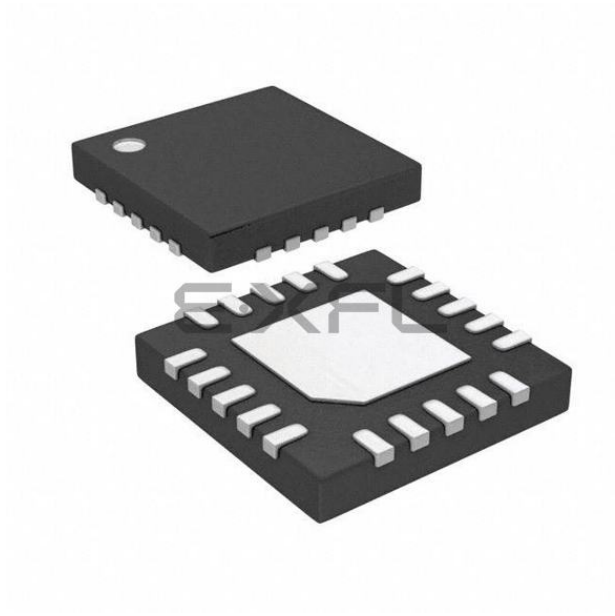


Welcome to E-XFL.COM

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	I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	18
Program Memory Size	14KB (8K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 17x10b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	20-UFQFN Exposed Pad
Supplier Device Package	20-UQFN (4x4)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf18345-e-gz

Pin Allocation Tables

TABLE 1: 14/16-PIN ALLOCATION TABLE (PIC16(L)F18325)

I/O(2)	14-Pin PDIP/SOIC/TSSOP	16-Pin UQFN	ADC	Reference	Comparator	NCO	DAC	DSM	Timers	CCP	PWM	CWG	MSSP	EUSART	CLC	CLKR	Interrupt	Pull-up	Basic
RA0	13	12	ANA0	—	C1IN0+	—	DAC1OUT	—	—	—	—	—	$\overline{SS2}$(1)	—	—	—	IOC	Y	ICDDAT/ ICSPDAT
RA1	12	11	ANA1	VREF+	C1IN0- C2IN0-	—	DAC1REF+	—	—	—	—	—	—	—	—	—	IOC	Y	ICDCLK/ ICSPCLK
RA2	11	10	ANA2	VREF-	—	—	DAC1REF-	—	T0CKI(1)	CCP3(1)	—	CWG1IN(1) CWG2IN(1)	—	—	—	—	INT(1) IOC	Y	—
RA3	4	3	—	—	—	—	—	—	—	—	—	—	—	—	—	—	IOC	Y	$\overline{MCLR}$ VPP
RA4	3	2	ANA4	—	—	—	—	—	T1G(1) SOSCO	—	—	—	—	—	—	—	IOC	Y	CLKOUT OSC2
RA5	2	1	ANA5	—	—	—	—	—	T1CKI(1) SOSCIN SOSCI	—	—	—	—	—	CLCIN3(1)	—	IOC	Y	CLKIN OSC1
RC0	10	9	ANC0	—	C2IN0+	—	—	—	T5CKI(1)	—	—	—	SCK1(1) SCL1(1,3,4)	—	—	—	IOC	Y	—
RC1	9	8	ANC1	—	C1IN1- C2IN1-	—	—	—	—	CCP4(1)	—	—	SDI1(1) SDA1(1,3,4)	—	CLCIN2(1)	—	IOC	Y	—
RC2	8	7	ANC2	—	C1IN2- C2IN2-	—	—	MDCIN1(1)	—	—	—	—	—	—	—	—	IOC	Y	—
RC3	7	6	ANC3	—	C1IN3- C2IN3-	—	—	MDMIN(1)	T5G(1)	CCP2(1)	—	—	$\overline{SS1}$(1)	—	CLCIN0(1)	—	IOC	Y	—
RC4	6	5	ANC4	—	—	—	—	—	T3G(1)	—	—	—	SCK2(1) SCL2(1,3,4)	—	CLCIN1(1)	—	IOC	Y	—
RC5	5	4	ANC5	—	—	—	—	MDCIN2(1)	T3CKI(1)	CCP1(1)	—	—	SDI2(1) SDA2(1,3,4)	RX(1)	—	—	IOC	Y	—
VDD	1	16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VDD
VSS	14	13	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VSS

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.
4: These pins are configured for I^2C logic levels; clock and data signals may be assigned to any of these pins. Assignments to the other pins (e.g., RA5) will operate, but logic levels will be standard TTL/ ST as selected by the INLVL register.

TABLE 4-4: SPECIAL FUNCTION REGISTER SUMMARY BANKS 0-31 (CONTINUED)

Address	Name	PIC16(L)F18325	PIC16(L)F18345	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													
CPU CORE REGISTERS; see Table 4-2 for specifics													
40Ch to 410h	—	—		Unimplemented								—	—
411h	TMR3L			TMR3L<7:0>								xxxx xxxx	uuuu uuuu
412h	TMR3H			TMR3H<7:0>								xxxx xxxx	uuuu uuuu
413h	T3CON			TMR3CS<1:0>		T3CKPS<1:0>		T3SOSC	$\overline{\text{T3SYNC}}$	—	TMR3ON	0000 00-0	uuuu uu-u
414h	T3GCON			TMR3GE	T3GPOL	T3GTM	T3GSPM	T3GGO/$\overline{\text{DONE}}$	T3GVAL	T3GSS<1:0>		0000 0x00	uuuu uxuu
415h	TMR4			TMR4<7:0>								0000 0000	0000 0000
416h	PR4			PR4<7:0>								1111 1111	1111 1111
417h	T4CON			—	T4OUTPS<3:0>				TMR4ON	T4CKPS<1:0>		-000 0000	-000 0000
418h	TMR5L			TMR5L<7:0>								xxxx xxxx	uuuu uuuu
419h	TMR5H			TMR5H<7:0>								xxxx xxxx	uuuu uuuu
41Ah	T5CON			TMR5CS<1:0>		T5CKPS<1:0>		T5SOSC	$\overline{\text{T5SYNC}}$	—	TMR5ON	0000 00-0	uuuu uu-u
41Bh	T5GCON			TMR5GE	T5GPOL	T5GTM	T5GSPM	T5GGO/$\overline{\text{DONE}}$	T5GVAL	T5GSS<1:0>		0000 0x00	uuuu uxuu
41Ch	TMR6			TMR6<7:0>								0000 0000	0000 0000
41Dh	PR6			PR6<7:0>								1111 1111	1111 1111
41Eh	T6CON			—	T6OUTPS<3:0>				TMR6ON	T6CKPS<1:0>		-000 0000	-000 0000
41Fh	—	—		Unimplemented								—	—

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

Note 1: Only on PIC16F18325/18345.

