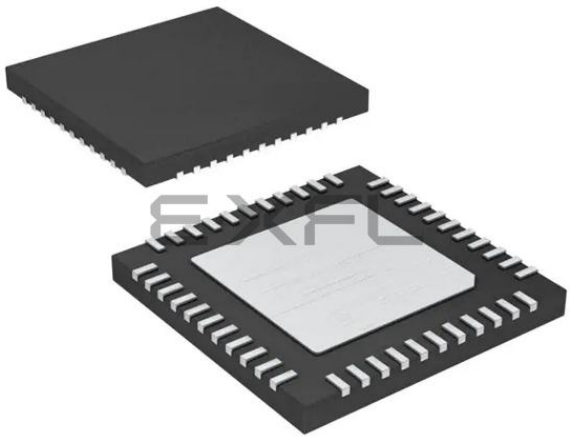


Welcome to E-XFL.COM

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	64MHz
Connectivity	I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, HLVD, POR, PWM, WDT
Number of I/O	36
Program Memory Size	64KB (32K x 16)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 35x12b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-VQFN Exposed Pad
Supplier Device Package	44-QFN (8x8)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18lf46k42t-i-ml

2.5 External Oscillator Pins

Many microcontrollers have options for at least two oscillators: a high-frequency primary oscillator and a low-frequency secondary oscillator (refer to **Section 7.0 "Oscillator Module (with Fail-Safe Clock Monitor)"** for details).

The oscillator circuit should be placed on the same side of the board as the device. Place the oscillator circuit close to the respective oscillator pins with no more than 0.5 inch (12 mm) between the circuit components and the pins. The load capacitors should be placed next to the oscillator itself, on the same side of the board.

Use a grounded copper pour around the oscillator circuit to isolate it from surrounding circuits. The grounded copper pour should be routed directly to the MCU ground. Do not run any signal traces or power traces inside the ground pour. Also, if using a two-sided board, avoid any traces on the other side of the board where the crystal is placed.

Layout suggestions are shown in Figure 2-3. In-line packages may be handled with a single-sided layout that completely encompasses the oscillator pins. With fine-pitch packages, it is not always possible to completely surround the pins and components. A suitable solution is to tie the broken guard sections to a mirrored ground layer. In all cases, the guard trace(s) must be returned to ground.

In planning the application's routing and I/O assignments, ensure that adjacent port pins, and other signals in close proximity to the oscillator, are benign (i.e., free of high frequencies, short rise and fall times, and other similar noise).

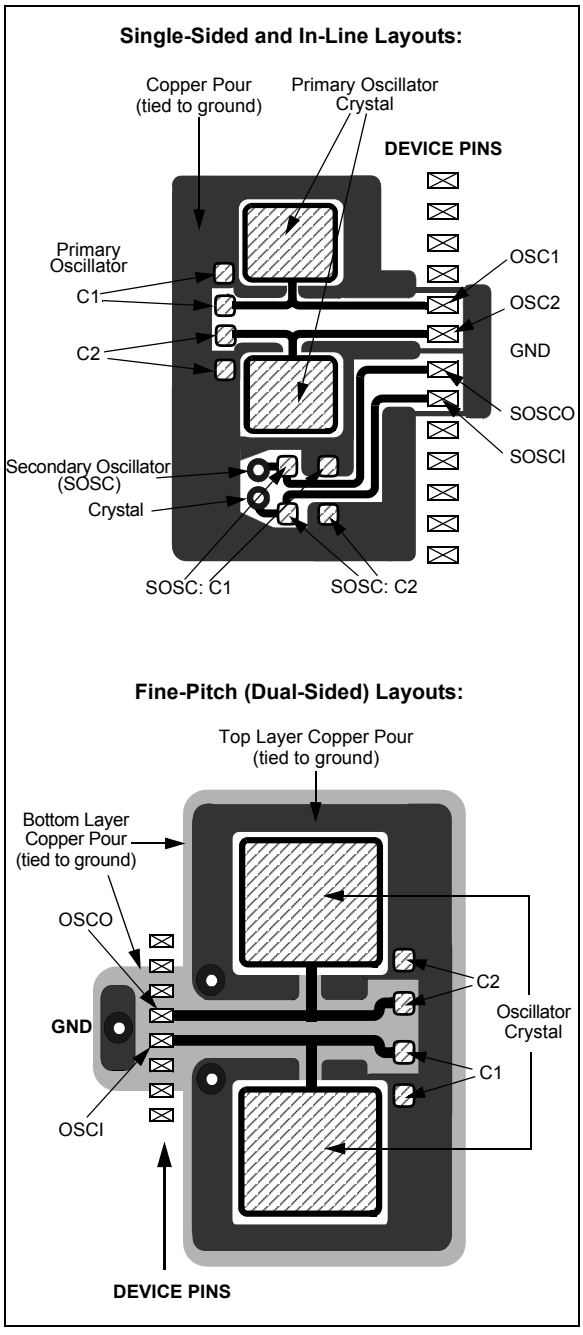
For additional information and design guidance on oscillator circuits, refer to these Microchip application notes, available at the corporate website (www.microchip.com):

- AN826, "*Crystal Oscillator Basics and Crystal Selection for rfPIC™ and PICmicro® Devices*"
- AN849, *"Basic PICmicro® Oscillator Design"*
- AN943, *"Practical PICmicro® Oscillator Analysis and Design"*
- AN949, *"Making Your Oscillator Work"*

2.6 Unused I/Os

Unused I/O pins should be configured as outputs and driven to a logic low state. Alternatively, connect a 1 kΩ to 10 kΩ resistor to Vss on unused pins and drive the output to logic low.

FIGURE 2-3: SUGGESTED PLACEMENT OF THE OSCILLATOR CIRCUIT



9.5 Context Saving

The Interrupt controller supports a two-level deep context saving (Main routine context and Low ISR context). Refer to state machine shown in Figure 9-6 for details.

The Program Counter (PC) is saved on the dedicated device PC stack. CPU registers saved include STATUS, WREG, BSR, FSR0/1/2, PRODL/H and PCLATH/U.

After WREG has been saved to the context registers, the resolved vector number of the interrupt source to be serviced is copied into WREG. Context save and restore operation is completed by the interrupt controller based on current state of the interrupts and the order in which they were sent to the CPU.

Context save/restore works the same way in both states of MVECEN. When IPEN = 0, there is only one level interrupt active. Hence, only the main context is saved when an interrupt is received.

