				—	SCS<1:0>		0011 1-00	0011 1-00
09Ah	OSCSTAT	—	PLL \overline{R}	OSTS	HFIOFR	HFIOFL	MFIOFR	LFIOFR	HFIOFS	-000 0000	-qqq qqqq
09Bh	ADRESL	ADC Result Register Low								xxxx xxxx	uuuu uuuu
09Ch	ADRESH	ADC Result Register High								xxxx xxxx	uuuu uuuu
09Dh	ADCON0	—	CHS<4:0>					GO/ \overline{DONE}	ADON	-000 0000	-000 0000
09Eh	ADCON1	ADFM	ADCS<2:0>			—	—	ADPREF<1:0>		0000 --00	0000 --00
09Fh	ADCON2	TRIGSEL<4:0>					—	—	—	0000 0---	0000 0---

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

- Note** 1: PIC16F1615/9 only.
 2: Unimplemented, read as '1'.
 3: PIC16(L)F1615 only.
 4: PIC16(L)F1619 only.

TABLE 3-14: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Addr	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 30 (Continued)											
F24h	CLC3CON	LC3EN	—	LC3OUT	LC3INTP	LC3INTN	LC3MODE<2:0>			0-x0 0000	0-x0 0000
F25h	CLC3POL	LC3POL	—	—	—	LC3G4POL	LC3G3POL	LC3G2POL	LC3G1POL	x--- xxxx	x--- xxxx
F26h	CLC3SEL0	—	—	LC3D1S<5:0>						--xx xxxx	--xx xxxx
F27h	CLC3SEL1	—	—	LC3D2S<5:0>						--xx xxxx	--xx xxxx
F28h	CLC3SEL2	—	—	LC3D3S<5:0>						--xx xxxx	--xx xxxx
F29h	CLC3SEL3	—	—	LC3D4S<5:0>						--xx xxxx	--xx xxxx
F2Ah	CLC3GLS0	LC3G1D4T	LC3G1D4N	LC3G1D3T	LC3G1D3N	LC3G1D2T	LC3G1D2N	LC3G1D1T	LC3G1D1N	xxxx xxxx	xxxx xxxx
F2Bh	CLC3GLS1	LC3G2D4T	LC3G2D4N	LC3G2D3T	LC3G2D3N	LC3G2D2T	LC3G2D2N	LC3G2D1T	LC3G2D1N	xxxx xxxx	xxxx xxxx
F2Ch	CLC3GLS2	LC3G3D4T	LC3G3D4N	LC3G3D3T	LC3G3D3N	LC3G3D2T	LC3G3D2N	LC3G3D1T	LC3G3D1N	xxxx xxxx	xxxx xxxx
F2Dh	CLC3GLS3	LC3G4D4T	LC3G4D4N	LC3G4D3T	LC3G4D3N	LC3G4D2T	LC3G4D2N	LC3G4D1T	LC3G4D1N	xxxx xxxx	xxxx xxxx
F2Eh	CLC4CON	LC4EN	—	LC4OUT	LC4INTP	LC4INTN	LC4MODE<2:0>			0-x0 0000	0-x0 0000
F2Fh	CLC4POL	LC4POL	—	—	—	LC4G4POL	LC4G3POL	LC4G2POL	LC4G1POL	x--- xxxx	x--- xxxx
F30h	CLC4SEL0	—	—	LC4D1S<5:0>						--xx xxxx	--xx xxxx
F31h	CLC4SEL1	—	—	LC4D2S<5:0>						--xx xxxx	--xx xxxx
F32h	CLC4SEL2	—	—	LC4D3S<5:0>						--xx xxxx	--xx xxxx
F33h	CLC4SEL3	—	—	LC4D4S<5:0>						--xx xxxx	--xx xxxx
F34h	CLC4GLS0	LC4G1D4T	LC4G1D4N	LC4G1D3T	LC4G1D3N	LC4G1D2T	LC4G1D2N	LC4G1D1T	LC4G1D1N	xxxx xxxx	xxxx xxxx
F35h	CLC4GLS1	LC4G2D4T	LC4G2D4N	LC4G2D3T	LC4G2D3N	LC4G2D2T	LC4G2D2N	LC4G2D1T	LC4G2D1N	xxxx xxxx	xxxx xxxx
F36h	CLC4GLS2	LC4G3D4T	LC4G3D4N	LC4G3D3T	LC4G3D3N	LC4G3D2T	LC4G3D2N	LC4G3D1T	LC4G3D1N	xxxx xxxx	xxxx xxxx
F37h	CLC4GLS3	LC4G4D4T	LC4G4D4N	LC4G4D3T	LC4G4D3N	LC4G4D2T	LC4G4D2N	LC4G4D1T	LC4G4D1N	xxxx xxxx	xxxx xxxx
F38h to F6Fh	—	Unimplemented								—	—

