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	28KB (16K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 17x12b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SOIC (0.295", 7.50mm Width)
Supplier Device Package	20-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf18446-e-so

Table 7-3. Core Registers

Addresses in BANKx	Core Registers
n00h or n80h	INDF0
n01h or n81h	INDF1
n02h or n82h	PCL
n03 or n83h	STATUS
n04h or n84h	FSR0L
n05h or n85h	FSR0H
n06h or n86h	FSR1L
n07h or n87h	FSR1H
n08h or n88h	BSR
n09h or n89h	WREG
n0Ah or n8Ah	PCLATH
n0Bh or n8Bh	INTCON

7.3.2.1 STATUS Register

The STATUS register contains:

- the arithmetic status of the ALU
- the Reset status

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

For example, `CLRF STATUS` will clear bits <4:3> and <1:0>, and set the Z bit. This leaves the STATUS register as '`000u u1uu`' (where u = unchanged).

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

Important: The C and DC bits operate as $\overline{\text{Borrow}}$ and $\overline{\text{Digit Borrow}}$ out bits, respectively, in subtraction.

Related Links

STATUS

7.8.1 INDF0

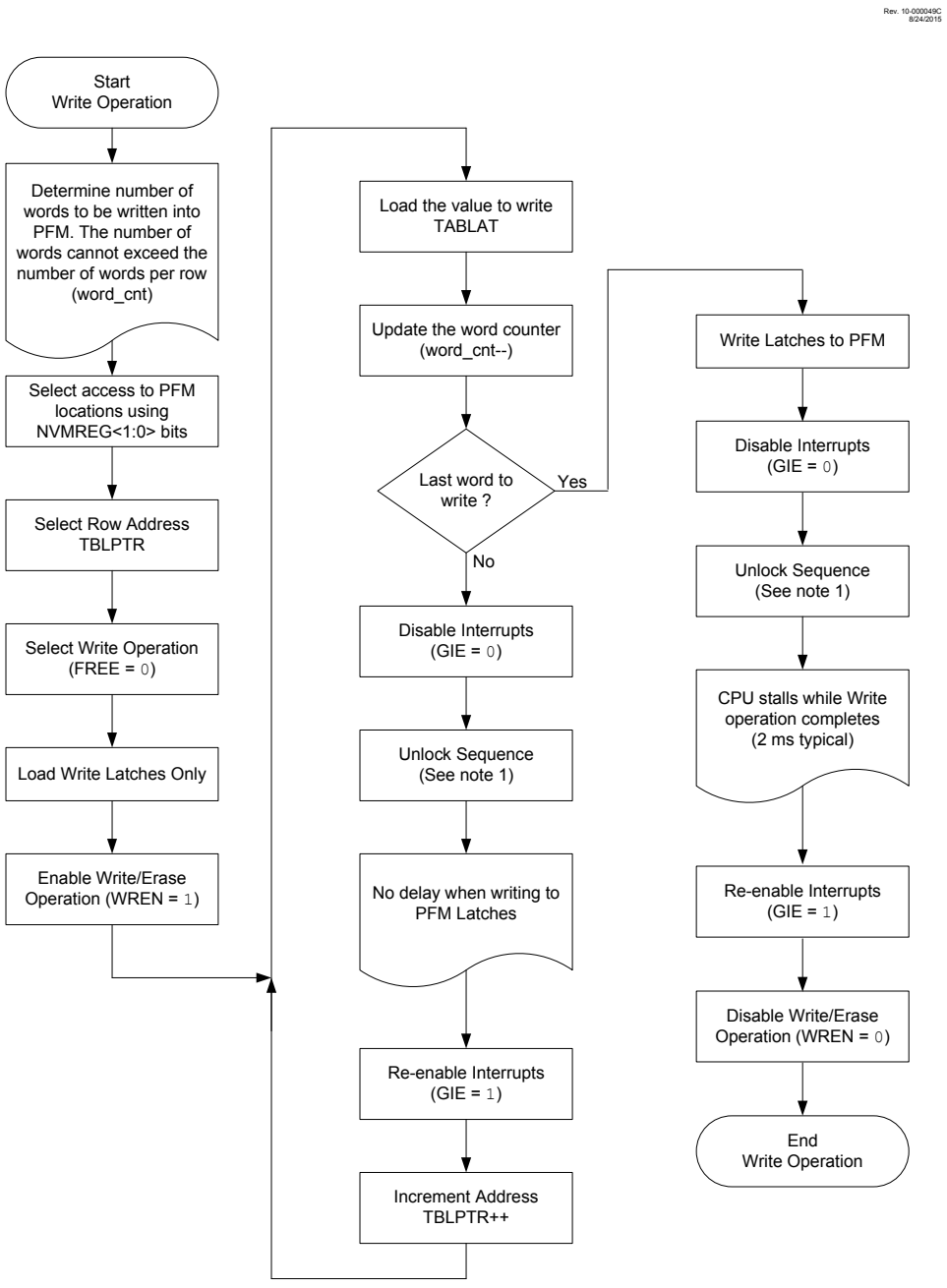
Name: INDF0
Address: 0x00 + n*0x80 [n=0..63]

Indirect Data Register. This is a virtual register. The GPR/SFR register addressed by the FSR0 register is the target for all operations involving the INDF0 register.