2: Register accessible from both User and ICD Debugger.

TABLE 4-4: SPECIAL FUNCTION REGISTER SUMMARY BANKS 0-31 (CONTINUED)

Address	Name	PIC16(L)F18325	PIC16(L)F18345	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 19-27													
CPU CORE REGISTERS; see Table 4-2 for specifics													
98Ch to 9EFh	—	—		Unimplemented								—	—
A0Ch to A6Fh	—	—		Unimplemented								—	—
A8Ch to AEFh	—	—		Unimplemented								—	—
B0Ch to B6Fh	—	—		Unimplemented								—	—
B8Ch to BEFh	—	—		Unimplemented								—	—
C0Ch to C6Fh	—	—		Unimplemented								—	—
C8Ch to CEFh	—	—		Unimplemented								—	—
D0Ch to D6Fh	—	—		Unimplemented								—	—
D8Ch to DEFh	—	—		Unimplemented								—	—

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

Note 1: Only on PIC16F18325/18345.

2: Register accessible from both User and ICD Debugger.

FIGURE 4-4: ACCESSING THE STACK EXAMPLE 1

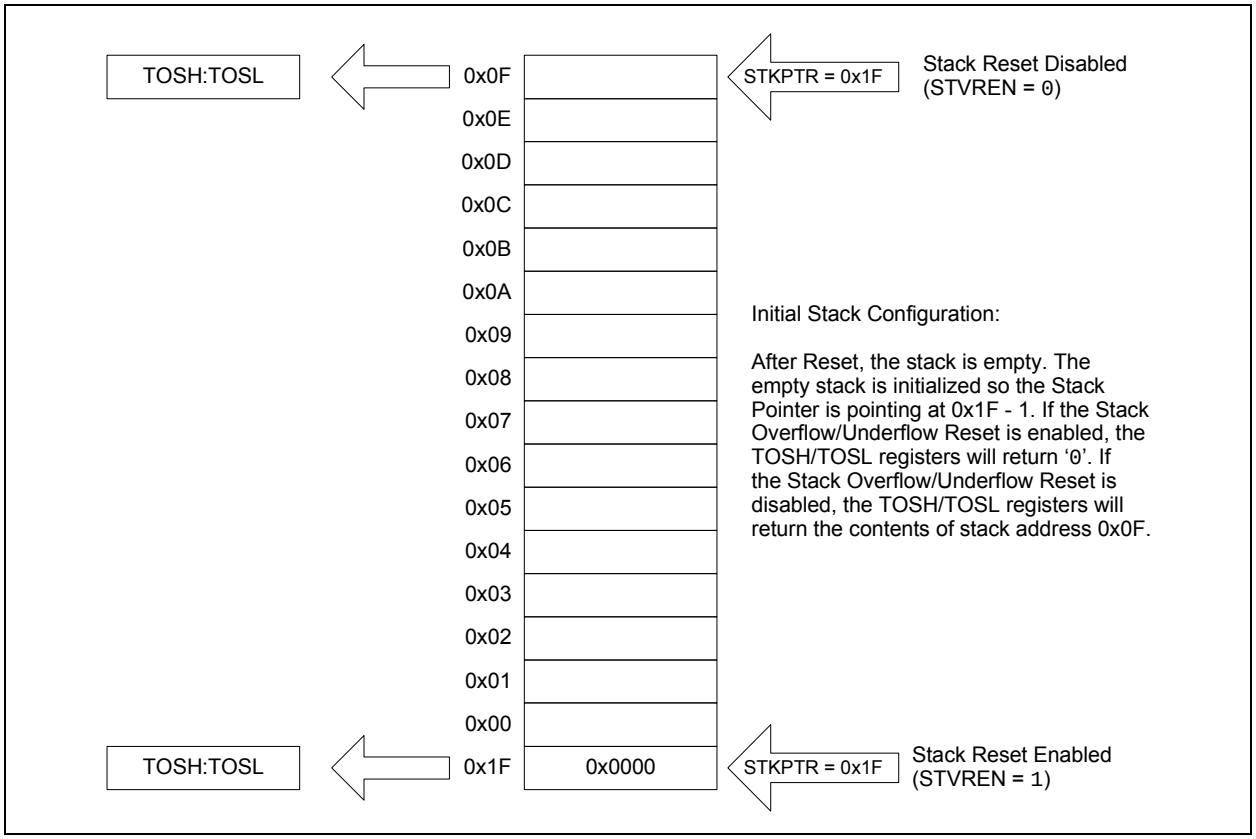
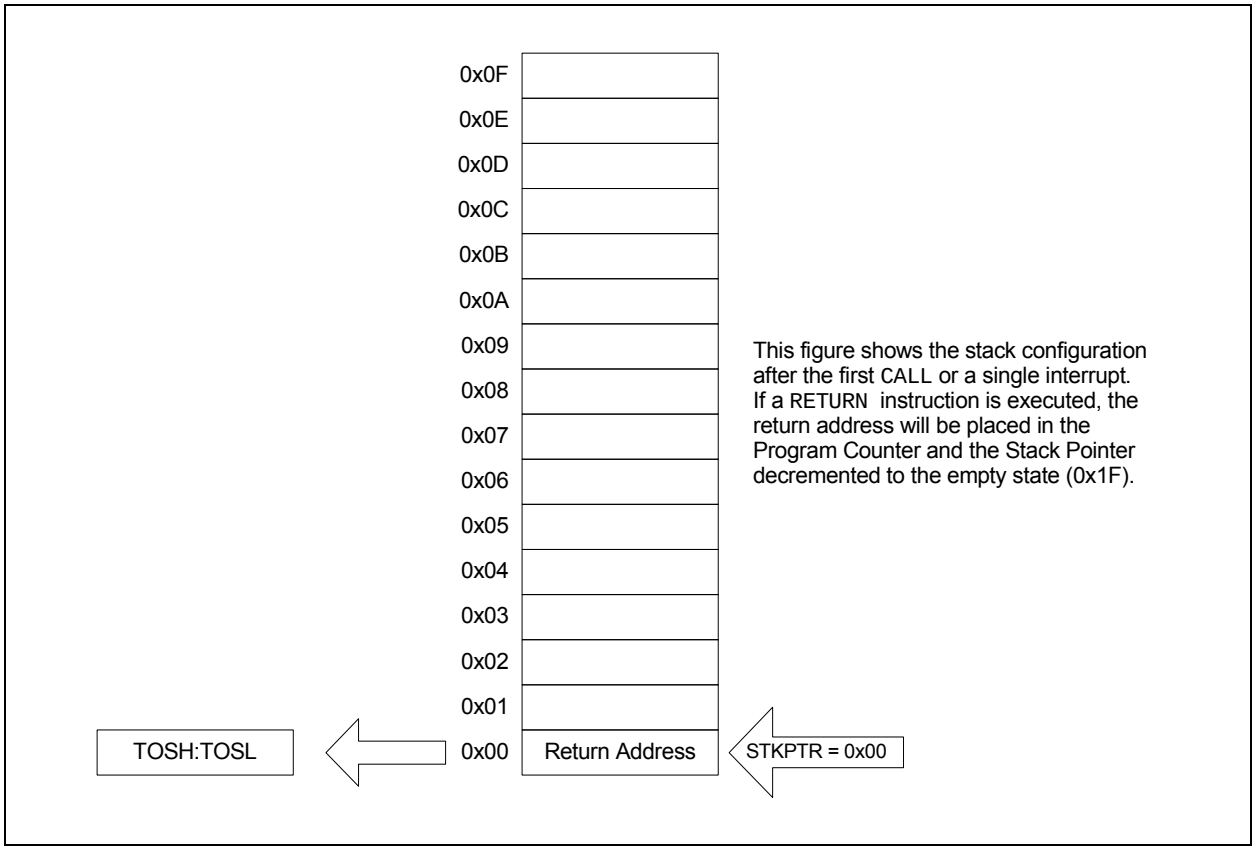