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

- Note**
- 1: PIC16F1615/9 only.
 - 2: Unimplemented, read as '1'.
 - 3: PIC16(L)F1615 only.
 - 4: PIC16(L)F1619 only.

REGISTER 4-3: CONFIG3: CONFIGURATION WORD 3

R/P-0	R/P-0	R/P-1	R/P-1	R/P-1	R/P-1
WDTCSS<2:0>			WDTCWS<2:0>		
bit 13			bit 8		

U-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	WDTE<1:0>		WDTCPSS<4:0>				
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 13-11 **WDTCSS<2:0>**: WDT Configuration Clock Select bits
 111 =Software Control; WDT clock selected by CS<2:0>
 110 =Reserved
 .
 .
 .
 010 =Reserved
 001 =WDT reference clock is MFINTOSC, 31.25 kHz (default value)
 000 =WDT reference clock is LFINTOSC, 31.00 kHz output

bit 10-8 **WDTCWS<2:0>**: WDT Configuration Window Select bits.

WDTCWS <2:0>	WINDOW at POR			Software control of WINDOW?	Keyed access required?
	Value	Window delay Percent of time	Window opening Percent of time		
111	111	n/a	100	Yes	No
110	111	n/a	100	No	Yes
101	101	25	75		
100	100	37.5	62.5		
011	011	50	50		
010	010	62.5	37.5		
001	001	75	25		
000	000	87.5	12.5 ⁽¹⁾		

Default fuse = 111

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

bit 6-5 **WDTE<1:0>**: Watchdog Timer Enable bits
 11 =WDT enabled in all modes, the SEN bit in the WDTCON0 register is ignored
 10 =WDT enabled while running and disabled in Sleep
 01 =WDT controlled by the SEN bit in the WDTCON0 register
 00 = WDT disabled

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

100000 = Minimum frequency

•

•

•

111111 =

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

000001 =

•

•

•

011110 =

011111 = Maximum frequency

TABLE 5-3: 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<1:0>		89
OSCSTAT	—	PLLRC	OSTS	HFIOFR	HFIOFL	MFIOFR	LFIOFR	HFIOFS	90
OSCTUNE	—	—	TUN<5:0>						91

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

TABLE 5-4: 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
CONFIG1	13:8	—	—	FCMEN	IESO	CLKOUTEN	BOREN<1:0>		—	69
	7:0	CP	MCLRE	PWRTÉ	—	—	FOSC<2:0>			

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

10.2.1 READING THE FLASH PROGRAM MEMORY

To read a program memory location, the user must:

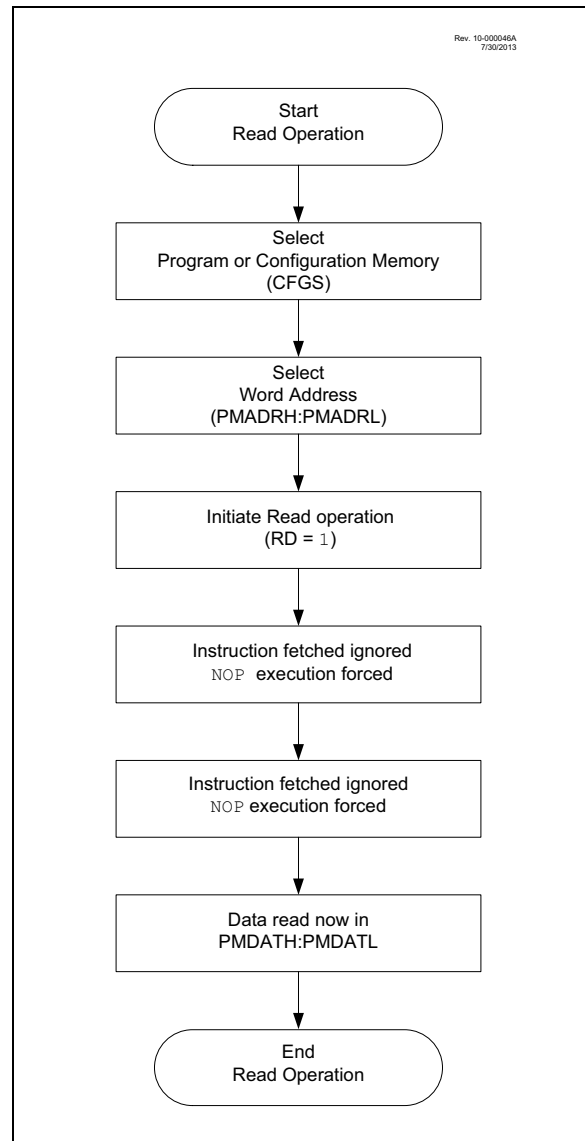
1. Write the desired address to the PMADRH:PMADRL register pair.
2. Clear the CFGS bit of the PMCON1 register.
3. Then, set control bit RD of the PMCON1 register.