9.5.1 ACCESSING SHADOW REGISTERS

The Interrupt controller automatically saves the context information in the shadow registers available in Bank 56. Both the saved context values (i.e., main routine and low ISR) can be accessed using the same set of shadow registers. By clearing the SHADLO bit in the SHADCON register (Register 9-43), the CPU register values saved for main routine context can accessed, and by setting the SHADLO bit of the CPU register, values saved for low ISR context can accessed. Low ISR context is automatically restored to the CPU registers upon exiting the high ISR. Similarly, the main context is automatically restored to the CPU registers upon exiting the low ISR.

The Shadow registers in Bank 56 are readable and writable, so if the user desires to modify the context, then the corresponding shadow register should be modified and the value will be restored when exiting the ISR. Depending on the user's application, other registers may also need to be saved.

REGISTER 16-4: ANSELx: ANALOG SELECT 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
ANSELx7	ANSELx6	ANSELx5	ANSELx4	ANSELx3	ANSELx2	ANSELx1	ANSELx0
bit 7							bit 0

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

bit 7-0 **ANSELx<7:0>**: Analog Select on Pins Rx<7:0>

1 = Digital Input buffers are disabled.
0 = ST and TTL input devices are enabled

TABLE 16-5: ANALOG SELECT PORT REGISTERS

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ANSELA	ANSELA7	ANSELA6	ANSELA5	ANSELA4	ANSELA3	ANSELA2	ANSELA1	ANSELA0
ANSELB	ANSELB7	ANSELB6	ANSELB5	ANSELB4	ANSELB3	ANSELB2	ANSELB1	ANSELB0
ANSELC	ANSELC7	ANSELC6	ANSELC5	ANSELC4	ANSELC3	ANSELC2	ANSELC1	ANSELC0
ANSELD(1)	ANSELD7	ANSELD6	ANSELD5	ANSELD4	ANSELD3	ANSELD2	ANSELD1	ANSELD0
ANSELE(1)	—	—	—	—	—	ANSELE2	ANSELE1	ANSELE0
ANSELF(2)	ANSELF7	ANSELF6	ANSELF5	ANSELF4	ANSELF3	ANSELF2	ANSELF1	ANSELF0

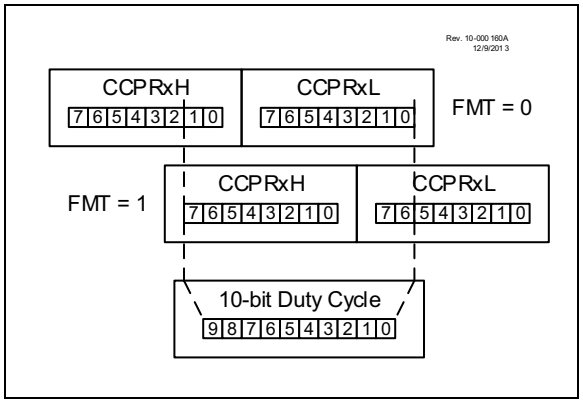
Note 1: Unimplemented in PIC18(L)F26/27K42.
2: Unimplemented in PIC18(L)F26/45/46/47K42.

23.4.5 PWM DUTY CYCLE

The PWM duty cycle is specified by writing a 10-bit value to the CCPRxH:CCPRxL register pair. The alignment of the 10-bit value is determined by the FMT bit of the CCPxCON register (see Figure 23-5). The CCPRxH:CCPRxL register pair can be written to at any time; however the duty cycle value is not latched into the 10-bit buffer until after a match between T2PR and T2TMR.

Equation 23-2 is used to calculate the PWM pulse width. Equation 23-3 is used to calculate the PWM duty cycle ratio.

FIGURE 23-5: PWM 10-BIT ALIGNMENT



EQUATION 23-2: PULSE WIDTH

$$Pulse\ Width = (CCPRxH{:}CCPRxL\ register\ pair) \bullet T_{OSC} \bullet (TMR2\ Prescale\ Value)$$

EQUATION 23-3: DUTY CYCLE RATIO

$$Duty\ Cycle\ Ratio = \frac{(CCPRxH{:}CCPRxL\ register\ pair)}{4(T2PR + 1)}$$

CCPRxH:CCPRxL register pair are used to double buffer the PWM duty cycle. This double buffering provides glitchless PWM operation.

The 8-bit timer T2TMR 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 prescaler is set to 1:1.

When the 10-bit time base matches the CCPRxH:CCPRxL register pair, then the CCPx pin is cleared (see Figure 23-4).

23.4.6 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 T2PR is 255. The resolution is a function of the T2PR register value as shown by Equation 23-4.

EQUATION 23-4: PWM RESOLUTION

$$Resolution = \frac{\log[4(T2PR + 1)]}{\log(2)}\ bits$$

Note: If the pulse-width value is greater than the period, the assigned PWM pin(s) will remain unchanged.