FIGURE 4-5: ACCESSING THE STACK EXAMPLE 2



5.2 Register Definitions: Configuration Words

REGISTER 5-1: CONFIGURATION WORD 1: OSCILLATORS

R/P-1	U-1	R/P-1	U-1	U-1	R/P-1
FCMEN	—	CSWEN	—	—	$\overline{\text{CLKOUTEN}}$
bit 13					bit 8

U-1	R/P-1	R/P-1	R/P-1	U-1	R/P-1	R/P-1	R/P-1
—	RSTOSC2	RSTOSC1	RSTOSC0	—	FEXTOSC2	FEXTOSC1	FEXTOSC0
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

bit 13 **FCMEN:** Fail-Safe Clock Monitor Enable bit
1 = ON FSCM timer enabled
0 = OFF FSCM timer disabled

bit 12 **Unimplemented:** Read as '1'

bit 11 **CSWEN:** Clock Switch Enable bit
1 = ON Writing to NOSC and NDIV is allowed
0 = OFF The NOSC and NDIV bits cannot be changed by user software

bit 10-9 **Unimplemented:** Read as '1'

bit 8 **$\overline{\text{CLKOUTEN}}$:** Clock Out Enable bit
If FEXTOSC = EC, HS, HT or LP, then this bit is ignored; otherwise:
1 = OFF CLKOUT function is disabled; I/O or oscillator function on OSC2
0 = ON CLKOUT function is enabled; FOSC/4 clock appears at OSC2

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

bit 6-4 **RSTOSC<2:0>:** Power-up Default Value for COSC bits
This value is the Reset default value for COSC, and selects the oscillator first used by user software
111 = EXT1X EXTOSC operating per FEXTOSC<2:0> bits
110 = HFINT1 HFINTOSC (1 MHz)
101 = Reserved
100 = LFINT LFINTOSC
011 = SOSC SOSC (32.768 kHz)
010 = Reserved
001 = EXT4X EXTOSC with 4x PLL; EXTOSC operating per FEXTOSC<2:0> bits
000 = HFINT32 HFINTOSC (32 MHz)

bit 3 **Unimplemented:** Read as '1'

bit 2-0 **FEXTOSC<2:0>:** FEXTOSC External Oscillator Mode Selection bits
111 = ECH EC (External Clock) above 8 MHz
110 = ECM EC (External Clock) for 100 kHz to 8 MHz
101 = ECL EC (External Clock) below 100 kHz
100 = OFF Oscillator not enabled
011 = Unimplemented
010 = HS HS (Crystal oscillator) above 8 MHz
001 = XT HT (Crystal oscillator) above 100 kHz, below 8 MHz
000 = LP LP (Crystal oscillator) optimized for 32.768 kHz

REGISTER 5-2: CONFIGURATION WORD 2: SUPERVISORS

R/P-1	R/P-1	R/P-1	U-1	R/P-1	U-1
$\overline{\text{DEBUG}}$	STVREN	PPS1WAY	—	BORV	—
bit 13					bit 8

R/P-1	R/P-1	R/P-1	U-1	R/P-1	R/P-1	R/P-1	R/P-1
BOREN1	BOREN0	$\overline{\text{LPBOREN}}$	—	WDTE1	WDTE0	$\overline{\text{PWRTE}}$	MCLRE
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

bit 13 **$\overline{\text{DEBUG}}$:** Debugger Enable bit(1)
1 = OFF Background debugger disabled; ICSPCLK and ICSPDAT are general purpose I/O pins
0 = ON Background debugger enabled; ICSPCLK and ICSPDAT are dedicated to the debugger

bit 12 **STVREN:** Stack Overflow/Underflow Reset Enable bit
1 = ON Stack Overflow or Underflow will cause a Reset
0 = OFF Stack Overflow or Underflow will not cause a Reset

bit 11 **PPS1WAY:** PPSLOCK One-Way Set Enable bit
1 = ON The PPSLOCK bit can be cleared and set only once; PPS registers remain locked after one clear/set cycle
0 = OFF The PPSLOCK bit can be set and cleared repeatedly (subject to the unlock sequence)

bit 10 **Unimplemented:** Read as '1'

bit 9 **BORV:** Brown-out Reset Voltage Selection bit(2)
1 = LOW Brown-out Reset voltage (V_{BOR}) set to 1.9V on LF, and 2.45V on F devices
0 = HIGH Brown-out Reset voltage (V_{BOR}) set to 2.7V
The higher voltage setting is recommended for operation at or above 16 MHz.