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 PMCON1, RD” instruction to be ignored. The data is available in the very next cycle, in the PMDATH:PMDATL register pair; therefore, it can be read as two bytes in the following instructions.

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

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

FIGURE 10-1: FLASH PROGRAM MEMORY READ FLOWCHART



10.6 Register Definitions: Flash Program Memory Control

REGISTER 10-1: PMDATL: PROGRAM MEMORY DATA LOW BYTE 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
PMDAT<7:0>							
bit 7							bit 0

Legend:

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

bit 7-0 **PMDAT<7:0>**: Read/write value for Least Significant bits of program memory

REGISTER 10-2: PMDATH: PROGRAM MEMORY 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
—	—	PMDAT<13:8>					
bit 7							bit 0

Legend:

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

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

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

REGISTER 10-3: PMADRL: PROGRAM MEMORY ADDRESS LOW BYTE 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
PMADR<7:0>							
bit 7							bit 0

Legend:

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

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

REGISTER 10-4: PMADRH: PROGRAM MEMORY 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
—(1)	PMADR<14:8>						
bit 7							bit 0

Legend:

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

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

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

Note 1: Unimplemented, read as '1'.

TABLE 13-3: SUMMARY OF REGISTERS ASSOCIATED WITH THE PPS MODULE (CONTINUED)

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
RC1PPS	—	—	—	RC1PPS<4:0>					180
RC2PPS	—	—	—	RC2PPS<4:0>					180
RC3PPS	—	—	—	RC3PPS<4:0>					180
RC4PPS	—	—	—	RC4PPS<4:0>					180
RC5PPS	—	—	—	RC5PPS<4:0>					180
RC6PPS ⁽¹⁾	—	—	—	RC6PPS<4:0>					180
RC7PPS ⁽¹⁾	—	—	—	RC7PPS<4:0>					180

Note 1: PIC16(L)F1619 only.

REGISTER 17-6: ADRESH: ADC RESULT REGISTER HIGH (ADRESH) ADFM = 1

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<9:8>	
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 17-7: ADRESL: ADC RESULT REGISTER LOW (ADRESL) ADFM = 1

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<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