FIGURE 25-3: TIMER MODE TIMING DIAGRAM

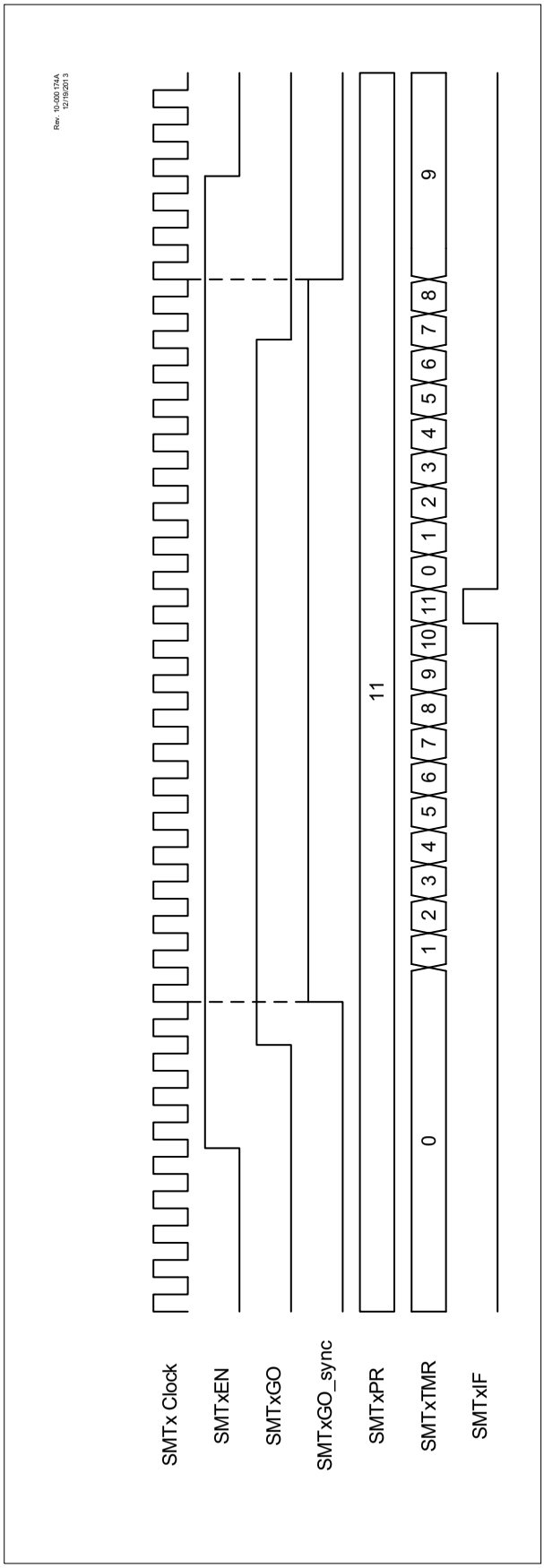
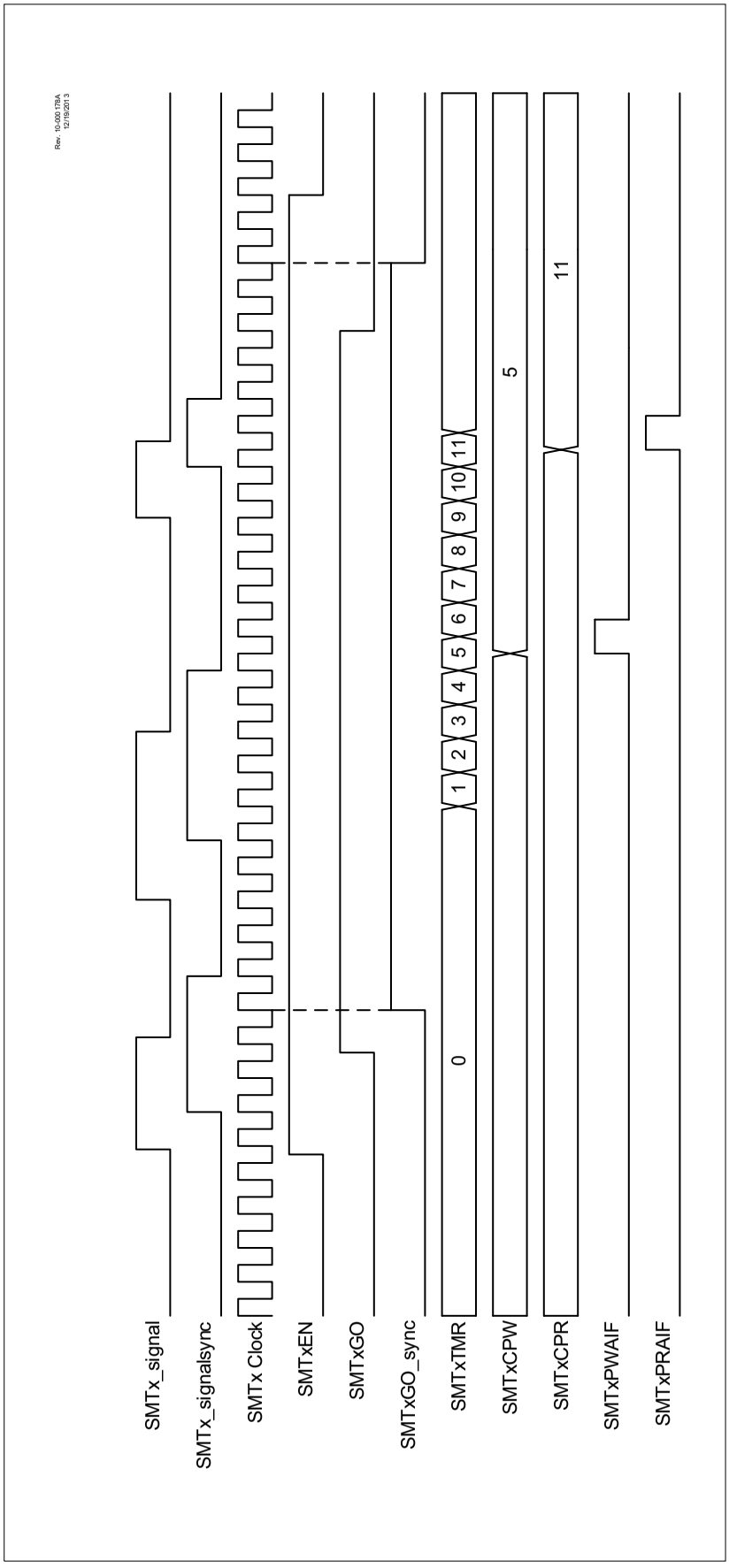


FIGURE 25-7: PERIOD AND DUTY-CYCLE SINGLE ACQUISITION TIMING DIAGRAM



28.2 FIXED DUTY CYCLE MODE

In Fixed Duty Cycle (FDC) mode, every time the accumulator overflows (NCO_overflow), the output is toggled. This provides a 50% duty cycle, provided that the increment value remains constant. For more information, see Figure 28-2.

28.3 PULSE FREQUENCY MODE

In Pulse Frequency (PF) mode, every time the Accumulator overflows, the output becomes active for one or more clock periods. Once the clock period expires, the output returns to an inactive state. This provides a pulsed output. The output becomes active on the rising clock edge immediately following the overflow event. For more information, see Figure 28-2.

The value of the active and inactive states depends on the polarity bit, POL in the NCO1CON register.

The PF mode is selected by setting the PFM bit in the NCO1CON register.

28.3.1 OUTPUT PULSE-WIDTH CONTROL

When operating in PF mode, the active state of the output can vary in width by multiple clock periods. Various pulse widths are selected with the PWS<2:0> bits in the NCO1CLK register.

When the selected pulse width is greater than the Accumulator overflow time frame, then DDS operation is undefined.

28.4 OUTPUT POLARITY CONTROL

The last stage in the NCO module is the output polarity. The POL bit in the NCO1CON register selects the output polarity. Changing the polarity while the interrupts are enabled will cause an interrupt for the resulting output transition. The NCO output signal is available to most of the other peripherals available on the device.