bit 8 **Unimplemented:** Read as '1'

bit 7-6 **BOREN<1:0>:** Brown-out Reset Enable bits
When enabled, Brown-out Reset Voltage (V_{BOR}) is set by the BORV bit
11 = ON Brown-out Reset is enabled; SBOREN bit is ignored
10 = SLEEP Brown-out Reset is enabled while running, disabled in Sleep; SBOREN bit is ignored
01 = SBOREN Brown-out Reset is enabled according to SBOREN
00 = OFF Brown-out Reset is disabled

bit 5 **$\overline{\text{LPBOREN}}$:** Low-Power BOR Enable bit
1 = OFF ULPBOR is disabled
0 = ON ULPBOR is enabled

bit 4 **Unimplemented:** Read as '1'

bit 3-2 **WDTE<1:0>:** Watchdog Timer Enable bit
11 = ON WDT is enabled; SWDTEN is ignored
10 = SLEEP WDT is enabled while running and disabled in Sleep/Idle; SWDTEN is ignored
01 = SWDTEN WDT is controlled by the SWDTEN bit in the WDTCON register
00 = OFF WDT is disabled; SWDTEN is ignored

bit 1 **$\overline{\text{PWRTE}}$:** Power-up Timer Enable bit
1 = OFF PWRT is disabled
0 = ON PWRT is enabled

bit 0 **MCLRE:** Master Clear ($\overline{\text{MCLR}}$) Enable bit
If LVP = 1:
RA3 pin function is $\overline{\text{MCLR}}$.
If LVP = 0:
1 = ON $\overline{\text{MCLR}}$ pin is $\overline{\text{MCLR}}$.
0 = OFF $\overline{\text{MCLR}}$ pin function is port-defined function.

Note 1: The DEBUG bit in Configuration Words is managed automatically by device development tools including debuggers and programmers. For normal device operation, this bit should be maintained as a '1'.
2: See V_{BOR} parameter for specific trip point voltages.

REGISTER 7-5: OSCEN: OSCILLATOR MANUAL ENABLE REGISTER

R/W-0/0	R/W-0/0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	U-0
EXTOEN	HFOEN	—	LFOEN	SOSCEN	ADOEN	—	—
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 **EXTOEN:** External Oscillator Manual Request Enable bit
1 = EXTOSC is explicitly enabled, operating as specified by FEXTOSC
0 = EXTOSC could be enabled by another module

bit 6 **HFOEN:** HFINTOSC Oscillator Manual Request Enable bit
1 = HFINTOSC is explicitly enabled, operating as specified by OSCFRQ (Register 7-6)
0 = HFINTOSC could be enabled by another module

bit 5 **Unimplemented:** Read as ‘0’

bit 4 **LFOEN:** LFINTOSC (31 kHz) Oscillator Manual Request Enable bit
1 = LFINTOSC is explicitly enabled
0 = LFINTOSC could be enabled by another module

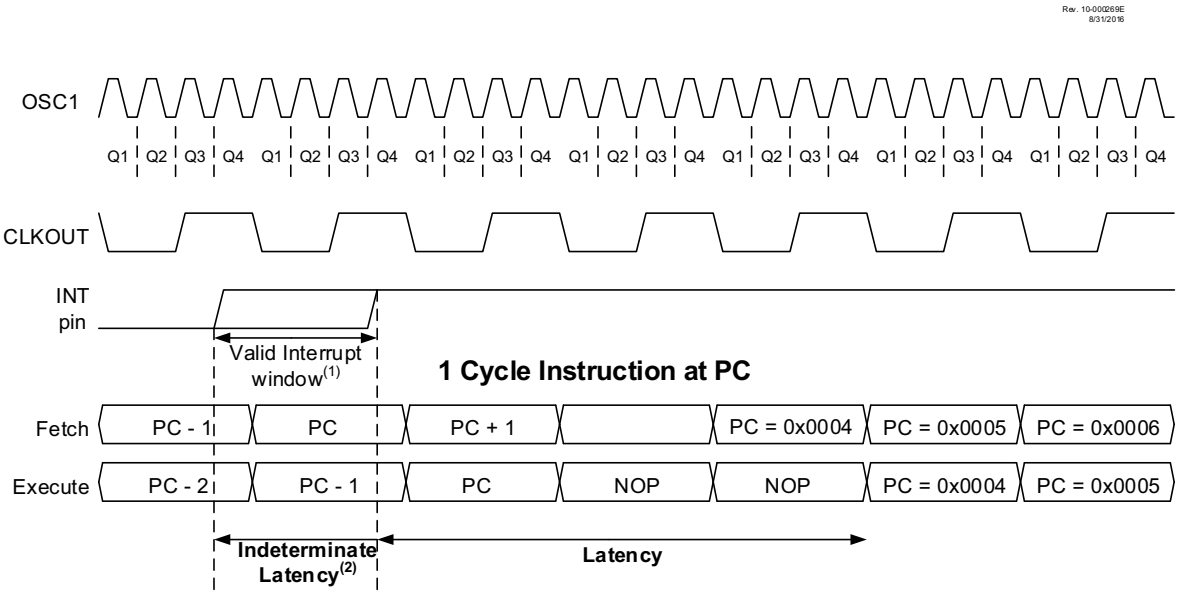
bit 3 **SOSCEN:** Secondary Oscillator Manual Request Enable bit
1 = Secondary Oscillator is explicitly enabled
0 = Secondary Oscillator could be enabled by another module

bit 2 **ADOEN:** ADOSC (600 kHz) Oscillator Manual Request Enable bit
1 = ADOSC is explicitly enabled
0 = ADOSC could be enabled by another module

bit 1 **Unimplemented:** Read as ‘0’

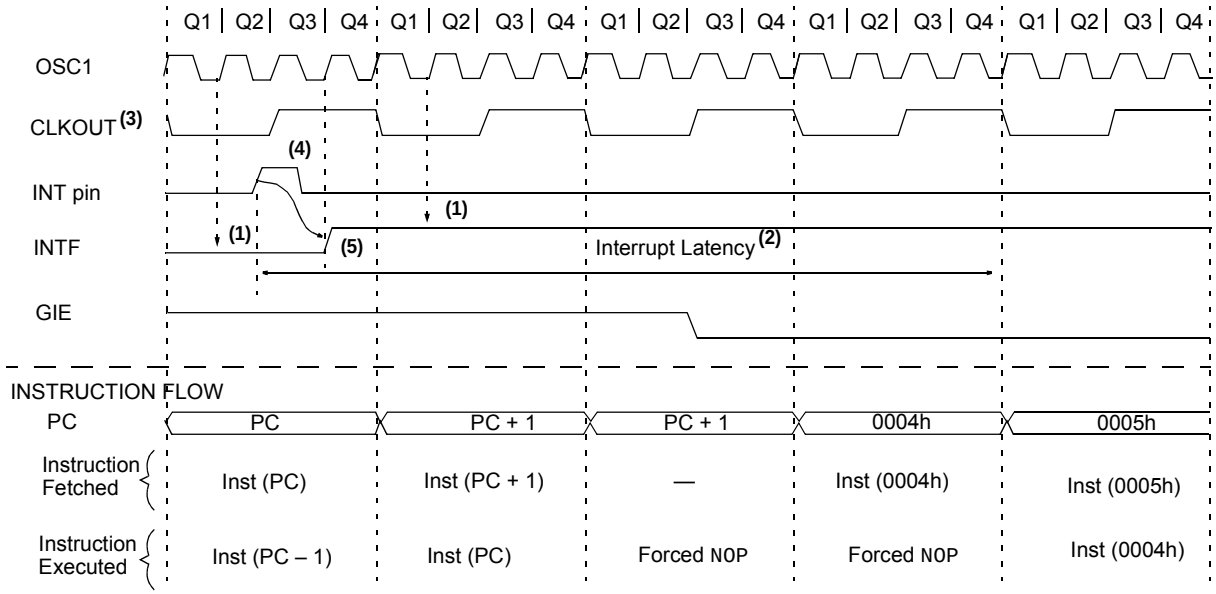
bit 0 **Unimplemented:** Read as ‘0’