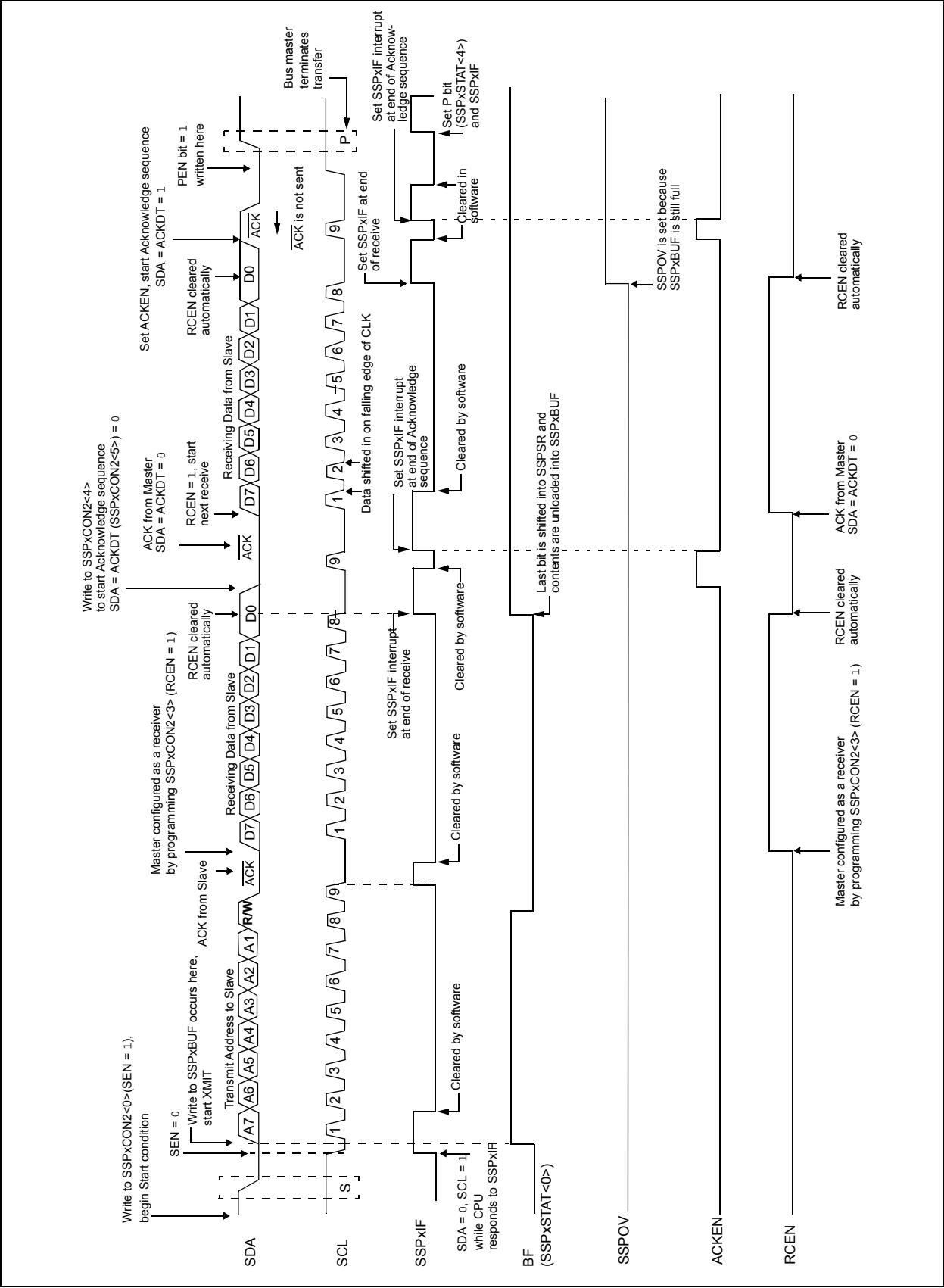
-n/n = Value at POR and BOR/Value at all other Resets

'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

FIGURE 24-29: I²C MASTER MODE WAVEFORM (RECEPTION, 7-BIT ADDRESS)



REGISTER 28-7: CWGxOCON0: CWGx STEERING CONTROL REGISTER⁽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	R/W-0/0
OVRD	OVRC	OVRB	OVRA	STRD ⁽²⁾	STRC ⁽²⁾	STRB ⁽²⁾	STRA ⁽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	q = Value depends on condition

bit 7	OVRD: Steering Data D bit
bit 6	OVRC: Steering Data C bit
bit 5	OVRB: Steering Data B bit
bit 4	OVRA: Steering Data A bit
bit 3	STRD: Steering Enable D bit ⁽²⁾ 1 = CWGxD output has the CWGx_data waveform with polarity control from POLD bit 0 = CWGxD output is assigned the value of OVRD bit
bit 2	STRC: Steering Enable C bit ⁽²⁾ 1 = CWGxC output has the CWGx_data waveform with polarity control from POLC bit 0 = CWGxC output is assigned the value of OVRC bit
bit 1	STRB: Steering Enable B bit ⁽²⁾ 1 = CWGxB output has the CWGx_data waveform with polarity control from POLB bit 0 = CWGxB output is assigned the value of OVRB bit
bit 0	STRA: Steering Enable A bit ⁽²⁾ 1 = CWGxA output has the CWGx_data waveform with polarity control from POLA bit 0 = CWGxA output is assigned the value of OVRA bit

Note 1: The bits in this register apply only when MODE<2:0> = 00x.