28.5 Interrupts

When the accumulator overflows (NCO_overflow), the NCO Interrupt Flag bit, NCO1IF, of the PIR4 register is set. To enable the interrupt event (NCO_interrupt), the following bits must be set:

- EN bit of the NCO1CON register
- NCO1IE bit of the PIE4 register
- GIE/GIEH bit of the INTCON0 register

The interrupt must be cleared by software by clearing the NCO1IF bit in the Interrupt Service Routine.

28.6 Effects of a Reset

All of the NCO registers are cleared to zero as the result of a Reset.

28.7 Operation in Sleep

The NCO module operates independently from the system clock and will continue to run during Sleep, provided that the clock source selected remains active.

The HFINTOSC remains active during Sleep when the NCO module is enabled and the HFINTOSC is selected as the clock source, regardless of the system clock source selected.

In other words, if the HFINTOSC is simultaneously selected as the system clock and the NCO clock source, when the NCO is enabled, the CPU will go idle during Sleep, but the NCO will continue to operate and the HFINTOSC will remain active.

This will have a direct effect on the Sleep mode current.

30.11 Register Definitions: Modulation Control

Long bit name prefixes for the Modulation peripheral is shown below. Refer to **Section 1.3.2.2 "Long Bit Names"** for more information.

Peripheral	Bit Name Prefix
MD1	MD1

REGISTER 30-1: MD1CON0: MODULATION CONTROL REGISTER 0

R/W-0/0	U-0	R-0/0	R/W-0/0	U-0	U-0	U-0	R/W-0/0
EN	—	OUT	OPOL	—	—	—	BIT
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 **EN:** Modulator Module Enable bit
1 = Modulator module is enabled and mixing input signals
0 = Modulator module is disabled and has no output

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

bit 5 **OUT:** Modulator Output bit
Displays the current output value of the Modulator module.(1)

bit 4 **OPOL:** Modulator Output Polarity Select bit
1 = Modulator output signal is inverted; idle high output
0 = Modulator output signal is not inverted; idle low output

bit 3-1 **Unimplemented:** Read as '0'

bit 0 **BIT:** Allows software to manually set modulation source input to module(2)
1 = Modulator selects Carrier High
0 = Modulator selects Carrier Low

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: BIT bit must be selected as the modulation source in the MD1SRC register for this operation.

31.21 Register Definitions: UART Control

Long bit name prefixes for the UART peripherals are shown below. Refer to **Section 1.3 "Register and Bit naming conventions"**for more information.

Peripheral	Bit Name Prefix
UART 1	U1
UART 2	U2

REGISTER 31-1: UxCON0: UART CONTROL REGISTER 0

R/W-0/0	R/W/HS/HC-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
BRGS	ABDEN	TXEN	RXEN	MODE<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	HC = Hardware clear

bit 7 **BRGS:** Baud rate Generator Speed Select bit
- 1 = Baud rate generator is high speed with 4 baud clocks per bit
- 0 = Baud rate generator is normal speed with 16 baud clocks per bit

bit 6 **ABDEN:** Auto-baud Detect Enable bit(3)
- 1 = Auto-baud is enabled. Receiver is waiting for Sync character (0x55)
- 0 = Auto-baud is not enabled or auto-baud is complete

bit 5 **TXEN:** Transmit Enable Control bit(2)
- 1 = Transmit is enabled. TX output pin drive is forced on when transmission is active, and controlled by PORT TRIS control when transmission is idle.
- 0 = Transmit is disabled. TX output pin drive is controlled by PORT TRIS control

bit 4 **RXEN:** Receive Enable Control bit(2)
- 1 = Receiver is enabled
- 0 = Receiver is disabled

bit 3-0 **MODE<3:0>:** UART Mode Select bits(1)
- 1111 = Reserved
- 1110 = Reserved
- 1101 = Reserved
- 1100 = LIN Master/Slave mode(4)
- 1011 = LIN Slave-Only mode(4)
- 1010 = DMX mode(4)
- 1001 = DALI Control Gear mode(4)
- 1000 = DALI Control Device mode(4)
- 0111 = Reserved
- 0110 = Reserved
- 0101 = Reserved
- 0100 = Asynchronous 9-bit UART Address mode. 9th bit: 1 = address, 0 = data
- 0011 = Asynchronous 8-bit UART mode with 9th bit even parity
- 0010 = Asynchronous 8-bit UART mode with 9th bit odd parity
- 0001 = Asynchronous 7-bit UART mode
- 0000 = Asynchronous 8-bit UART mode

Note 1: Changing the UART MODE while ON = 1 may cause unexpected results.
2: Clearing TXEN or RXEN will not clear the corresponding buffers. Use TXBE or RXBE to clear the buffers.
3: When MODE = 100x, then ABDEN bit is ignored.
4: UART1 only.

REGISTER 31-8: UxBRGL: UART BAUD RATE GENERATOR LOW 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
BRG<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 **BRG<7:0>:** Least Significant Byte of Baud Rate Generator

REGISTER 31-9: UxBRGH: UART BAUD RATE GENERATOR HIGH 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
BRG<15:8>							
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 7-0 **BRG<15:8>:** Most Significant Byte of Baud Rate Generator

Note 1: The UxBRG registers should only be written when ON = 0.

2: Maximum BRG value when MODE = '100x' and BRGS = 1 is 0x7FFE.

3: Maximum BRG value when MODE = '100x' and BRGS = 0 is 0x1FFE.