Bit	7	6	5	4	3	2	1	0
	INDF0[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 7:0 – INDF0[7:0]
Indirect data pointed to by the FSR0 register
Related Links
Core Registers

Figure 13-5. Program Flash Memory Flowchart



Note:

1. See NVM Unlock Sequence Flowchart

Writing to Program Flash Memory

```
   ; This write routine assumes the following:
   ; 1. 64 bytes of data are loaded, starting at the address in DATA_ADDR
   ; 2. Each word of data to be written is made up of two adjacent bytes in
DATA_ADDR,
   ; stored in little endian format
   ; 3. A valid starting address (the least significant bits = 00000) is
```

14.6.15 WPUC

Name: WPUC
Address: 0x1F4F

Weak Pull-up Register

Bit	7	6	5	4	3	2	1	0
	WPUC7	WPUC6	WPUC5	WPUC4	WPUC3	WPUC2	WPUC1	WPUC0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 0, 1, 2, 3, 4, 5, 6, 7 – WPUCn Weak Pull-up PORTC Control bits

Value	Description
1	Weak Pull-up enabled
0	Weak Pull-up disabled

Note: Bits WPUC6 and WPUC7 available on 20-pin or higher pin-count devices only; Bits unimplemented for lower pin-count devices.

21.7.1 DAC1CON0

Name: DAC1CON0
Address: 0x90E

DAC Control Register

Bit	7	6	5	4	3	2	1	0
	EN		OE1		PSS[1:0]			NSS
Access	R/W		R/W		R/W	R/W		R/W
Reset	0		0		0	0		0

Bit 7 – EN DAC Enable bit

Value	Description
1	DAC is enabled
0	DAC is disabled

Bit 5 – OE1 DAC Voltage Output Enable bit

Value	Description
1	DAC voltage level is output on the DAC1OUT1 pin
0	DAC voltage level is disconnected from the DAC1OUT1 pin

Bits 3:2 – PSS[1:0] DAC Positive Source Select bit

Value	Description
11	Reserved
10	FVR buffer
01	$V_{REF}+$
00	AV_{DD}

Bit 0 – NSS DAC Negative Source Select bit

Value	Description
1	$V_{REF}-$
0	AV_{SS}

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

23. (CMP) Comparator Module

Comparators are used to interface analog circuits to a digital circuit by comparing two analog voltages and providing a digital indication of their relative magnitudes. Comparators are very useful mixed-signal building blocks because they provide analog functionality independent of program execution. The PIC16(L)F18426/46 devices have 2 comparators (C1/C2).

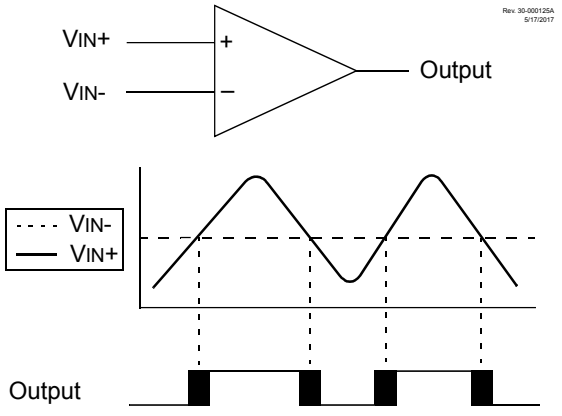
The analog comparator module includes the following features:

- Programmable input selection
- Programmable output polarity
- Rising/falling output edge interrupts
- Wake-up from Sleep
- CWG Auto-shutdown source
- Selectable voltage reference
- ADC Auto-trigger
- Odd numbered timers (Timer1, Timer3, etc.) Gate
- Even numbered timers (Timer2, Timer4, etc.) Reset
- CCP Capture Mode Input
- DSM Modulator Source
- Input and Window Signal-to-Signal Measurement Timer

23.1 Comparator Overview

A single comparator is shown in Figure 23-1 along with the relationship between the analog input levels and the digital output. When the analog voltage at V_{IN}+ is less than the analog voltage at V_{IN}-, the output of the comparator is a digital low level. When the analog voltage at V_{IN}+ is greater than the analog voltage at V_{IN}-, the output of the comparator is a digital high level.