2: This bit is effectively double-buffered when MODE<2:0> = 001.

30.6.8 CAPTURE MODE

This mode captures the Timer value based on a rising or falling edge on the SMTWINx input and triggers an interrupt. This mimics the capture feature of a CCP module. The timer begins incrementing upon the SMTxGO bit being set, and updates the value of the SMTxCPR register on each rising edge of SMTWINx, and updates the value of the CPW register on each falling edge of the SMTWINx. The timer is not reset by any hardware conditions in this mode and must be reset by software, if desired. See Figure 30-16 and Figure 30-17.

REGISTER 30-8: SMT2SIG: SMT2 SIGNAL INPUT SELECT REGISTER

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
—	—	—	SSEL<4:0>				
bit 7			bit 0				

31.2.3 MISSING PULSE DETECTION

In both Single-Pulse and Multi-Pulse modes, the AT module monitors for missing pulses in the following manner. The latched value of the ATxPER register pair is continuously subtracted from the value of the period counter as it counts up. The result of this subtraction is compared to a third value and a missing pulse event is generated when the comparison is equal.

The third value is either the ATxMISS register pair or the ATxPER register pair divided by two. The APMOD bit of ATxCON0 register (Register 31-1) selects which of these two values is used.

In Single-Pulse mode, a missing pulse event generates the missing pulse output of the module as well as triggering the MISSIF interrupt.

In Multi-Pulse mode, a missing pulse event generates the output and interrupt, and is also used to determine the period signal timing.

31.2.4 MISSING PULSE MODES

Missing pulse detection has two modes of operation selected with the APMOD bit of the ATxCON0 register:

- Adaptive
- Fixed

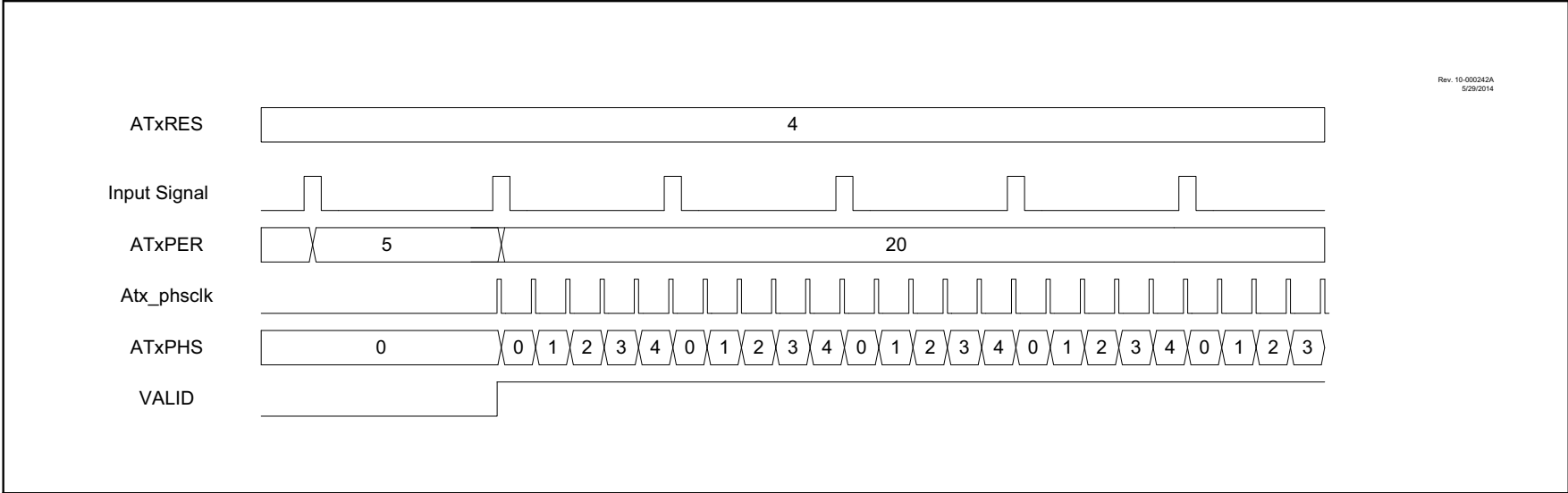
31.2.4.1 Adaptive Missing Pulse Mode

When APMOD = 1, the missing pulse detection is in the Adaptive mode. In Adaptive mode, the difference between the period counter and the latched ATxPER value is compared to the latched ATxPER value divided by two. A missing pulse event will occur when an input signal pulse is not detected within 1.5 times the previous time between pulses. If the signal input period changes, the missing pulse comparison adapts to the change to maintain the relative time to the missing pulse event at 1.5 times the previous pulse interval.