FIGURE 32-7: CLOCKING DETAIL-MASTER MODE, CKE/SMP = 0/0

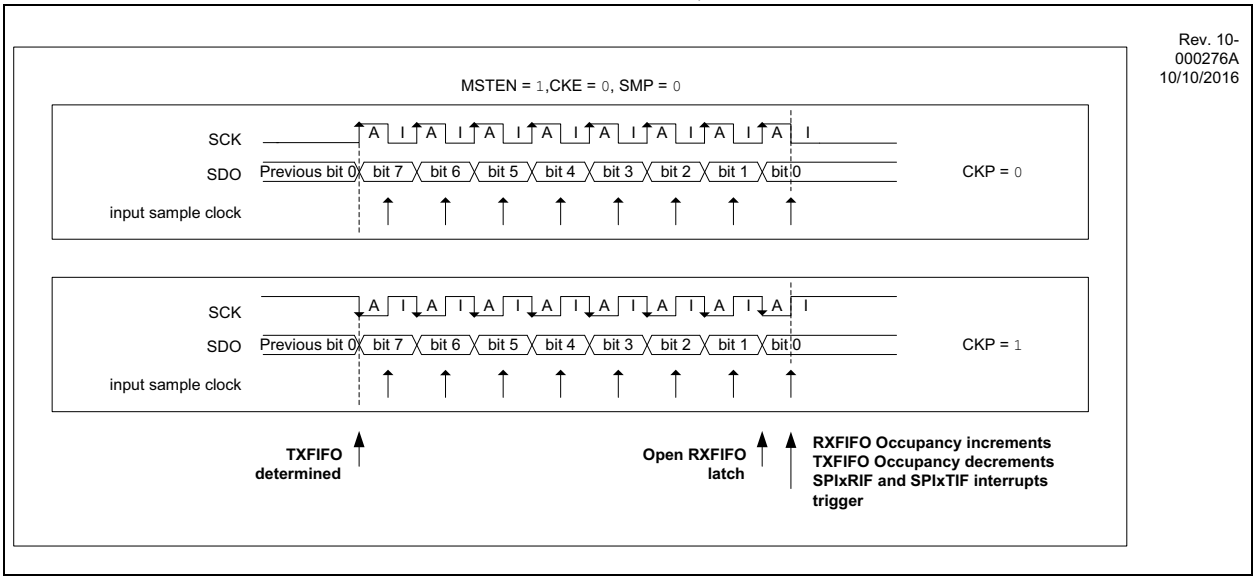


FIGURE 32-8: CLOCKING DETAIL - MASTER MODE, CKE/SMP = 1/1

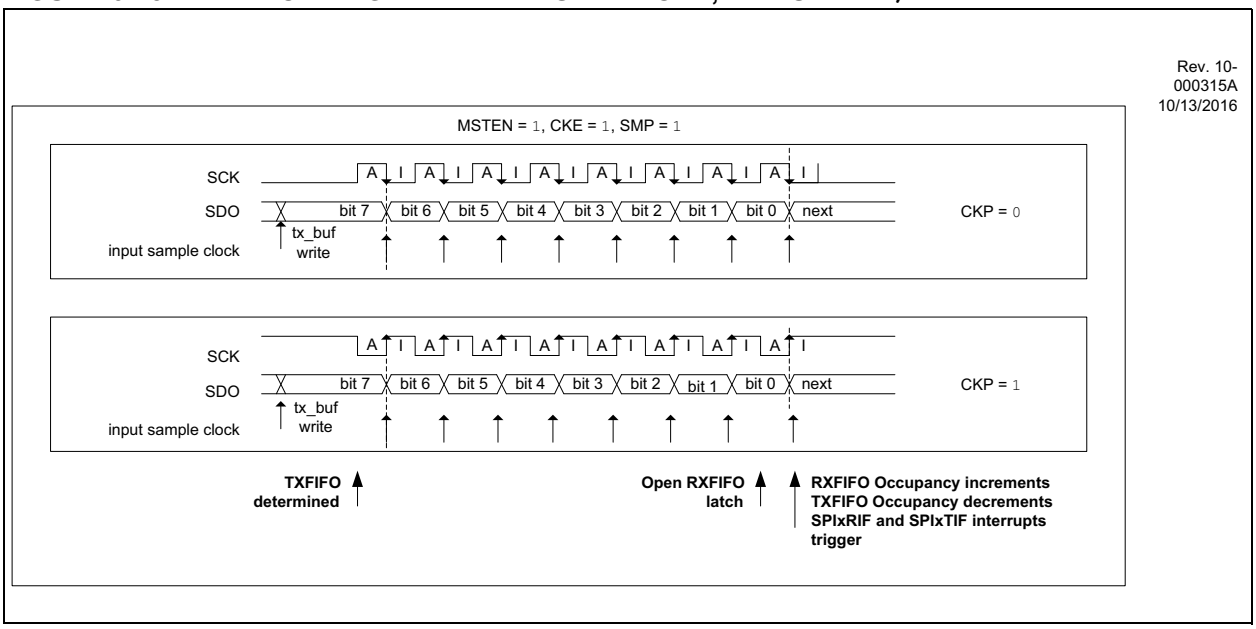


FIGURE 32-9: CLOCKING DETAIL-MASTER MODE, CKE = 0, SMP = 1

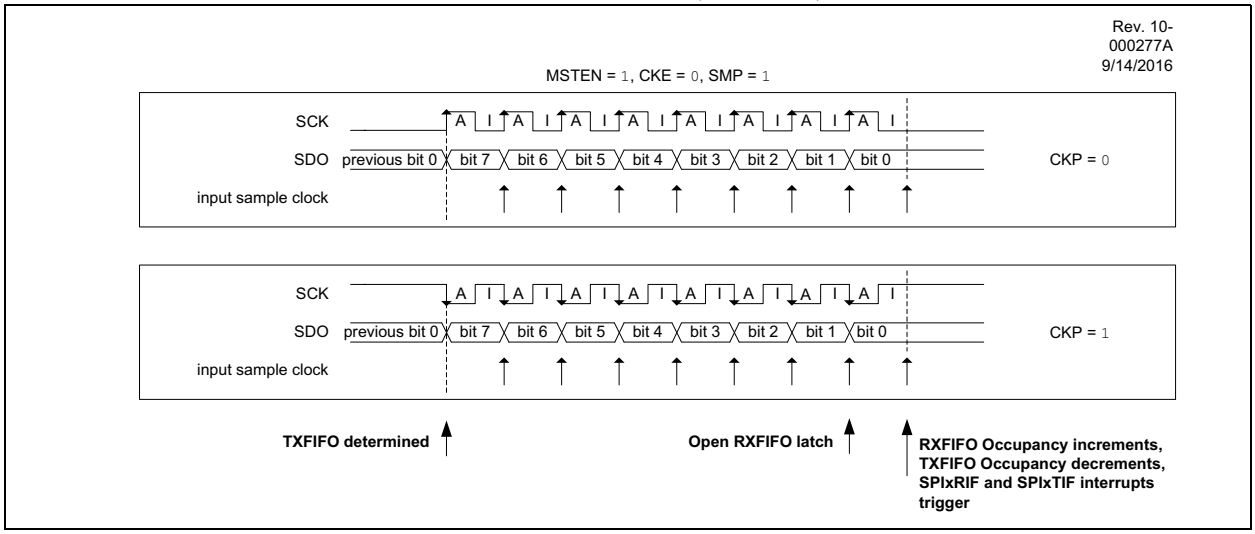
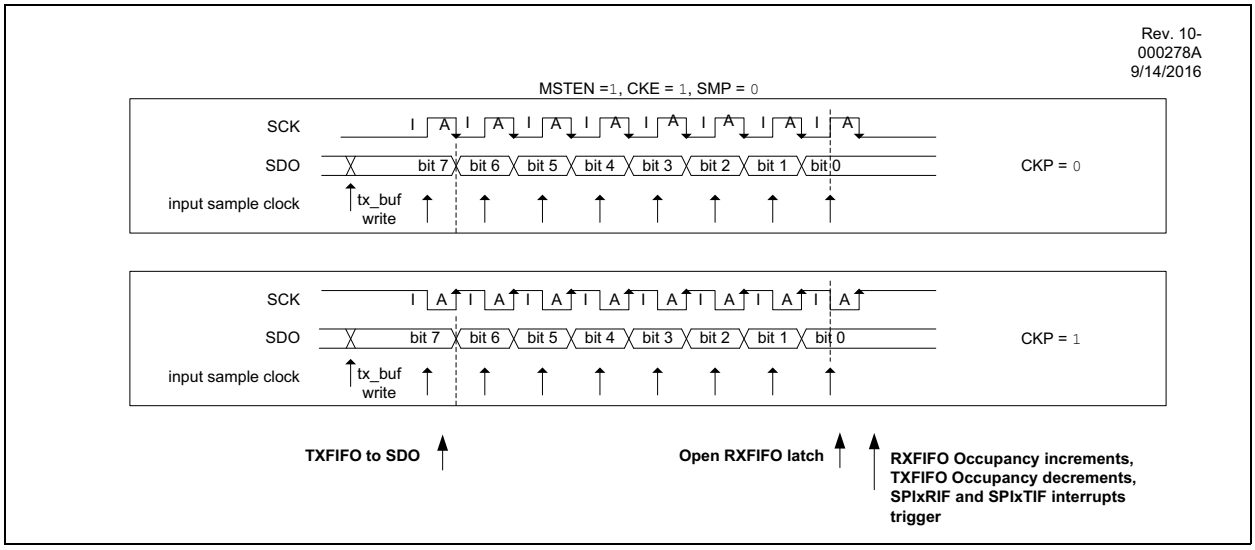