Figure 23-1. Single Comparator



Note:

1. The black areas of the output of the comparator represent the uncertainty due to input offsets and response time.

26.14.1 TxCON

Name: TxCON
Address: 0x20E,0x214,0x21A

Timer Control Register

Bit	7	6	5	4	3	2	1	0
			CKPS[1:0]			$\overline{\text{SYNC}}$	RD16	ON
Access			R/W	R/W		R/W	R/W	R/W
Reset			0	0		0	0	0

Bits 5:4 – CKPS[1:0] Timer Input Clock Prescale Select bits
Reset States: POR/BOR = 00
All Other Resets = uu

Value	Description
11	1:8 Prescale value
10	1:4 Prescale value
01	1:2 Prescale value
00	1:1 Prescale value

Bit 2 – $\overline{\text{SYNC}}$ Timer External Clock Input Synchronization Control bit
Reset States: POR/BOR = 0
All Other Resets = u

Value	Condition	Description
X	CS = $F_{OSC}/4$ or F_{OSC}	This bit is ignored. Timer uses the incoming clock as is.
1	Else	Do not synchronize external clock input
0	Else	Synchronize external clock input with system clock

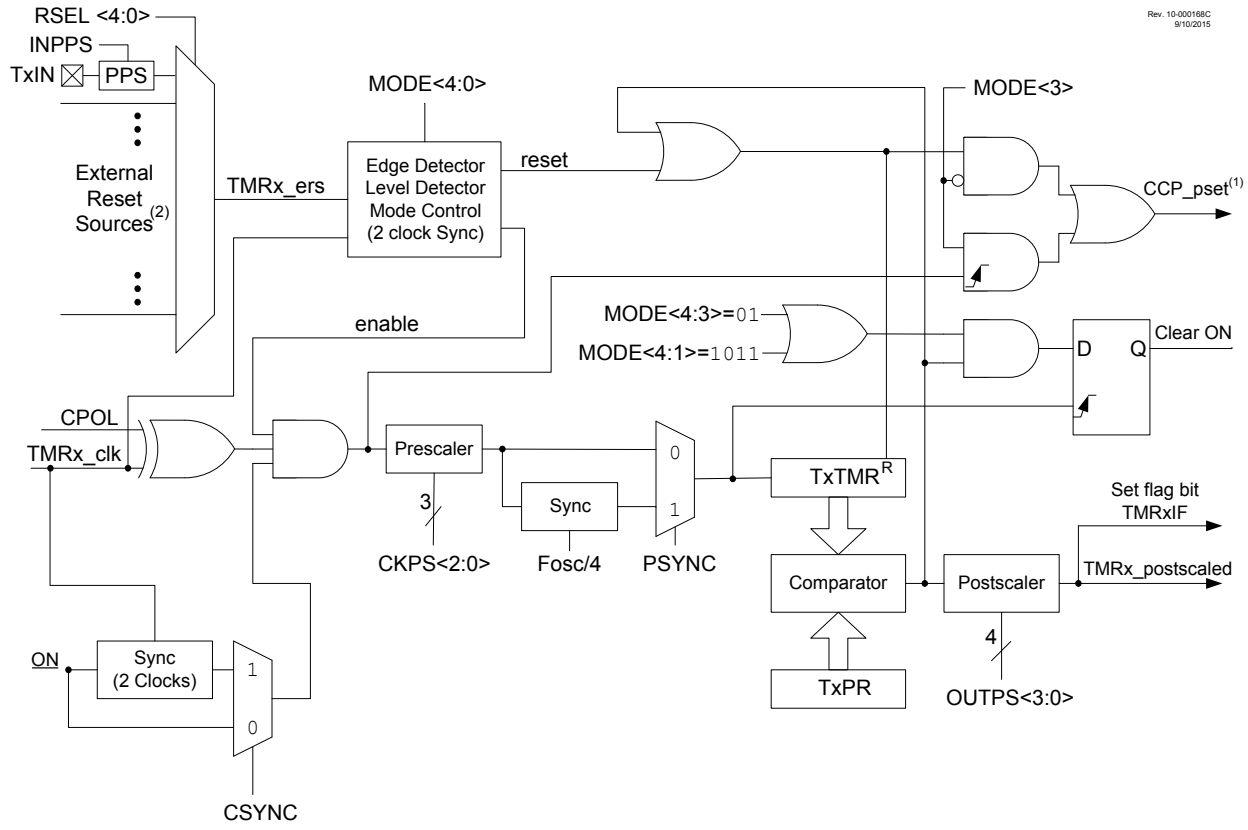
Bit 1 – RD16 16-Bit Read/Write Mode Enable bit
Reset States: POR/BOR = 0
All Other Resets = u