31.2.4.2 Fixed Missing Pulse Mode

When APMOD = 0, the missing pulse detection is in the Fixed mode. In Fixed mode, the difference between the period counter and the latched ATxPER value is compared to the value in the ATxMISS register pair. This gives the user absolute control over when the missing pulse will be detected, with the trade-off of not being adaptive to changes in the period.

FIGURE 31-6: EXAMPLE OPERATION (ATxRES = 4)



32.0 MATH ACCELERATOR WITH PROPORTIONAL-INTEGRAL-DERIVATIVE (PID) MODULE

The math accelerator module is a mathematics module that can perform a variety of operations, most prominently acting as a PID (Proportional-Integral-Derivative) controller. A PID controller is an algorithm that uses the present error (proportional), the sum of the present and all previous errors (integral), and the difference between the present and previous change (derivative) to correct errors and provide stability in a system. It provides feedback to a system through a series of iterations, using the present error as well as previous errors to calculate a new input to the controller. The data flow for both PID modes is illustrated in Figure 32-1.

The module accomplishes the task of calculating the PID algorithm by utilizing user-provided coefficients along with a multiplier and accumulator. As such, this multiplier and accumulator can also be configured to quickly and efficiently perform signed and unsigned multiply-and-add calculations both with and without accumulation. The data flow for these modes is illustrated in Figure 32-2.

Features of this module include:

- Signed multiplier
- 35-bit signed accumulator
- PID controller support with user inputs for K1, K2, K3, system error and desired set point
- Completion and Error interrupts
- Multiple user modes allowing for PID with or without accumulation as well as several multiplication operations

32.1 PID Module Setup Summary

The PID module can be configured either as a PID controller or as a multiply and accumulate module. Multiply and accumulate can be performed in four modes:

- Unsigned multiply and add, without accumulation
- Unsigned multiply and accumulate
- Signed multiply and add, without accumulation
- Signed multiply and accumulate

All of the modes are selected by the MODE<2:0> bits of the PIDxCON register.

32.1.1 PID MODE SETUP AND OPERATION

When the MODE<2:0> bits of the PIDxCON register are equal to '101', the module is in PID controller mode. The operation of the module in PID controller mode is generally performed as a loop. The input from an external system is fed into the controller, and the controller's output is fed back into the external system. This will produce a new response from the system that is then looped back into the PID controller. The data flow for the PID operation is illustrated in Figure 32-1.

TABLE 34-3: ENHANCED MID-RANGE INSTRUCTION SET

Mnemonic, Operands		Description	Cycles	14-Bit Opcode				Status Affected	Notes
				MSb		LSb			
BYTE-ORIENTED FILE REGISTER OPERATIONS									
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	1fff	ffff	Z	2
CLRW	—	Clear W	1	00	0001	0000	00xx	Z	
COMF	f, d	Complement f	1	00	1001	dfff	ffff	Z	2
DECF	f, d	Decrement f	1	00	0011	dfff	ffff	Z	2
INCF	f, d	Increment f	1	00	1010	dfff	ffff	Z	2
IORWF	f, d	Inclusive OR W with f	1	00	0100	dfff	ffff	Z	2
MOVF	f, d	Move f	1	00	1000	dfff	ffff	Z	2
MOVWF	f	Move W to f	1	00	0000	1fff	ffff		2
RLF	f, d	Rotate Left f through Carry	1	00	1101	dfff	ffff	C	2
RRF	f, d	Rotate Right f through Carry	1	00	1100	dfff	ffff	C	2
SUBWF	f, d	Subtract W from f	1	00	0010	dfff	ffff	C, DC, Z	2
SUBWFB	f, d	Subtract with Borrow W from f	1	11	1011	dfff	ffff	C, DC, Z	2
SWAPF	f, d	Swap nibbles in f	1	00	1110	dfff	ffff		2
XORWF	f, d	Exclusive OR W with f	1	00	0110	dfff	ffff	Z	2
BYTE ORIENTED SKIP OPERATIONS									
DECFSZ	f, d	Decrement f, Skip if 0	1(2)	00	1011	dfff	ffff		1, 2
INCFSZ	f, d	Increment f, Skip if 0	1(2)	00	1111	dfff	ffff		1, 2
BIT-ORIENTED FILE REGISTER OPERATIONS									
BCF	f, b	Bit Clear f	1	01	00bb	bfff	ffff		2
BSF	f, b	Bit Set f	1	01	01bb	bfff	ffff		2
BIT-ORIENTED SKIP OPERATIONS									
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
LITERAL OPERATIONS									
ADDLW	k	Add literal and W	1	11	1110	kkkk	kkkk	C, DC, Z	
ANDLW	k	AND literal with W	1	11	1001	kkkk	kkkk	Z	
IORLW	k	Inclusive OR literal with W	1	11	1000	kkkk	kkkk	Z	
MOVLB	k	Move literal to BSR	1	00	0000	001k	kkkk		
MOVLW	k	Move literal to PCLATH	1	11	0001	1kkk	kkkk		
MOVLW	k	Move literal to W	1	11	0000	kkkk	kkkk		
SUBLW	k	Subtract W from literal	1	11	1100	kkkk	kkkk	C, DC, Z	
XORLW	k	Exclusive OR literal with W	1	11	1010	kkkk	kkkk	Z	