FIGURE 32-10: CLOCKING DETAIL-MASTER MODE, CKE = 1, SMP = 0



32.5.6.3 SCK Start-Up Delay

When starting an SPI data exchange, the master device sets the SS output (either through hardware or software) and then triggers the module to send data. These data triggers are synchronized to the clock selected by the SPIxCLK register before the first SCK pulse appears, usually requiring one or two clocks of the selected clock.

The SPI module includes synchronization delays on SCK generation specifically designed to ensure that the Slave Select output timing is correct, without requiring precision software timing loops.

When the value of the SPIxBAUD register is a small number (indicating higher SCK frequencies), the synchronization delay can be relatively long between setting SS and the first SCK. With larger values of SPIxBAUD (indicating lower SCK frequencies), this delay is much smaller and the first SCK can appear relatively quickly after SS is set.

By default, the SPI module inserts a ½ baud delay (half of the period of the clock selected by the SPIxCLK register) before the first SCK pulse. This allows for systems with a high SPIxBAUD value to have extra set-up time before the first clock. Setting the FST bit in SPIxCON1 removes this additional delay, allowing systems with low SPIxBAUD values (and thus, long synchronization delays) to forego this unnecessary extra delay.

36.0 ANALOG-TO-DIGITAL CONVERTER WITH COMPUTATION (ADC^2) MODULE

The Analog-to-Digital Converter with Computation (ADC^2) allows conversion of an analog input signal to a 12-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 12-bit binary result via successive approximation and stores the conversion result into the ADC result registers (ADRESH:ADRESL register pair).

Additionally, the following features are provided within the ADC module:

- 13-bit Acquisition Timer
- Hardware Capacitive Voltage Divider (CVD) support:
  - 13-bit Precharge Timer
  - Adjustable sample and hold capacitor array
  - Guard ring digital output drive
- Automatic repeat and sequencing:
  - Automated double sample conversion for CVD
  - Two sets of result registers (Result and Previous result)
  - Auto-conversion trigger
  - Internal retrigger
- Computation features:
  - Averaging and Low-Pass Filter functions
  - Reference Comparison
  - 2-level Threshold Comparison
  - Selectable Interrupts

Figure 36-1 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 and upon threshold comparison. These interrupts can be used to wake up the device from Sleep.

REGISTER 36-5: ADSTAT: ADC STATUS REGISTER

R-0/0	R-0/0	R-0/0	R/HS/HC-0/0	U-0	R-0/0	R-0/0	R-0/0
AOV	UTHR	LTHR	MATH	—	STAT<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 **AOV**: ADC Accumulator Overflow bit
1 = ADC accumulator or ADC filter or ERR calculation have overflowed
0 = ADC accumulator, ADC filter and ERR calculation have not overflowed

bit 6 **UTHR**: ADC Module Greater-than Upper Threshold Flag bit
1 = ERR >UTH
0 = ERR ≤ UTH

bit 5 **LTHR**: ADC Module Less-than Lower Threshold Flag bit
1 = ERR < LTH
0 = ERR ≥ LTH

bit 4 **MATH**: ADC Module Computation Status bit
1 = Registers ACC, FLTR, UTH, LTH and the AOV bit are updating or have already updated
0 = Associated registers/bits have not changed since this bit was last cleared

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

bit 2-0 **STAT<2:0>**: ADC Module Cycle Multistage Status bits(1)
111 = ADC module is in 2nd conversion stage
110 = ADC module is in 2nd acquisition stage
101 = ADC module is in 2nd precharge stage
100 = Not used
011 = ADC module is in 1st conversion stage
010 = ADC module is in 1st acquisition stage
001 = ADC module is in 1st precharge stage
000 = ADC module is not converting

Note 1: If CS = 1, and FOSC<FRC, these bits may be invalid.

38.7 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 Table 44-17 and Table 44-19 for more details.