FIGURE 8-2: INTERRUPT LATENCY



Note 1: An interrupt may occur at any time during the interrupt window.
2: Since an interrupt may occur any time during the interrupt window, the actual latency can vary.

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 35.0 "Electrical Specifications"**".

5: INTF is enabled to be set any time during the Q4-Q1 cycles.

11.4.2 NVM UNLOCK SEQUENCE

The unlock sequence is a mechanism that protects the NVM from unintended self-write programming or erasing. The sequence must be executed and completed without interruption to successfully complete any of the following operations:

- Program Flash Memory Row Erase
- Load of Program Flash Memory write latches
- Write of Program Flash Memory write latches to Program Flash Memory memory
- Write of Program Flash Memory write latches to user IDs
- Write to EEPROM

The unlock sequence consists of the following steps and must be completed in order:

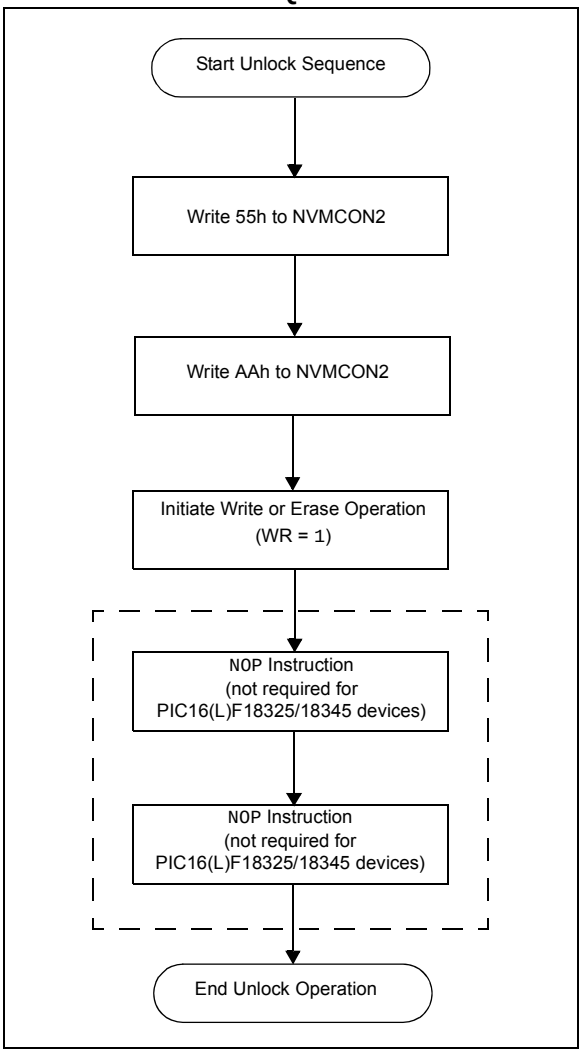
- Write 55h to NVMCON2
- Write AAh to NMVCON2
- Set the WR bit of NVMCON1

Once the WR bit is set, the processor will stall internal operations until the operation is complete and then resume with the next instruction.

Note:	The two NOP instructions after setting the WR bit, which were required in previous devices, are not required for PIC16(L)F18325/18345 devices. See Figure 11-2.

Since the unlock sequence must not be interrupted, global interrupts should be disabled prior to the unlock sequence and re-enabled after the unlock sequence is completed.