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 35-6: THERMAL CHARACTERISTICS

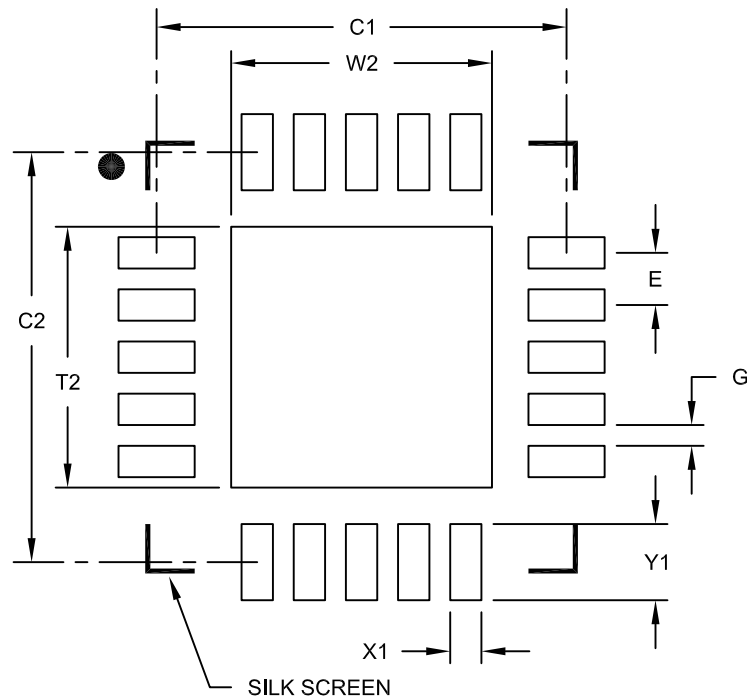
Standard Operating Conditions (unless otherwise stated)					
Param. No.	Sym.	Characteristic	Typ.	Units	Conditions
TH01	θ_{JA}	Thermal Resistance Junction to Ambient	62.2	°C/W	20-pin DIP package
			77.7	°C/W	20-pin SOIC package
			87.3	°C/W	20-pin SSOP package
			43	°C/W	20-pin QFN 4X4mm package
TH02	θ_{JC}	Thermal Resistance Junction to Case	27.5	°C/W	20-pin DIP package
			23.1	°C/W	20-pin SOIC package
			31.1	°C/W	20-pin SSOP package
			5.3	°C/W	20-pin QFN 4X4mm package
TH03	TJMAX	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	PI/O	I/O Power Dissipation	—	W	PI/O = $\sum (I_{OL} * V_{OL}) + \sum (I_{OH} * (V_{DD} - V_{OH}))$
TH07	PDER	Derated Power	—	W	PDER = PDMAX (TJ - TA)/ θ_{JA} ⁽²⁾

Note 1: IDD is current to run the chip alone without driving any load on the output pins.

2: TA = Ambient Temperature; TJ = Junction Temperature

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>



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E	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	X1			0.30
Contact Pad Length	Y1			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