38.8 Analog Input Connection Considerations

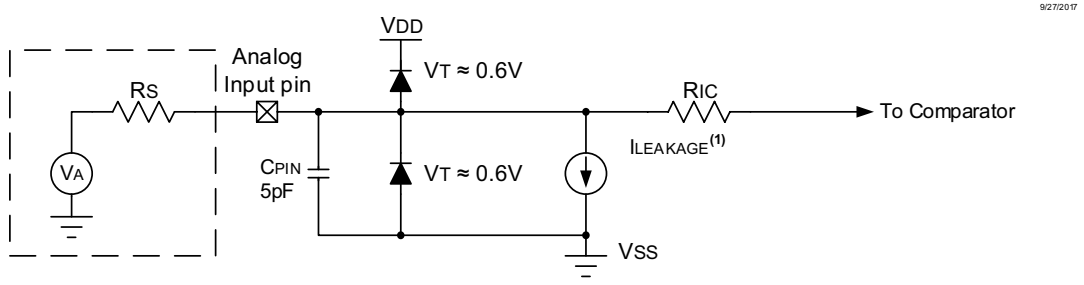
A simplified circuit for an analog input is shown in Figure 38-3. Since the analog input pins share their connection with a digital input, they have reverse biased ESD protection diodes to V_{DD} and V_{SS}. The analog input, therefore, must be between V_{SS} and V_{DD}. If the input voltage deviates from this range by more than 0.6V in either direction, one of the diodes is forward biased and a latch-up may occur.

The maximum source impedance for analog sources is mentioned in Parameter AD08 in Table 44-15. 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 38-3: ANALOG INPUT MODEL



Legend: C_{PIN} = Input Capacitance
$I_{LEAKAGE}$ = Leakage Current at the pin due to various junctions
R_{IC} = Interconnect Resistance
R_S = Source Impedance
V_A = Analog Voltage
V_T = Threshold Voltage

Note 1: See **Section 44.0 "Electrical Specifications"**.

38.9 CWG1 Auto-Shutdown Source

The output of the comparator module can be used as an auto-shutdown source for the CWG1 module. When the output of the comparator is active and the corresponding WGASxE is enabled, the CWG operation will be suspended immediately (see **Section 26.10.1.2 "External Input Source"**).

38.10 ADC Auto-Trigger Source

The output of the comparator module can be used to trigger an ADC conversion. When the ADACT register is set to trigger on a comparator output, an ADC conversion will trigger when the Comparator output goes high.

38.11 TMR2/4/6 Reset

The output of the comparator module can be used to reset Timer2. When the TxRST register is appropriately set, the timer will reset when the Comparator output goes high.

38.12 Operation in Sleep Mode

The comparator module can operate during Sleep. The comparator clock source is based on the Timer1 clock source. If the Timer1 clock source is either the system clock (FOSC) or the instruction clock (FOSC/4), Timer1 will not operate during Sleep, and synchronized comparator outputs will not operate.

A comparator interrupt will wake the device from Sleep. The CxIE bits of the respective PIE register must be set to enable comparator interrupts.

38.13 Register Definitions: Comparator Control

Long bit name prefixes for the Comparators are shown in Table 38-2. Refer to **Section 1.3.2.2 "Long Bit Names"** for more information.

TABLE 38-2:

Peripheral	Bit Name Prefix
C1	C1
C2	C2

REGISTER 38-1: CMxCON0: COMPARATOR x CONTROL REGISTER 0

R/W-0/0	R-0/0	U-0	R/W-0/0	U-0	U-1	R/W-0/0	R/W-0/0
EN	OUT	—	POL	—	—	HYS	SYNC
bit 7							bit 0

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

bit 7 **EN:** Comparator Enable bit
- 1 = Comparator is enabled
- 0 = Comparator is disabled and consumes no active power

bit 6 **OUT:** Comparator Output bit

If POL = 0 (noninverted polarity):
- 1 = CxVP > CxVN
- 0 = CxVP < CxVN

If POL = 1 (inverted polarity):
- 1 = CxVP < CxVN
- 0 = CxVP > CxVN

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

bit 4 **POL:** Comparator Output Polarity Select bit
- 1 = Comparator output is inverted
- 0 = Comparator output is not inverted

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

bit 2 **Unimplemented:** Read as '1'

bit 1 **HYS:** Comparator Hysteresis Enable bit
- 1 = Comparator hysteresis enabled
- 0 = Comparator hysteresis disabled

bit 0 **SYNC:** Comparator Output Synchronous Mode bit
- 1 = Comparator output to Timer1/3/5 and I/O pin is synchronous to changes on Timer1 clock source.
- 0 = Comparator output to Timer1/3/5 and I/O pin is asynchronous
  Output updated on the falling edge of Timer1/3/5 clock source.

INCF **Increment f**

Syntax: INCF f {,d {,a}}

Operands: $0 \le f \le 255$
$d \in [0,1]$
$a \in [0,1]$

Operation: (f) + 1 → dest

Status Affected: C, DC, N, OV, Z

Encoding:

0010	10da	ffff	ffff

Description: The contents of register 'f' are incremented. If 'd' is '0', the result is placed in W. If 'd' is '1', the result is placed back in register 'f' (default).
If 'a' is '0', the Access Bank is selected. If 'a' is '1', the BSR is used to select the GPR bank.
If 'a' is '0' and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever f ≤ 95 (5Fh). See **Section 41.2.3 "Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode"** for details.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	Write to destination

Example: `INCF CNT, 1, 0`

Before Instruction
CNT = FFh
Z = 0
C = ?
DC = ?
After Instruction
CNT = 00h
Z = 1
C = 1
DC = 1

INCFSZ **Increment f, skip if 0**

Syntax: INCFSZ f {,d {,a}}

Operands: $0 \le f \le 255$
$d \in [0,1]$
$a \in [0,1]$

Operation: (f) + 1 → dest,
skip if result = 0

Status Affected: None

Encoding:

0011	11da	ffff	ffff

Description: The contents of register 'f' are incremented. If 'd' is '0', the result is placed in W. If 'd' is '1', the result is placed back in register 'f' (default).
If the result is '0', the next instruction, which is already fetched, is discarded and a NOP is executed instead, making it a 2-cycle instruction.
If 'a' is '0', the Access Bank is selected. If 'a' is '1', the BSR is used to select the GPR bank.
If 'a' is '0' and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever f ≤ 95 (5Fh). See **Section 41.2.3 "Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode"** for details.

Words: 1

Cycles: 1(2)
Note: 3 cycles if skip and followed by a 2-word instruction.