FIGURE 11-2: NVM UNLOCK SEQUENCE FLOWCHART



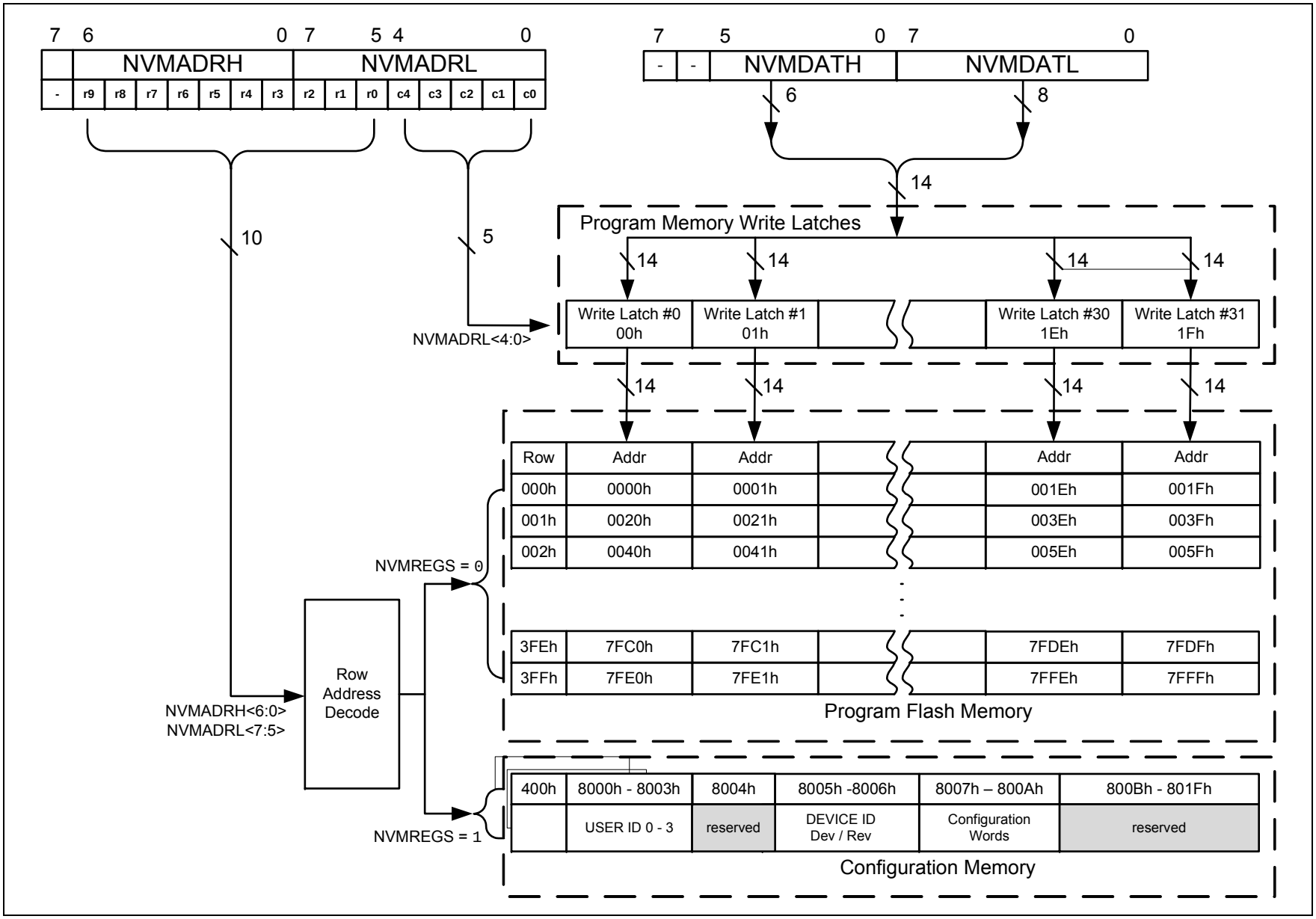
EXAMPLE 11-2: NVM UNLOCK SEQUENCE

```
BANKSEL     NVMCON1
BSF         NVMCON1,WREN    ; Enable write/erase
MOVLW       55h             ; Load 55h
BCF         INTCON,GIE      ; Recommended so sequence is not interrupted

MOVWF       NVMCON2         ; Step 1: Load 55h into NVMCON2
MOVLW       AAh             ; Step 2: Load W with AAh
MOVWF       NVMCON2         ; Step 3: Load AAh into NVMCON2
BSF         NVMCON1,WR      ; Step 4: Set WR bit to begin write/erase

BSF         INTCON,GIE      ; Re-enable interrupts
```

Note 1: Sequence begins when NVMCON2 is written; steps 1-4 must occur in the cycle-accurate order shown.

2: Opcodes shown are illustrative; any instruction that has the indicated effect may be used.

FIGURE 11-4: BLOCK WRITES TO PROGRAM FLASH MEMORY WITH 32 WRITE LATCHES



REGISTER 12-18: 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 7-6 **TRISC<7:6>**: PORTC Tri-State Control bits(1)
1 = PORTC pin configured as an input (tri-stated)
0 = PORTC pin configured as an output

bit 5-0 **TRISC<5:0>:** PORTC Tri-State Control bits
1 = PORTC pin configured as an input (tri-stated)
0 = PORTC pin configured as an output

Note 1: PIC16(L)F18345 only; otherwise read as '0'.

REGISTER 12-19: 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(1)	LATC6(1)	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 7-6 **LATC<7:6>**: PORTC Output Latch Value bits(1)

bit 5-0 **LATC<5:0>**: PORTC Output Latch Value bits

Note 1: PIC16(L)F18345 only; otherwise read as '0'.

REGISTER 12-22: ODCONC: PORTC OPEN-DRAIN 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
ODCC7(1)	ODCC6(1)	ODCC5	ODCC4	ODCC3	ODCC2	ODCC1	ODCC0
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 **ODCC<7:6>**: PORTC Open-Drain Enable bits(1)
For RC<7:6> pins, respectively
1 = Port pin operates as open-drain drive (sink current only)
0 = Port pin operates as standard push-pull drive (source and sink current)

bit 5-0 **ODCC<5:0>:** PORTC Open-Drain Enable bits
For RC<5:0> pins, respectively
1 = Port pin operates as open-drain drive (sink current only)
0 = Port pin operates as standard push-pull drive (source and sink current)

Note 1: PIC16(L)F18345 only; otherwise read as '0'.

REGISTER 12-23: SLRCONC: PORTC SLEW RATE CONTROL 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
SLRC7(1)	SLRC6(1)	SLRC5	SLRC4	SLRC3	SLRC2	SLRC1	SLRC0
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 **SLRC<7:6>:** PORTC Slew Rate Enable bits(1)
For RC<7:6> pins, respectively
1 = Port pin slew rate is limited
0 = Port pin slews at maximum rate

bit 5-0 **SLRC<5:0>:** PORTC Slew Rate Enable bits
For RC<5:0> pins, respectively
1 = Port pin slew rate is limited
0 = Port pin slews at maximum rate

Note 1: PIC16(L)F18345 only; otherwise read as '0'.

18.2 Comparator Control

Each comparator has two control registers: CMxCON0 and CMxCON1.

The CMxCON0 register (see Register 18-1) contains Control and Status bits for the following:

- Enable
- Output
- Output polarity
- Hysteresis enable
- Timer1 output synchronization

The CMxCON1 register (see Register 18-2) contains Control bits for the following:

- Interrupt on positive/negative edge enables
- Positive input channel selection
- Negative input channel selection

18.2.1 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.

18.2.2 COMPARATOR OUTPUT

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.

The comparator output can also be routed to an external pin through the RxyPPS register (Register 13-2). The corresponding TRIS bit must be clear to enable the pin as an output.

Note 1:	The internal output of the comparator is latched with each instruction cycle. Unless otherwise specified, external outputs are not latched.

18.2.3 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 18-2 shows the output state versus input conditions, including polarity control.

TABLE 18-2: 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

18.3 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 Comparator Specifications in Table 35-14 for more information.

18.4 Timer1 Gate Operation

The output resulting from a comparator operation can be used as a source for gate control of Timer1. See **Section 27.5 "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.

18.4.1 COMPARATOR OUTPUT SYNCHRONIZATION

The output from a comparator 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. This allows the timer/counter to synchronize with the CxOUT bit so that the software sees no ambiguity due to timing. See the Comparator Block Diagram (Figure 18-2) and the Timer1 Block Diagram (Figure 27-1) for more information.