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	Write to destination

If skip:

Q1	Q2	Q3	Q4
No operation	No operation	No operation	No operation

If skip and followed by 2-word instruction:

Q1	Q2	Q3	Q4
No operation	No operation	No operation	No operation
No operation	No operation	No operation	No operation

Example:
```
HERE    INCFSZ  CNT, 1, 0
NZERO   :
ZERO    :
```

Before Instruction
PC = Address (HERE)
After Instruction
CNT = CNT + 1
If CNT = 0;
PC = Address (ZERO)
If CNT ≠ 0;
PC = Address (NZERO)

42.0 REGISTER SUMMARY

TABLE 42-1: REGISTER FILE SUMMARY FOR PIC18(L)F26/27/45/46/47/55/56/57K42 DEVICES

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
3FFFh	TOSU	—	—	—	Top of Stack Upper byte					37
3FFEh	TOSH	Top of Stack High byte								37
3FFDh	TOSL	Top of Stack Low byte								37
3FFCh	STKPTR	—	—	—	Stack Pointer					39
3FFBh	PCLATU	—	—	—	Holding Register for PC Upper byte					36
3FFAh	PCLATH	Holding Register for PC High byte								36
3FF9h	PCL	PC Low byte								36
3FF8h	TBLPTRU	—	—	Program Memory Table Pointer Upper byte						192
3FF7h	TBLPTRH	Program Memory Table Pointer High byte								192
3FF6h	TBLPTRL	Program Memory Table Pointer Low byte								192
3FF5h	TABLAT	Table Latch								192
3FF4h	PRODH	Product Register High byte								187
3FF3h	PRODL	Product Register Low byte								187
3FF2h	—	Unimplemented								
3FF1h	PCON1	—	—	—	—	—	—	$\overline{\text{MEMV}}$	—	91
3FF0h	PCON0	STKOVF	STKUNF	$\overline{\text{WDTWV}}$	$\overline{\text{RWDT}}$	$\overline{\text{RMCLR}}$	$\overline{\text{RI}}$	$\overline{\text{POR}}$	$\overline{\text{BOR}}$	90
3FEFh	INDF0	Uses contents of FSR0 to address data memory – value of FSR0 not changed								60
3FEEh	POSTINC0	Uses contents of FSR0 to address data memory – value of FSR0 post-incremented								61
3FEDh	POSTDEC0	Uses contents of FSR0 to address data memory – value of FSR0 post-decremented								61
3FECh	PREINC0	Uses contents of FSR0 to address data memory – value of FSR0 pre-incremented								61
3FEBh	PLUSW0	Uses contents of FSR0 to address data memory – value of FSR0 pre-incremented – value of FSR0 offset by W								61
3FEAh	FSR0H	—	—	Indirect Data Memory Address Pointer 0 High						61
3FE9h	FSR0L	Indirect Data Memory Address Pointer 0 Low								61
3FE8h	WREG	Working Register								
3FE7h	INDF1	Uses contents of FSR1 to address data memory – value of FSR1 not changed								61
3FE6h	POSTINC1	Uses contents of FSR1 to address data memory – value of FSR1 post-incremented								61
3FE5h	POSTDEC1	Uses contents of FSR1 to address data memory – value of FSR1 post-decremented								61
3FE4h	PREINC1	Uses contents of FSR1 to address data memory – value of FSR1 pre-incremented								61
3FE3h	PLUSW1	Uses contents of FSR1 to address data memory – value of FSR1 pre-incremented – value of FSR1 offset by W								61
3FE2h	FSR1H	—	—	Indirect Data Memory Address Pointer 1 High						61
3FE1h	FSR1L	Indirect Data Memory Address Pointer 1 Low								61
3FE0h	BSR	—	—	Bank Select Register						44
3FDFh	INDF2	Uses contents of FSR2 to address data memory – value of FSR2 not changed								61
3FDEh	POSTINC2	Uses contents of FSR2 to address data memory – value of FSR2 post-incremented								61
3FDDh	POSTDEC2	Uses contents of FSR2 to address data memory – value of FSR2 post-decremented								61
3FDCh	PREINC2	Uses contents of FSR2 to address data memory – value of FSR2 pre-incremented								61
3FDBh	PLUSW2	Uses contents of FSR2 to address data memory – value of FSR2 pre-incremented – value of FSR2 offset by W								61
3FDAh	FSR2H	—	—	Indirect Data Memory Address Pointer 2 High						61
3FD9h	FSR2L	Indirect Data Memory Address Pointer 2 Low								61
3FD8h	STATUS	—	$\overline{\text{TO}}$	$\overline{\text{PD}}$	N	OV	Z	DC	C	58
3FD7h	IVTBASEU	—	—	—	BASE20	BASE19	BASE18	BASE17	BASE16	166
3FD6h	IVTBASEH	BASE15	BASE14	BASE13	BASE12	BASE11	BASE10	BASE9	BASE8	166
3FD5h	IVTBASEL	BASE7	BASE6	BASE5	BASE4	BASE3	BASE2	BASE1	BASE0	166
3FD4h	IVTLOCK	—	—	—	—	—	—	—	IVTLOCKED	168
3FD3h	INTCON1	STAT		—	—	—	—	—	—	136
3FD2h	INTCON0	GIE	GIEL	IPEN	—	—	INT2EDG	INT1EDG	INT0EDG	135

Legend: x = unknown, u = unchanged, — = unimplemented, q = value depends on condition

Note 1: Unimplemented in LF devices.
2: Unimplemented in PIC18(L)F26/27K42.
3: Unimplemented on PIC18(L)F26/27/45/46/47K42 devices.
4: Unimplemented in PIC18(L)F45/55K42.

44.2 Standard Operating Conditions

The standard operating conditions for any device are defined as:

Operating Voltage: $V_{DDMIN} \leq V_{DD} \leq V_{DDMAX}$
Operating Temperature: $T_{A\_MIN} \leq T_A \leq T_{A\_MAX}$

V_{DD} — Operating Supply Voltage(1)

PIC18(L)F26/27/45/46/47/55/56/57K42

- V_{DDMIN} (Fosc ≤ 16 MHz) ..... +1.8V
- V_{DDMIN} (Fosc ≤ 32 MHz) ..... +2.5V
- V_{DDMIN} (Fosc ≤ 64 MHz) ..... +2.7V
- V_{DDMAX} ..... +3.6V

PIC18F26/45/46/55/56K42

- V_{DDMIN} (Fosc ≤ 16 MHz) ..... +2.3V
- V_{DDMIN} (Fosc ≤ 32 MHz) ..... +2.5V
- V_{DDMIN} (Fosc ≤ 64 MHz) ..... +3.0V
- V_{DDMAX} ..... +5.5V

T_A — Operating Ambient Temperature Range

Industrial Temperature

- T_{A_MIN} ..... -40°C
- T_{A_MAX} ..... +85°C

Extended Temperature

- T_{A_MIN} ..... -40°C
- T_{A_MAX} ..... +125°C

Note 1: See Parameter Supply Voltage, DS Characteristics: Supply Voltage.