20.11 Register Definitions: CWG Control

REGISTER 20-1: CWGxCON0: CWGx CONTROL REGISTER 0

R/W-0/0	R/W/HC-0/0	U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0
EN	LD(1)	—	—	—	MODE<2:0>		
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	HS/HC = Bit is set/cleared by hardware

bit 7 **EN:** CWGx Enable bit
- 1 = CWGx is enabled
- 0 = CWGx is disabled

bit 6 **LD:** CWG Load Buffers bit(1)
- 1 = Dead-band count buffers to be loaded on CWG data rising edge following first falling edge after this bit is set.
- 0 = Buffers remain unchanged

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

bit 2-0 **MODE<2:0>**: CWGx Mode bits
- 111 = Reserved
- 110 = Reserved
- 101 = CWG outputs operate in Push-Pull mode
- 100 = CWG outputs operate in Half-Bridge mode
- 011 = CWG outputs operate in Reverse Full-Bridge mode
- 010 = CWG outputs operate in Forward Full-Bridge mode
- 001 = CWG outputs operate in Synchronous Steering mode
- 000 = CWG outputs operate in Asynchronous Steering mode

Note 1: This bit can only be set after EN = 1; it cannot be set in the same cycle when EN is set.

REGISTER 20-2: CWGxCON1: CWGx CONTROL REGISTER 1

U-0	U-0	R-x	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	IN	—	POLD	POLC	POLB	POLA
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 7-6 **Unimplemented:** Read as ‘0’

bit 5 **IN:** CWGx Data Input Signal (read-only)

bit 4 **Unimplemented:** Read as ‘0’

bit 3 **POLD:** WGxD Output Polarity bit
1 = Signal output is inverted polarity
0 = Signal output is normal polarity

bit 2 **POLC:** WGxC Output Polarity bit
1 = Signal output is inverted polarity
0 = Signal output is normal polarity

bit 1 **POLB:** WGxB Output Polarity bit
1 = Signal output is inverted polarity
0 = Signal output is normal polarity

bit 0 **POLA:** WGxA Output Polarity bit
1 = Signal output is inverted polarity
0 = Signal output is normal polarity

REGISTER 20-3: CWGxCLKCON: CWGx CLOCK INPUT SELECTION REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0/0
—	—	—	—	—	—	—	CS
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 7-1 **Unimplemented:** Read as ‘0’

bit 0 **CS:** CWG Clock Source Selection Select bits

CS	Clock Source
0	FOSC
1	HFINTOSC (remains operating during Sleep)

27.1 Timer1 Operation

The Timer1 module is a 16-bit incrementing timer which is accessed through the TMR1H:TMR1L register pair. Writes to TMR1H or TMR1L directly update the timer.

The module can be used with either internal or external clock sources, and has the Timer1 Gate Enable function. When Timer1 is used with the Timer1 Gate Enable, the timer can measure time intervals or count signal pulses between two points of interest. When used without the Timer1 Gate Enable, the timer simply measures time intervals.

Timer1 is enabled by configuring the TMR1ON and TMR1GE bits in the T1CON and T1GCON registers, respectively. Table 27-1 displays the Timer1 enable selections.

TABLE 27-1: TIMER1 ENABLE SELECTIONS

TMR1ON	TMR1GE	Timer1 Operation
0	0	Off
0	1	Off
1	0	Always On
1	1	Count Enabled

27.2 Clock Source Selection

The TMR1CS<1:0> and T1OSC bits of the T1CON register are used to select the clock source for Timer1. Table 27-2 displays the clock source selections. The TMR1H:TMR1L register pair will increment on multiples of the clock source as determined by the Timer1 prescaler.

When either the FOSC or LFINTOSC clock source is selected, the TMR1H:TMR1L register pair will increment every rising clock edge. Reading from the TMR1H:TMR1L register pair when either the FOSC or LFINTOSC is the clock source will cause a 2 LSb loss in resolution, which can be mitigated by using an asynchronous input signal to gate the Timer1 clock input (see **Section 26.5 “Operation During Sleep”** for more information on the Timer1 Gate Enable).

When the FOSC/4 clock source is selected, the TMR1H:TMR1L register pair increments every instruction cycle (once every four FOSC pulses).

In addition to the internal clock sources, Timer1 has a dedicated external clock input pin, T1CKI. T1CKI can be either synchronized to the system clock or can run asynchronously via the T1SYNC bit of the T1CON register. When the T1CKI pin is used as the clock source, the TMR1H:TMR1L register pair increments on the rising edge of the T1CKI clock input.

Note: When using Timer1 to count events, 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 27-2: CLOCK SOURCE SELECTIONS

TMR1CS<1:0>	Clock Source
11	LFINTOSC
10	External Clocking on T1CKI Pin
01	System Clock (FOSC)
00	Instruction Clock (FOSC/4)

30.5.6 CLOCK STRETCHING

Clock stretching occurs when a device on the bus holds the SCL 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 SCL.

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 SCL line to go low and then hold it. Setting CKP will release SCL and allow more communication.

30.5.6.1 Normal Clock Stretching

Following an $\overline{ACK}$ if the R/$\overline{W}$ 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 $\overline{ACK}$ sequence. Once the slave is ready; CKP is set by software and communication resumes.

30.5.6.2 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 SCL is stretched without CKP being cleared. SCL 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.

30.5.6.3 Byte NACKing

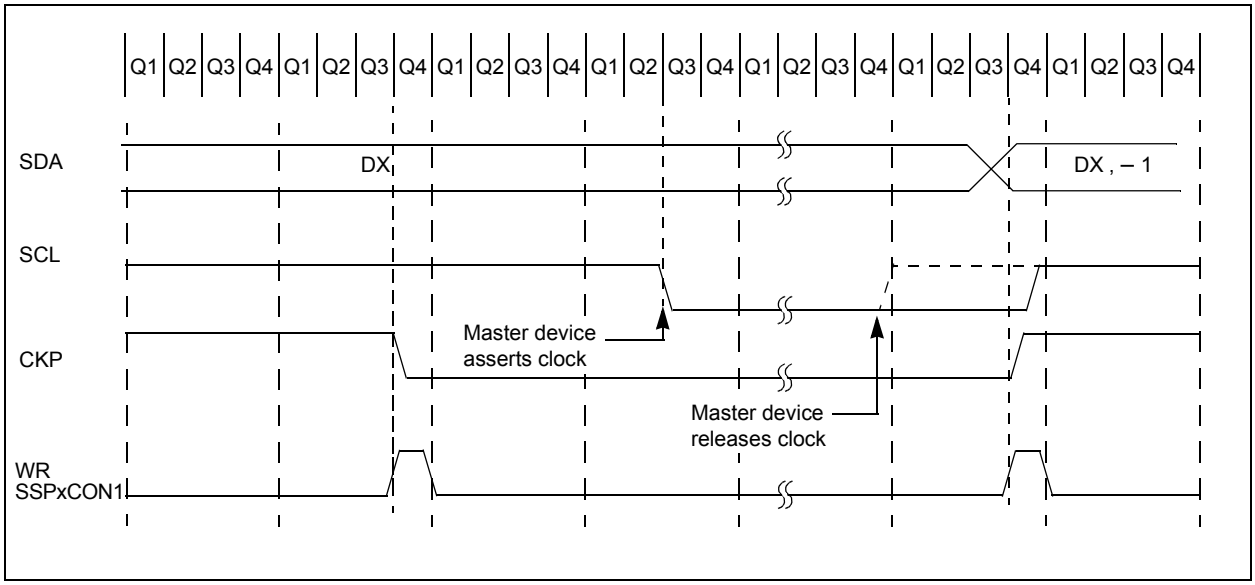
When AHEN bit of SSPxCON3 is set; CKP is cleared by hardware after the eighth falling edge of SCL for a received matching address byte. When DHEN bit of SSPxCON3 is set; CKP is cleared after the eighth falling edge of SCL for received data.

Stretching after the eighth falling edge of SCL allows the slave to look at the received address or data and decide if it wants to ACK the received data.

30.5.7 CLOCK SYNCHRONIZATION AND THE CKP BIT

Any time the CKP bit is cleared, the module will wait for the SCL line to go low and then hold it. However, clearing the CKP bit will not assert the SCL output low until the SCL output is already sampled low. Therefore, the CKP bit will not assert the SCL line until an external I^2C master device has already asserted the SCL line. The SCL output will remain low until the CKP bit is set and all other devices on the I^2C bus have released SCL. This ensures that a write to the CKP bit will not violate the minimum high time requirement for SCL (see Figure 30-23).

FIGURE 30-23: CLOCK SYNCHRONIZATION TIMING



30.7 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 30-6). 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.

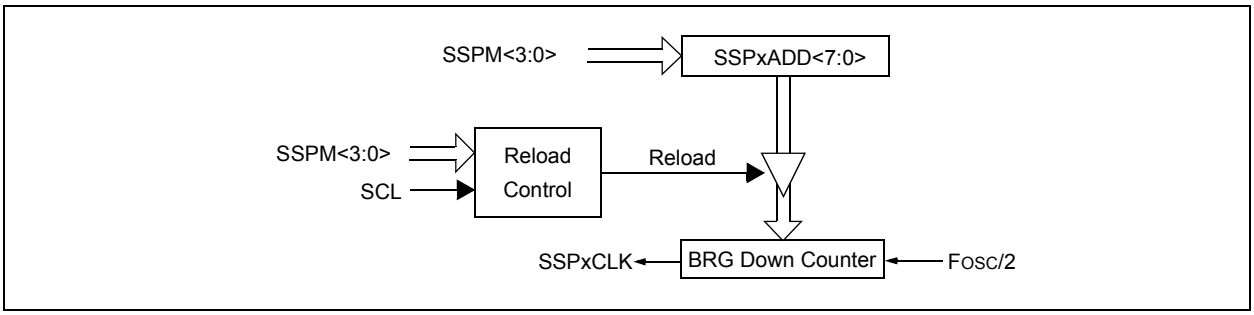
An internal signal “Reload” in Figure 30-40 triggers the value from SSPxADD to be loaded into the BRG counter. This occurs twice for each oscillation of the module clock line. The logic dictating when the reload signal is asserted depends on the mode the MSSPx is being operated in.

Table 30-4 demonstrates clock rates based on instruction cycles and the BRG value loaded into SSPADD.

EQUATION 30-1:

$$F_{CLOCK} = \frac{F_{OSC}}{(SSPxADD + 1)(4)}$$

FIGURE 30-40: 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 I^2C. This is an implementation limitation.

REGISTER 30-3: SSPxCON2: SSPx CONTROL REGISTER 2 (I^2C MODE ONLY)(1)

R/W-0/0	R/HS/HC-0	R/W-0/0	R/S/HC-0/0	R/S/HC-0/0	R/S/HC-0/0	R/S/HC-0/0	R/S/HC-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 I^2C Slave mode only)
1 = Enable interrupt when a general call address (0x00 or 00h) is received in the SSPSR
0 = General call address disabled

bit 6 **ACKSTAT:** Acknowledge Status bit (in I^2C mode only)
1 = Acknowledge was not received
0 = Acknowledge was received

bit 5 **ACKDT:** Acknowledge Data bit (in I^2C 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 I^2C Master mode only)
In Master Receive mode:
1 = Initiate Acknowledge sequence on SDA and SCL pins, and transmit ACKDT data bit. Automatically cleared by hardware.
0 = Acknowledge sequence idle

bit 3 **RCEN:** Receive Enable bit (in I^2C Master mode only)
1 = Enables Receive mode for I^2C
0 = Receive idle

bit 2 **PEN:** Stop Condition Enable bit (in I^2C Master mode only)
1 = Initiate Stop condition on SDA and SCL pins. Automatically cleared by hardware.
0 = Stop condition Idle

bit 1 **RSEN:** Repeated Start Condition Enable bit (in I^2C Master mode only)
1 = Initiate Repeated Start condition on SDA and SCL 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 SDA and SCL 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 I^2C module is not in the idle state, these bits may not be set (no spooling) and the SSPBUF may not be written.

TABLE 34-3: PIC16(L)F18325/18345 INSTRUCTION SET (CONTINUED)

Mnemonic, Operands		Description	Cycles	14-bit Opcode MSb	LSb	Status Affected	Notes
INHERENT OPERATIONS							
CLRWDT	–	Clear Watchdog Timer	1	00 0000 0110 0100		$\overline{TO}$, $\overline{PD}$	
NOP	–	No Operation	1	00 0000 0000 0000			
RESET	–	Software device Reset	1	00 0000 0000 0001			
SLEEP	–	Go into Standby mode	1	00 0000 0110 0011		$\overline{TO}$, $\overline{PD}$	
TRIS	f	Load TRIS register with W	1	00 0000 0110 0fff			
C-COMPILER OPTIMIZED							
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 modifier, mm	1	00 0000 0001 0nmm		Z	**2, 3**
	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 modifier, mm	1	00 0000 0001 1nmm			**2, 3**
	k[n]	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 **Section 34.2 "Instruction Descriptions"** for detailed MOVIW and MOVWI instruction descriptions.