Value	Description
1	Enables register read/write of Timer in one 16-bit operation
0	Enables register read/write of Timer in two 8-bit operations

Bit 0 – ON Timer On bit
Reset States: POR/BOR = 0
All Other Resets = u

Value	Description
1	Enables Timer
0	Disables Timer

Figure 27-1. Timer2 with Hardware Limit Timer (HLT) Block Diagram



Note:

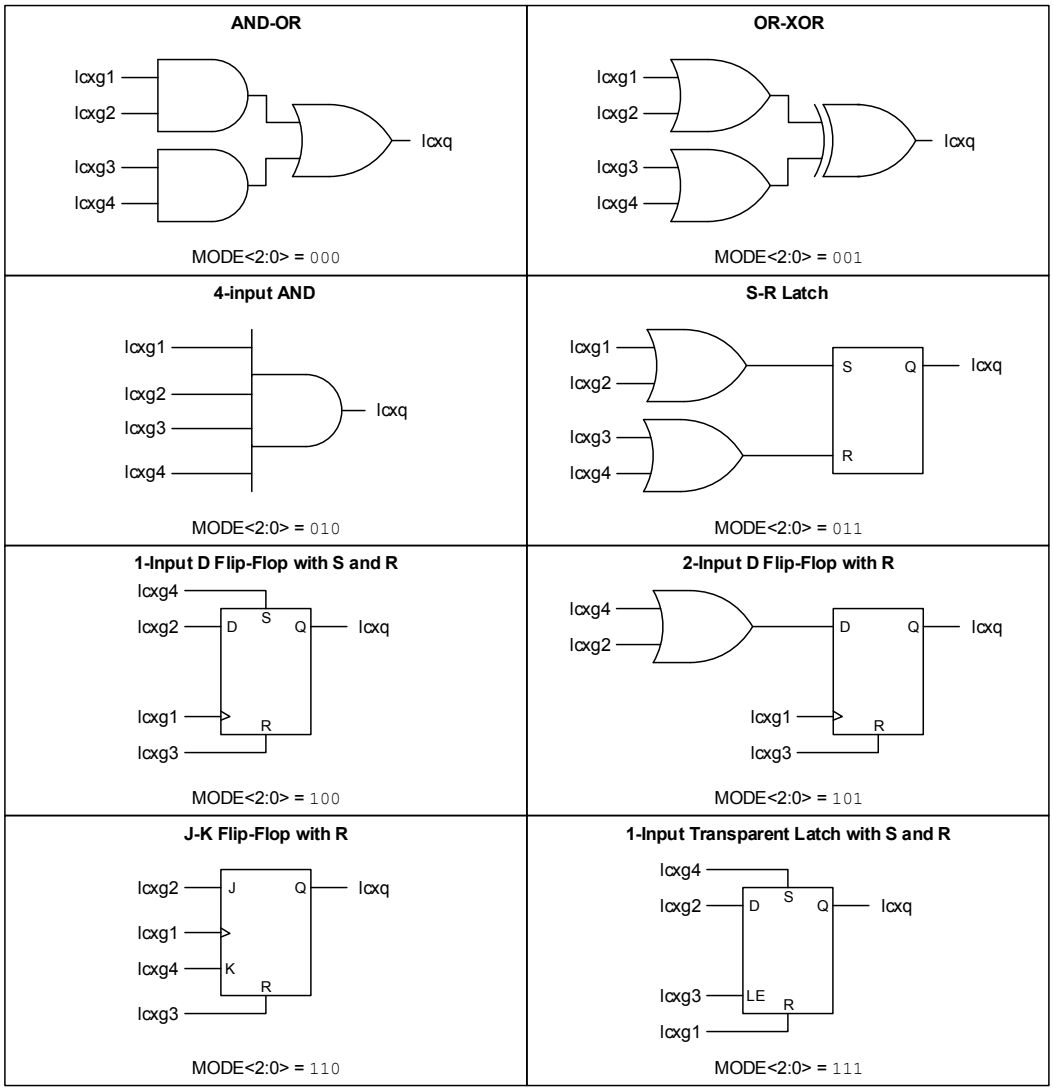
1. Signal to the CCP to trigger the PWM pulse.
2. See TxRST for external Reset sources.

Table 27-1. Clock Source Selection

CS<3:0>	Clock Source		
	Timer2	Timer4	Timer6
1111	Reserved	Reserved	Reserved
1110	CLC4_out	CLC4_out	CLC4_out
1101	CLC3_out	CLC3_out	CLC3_out
1100	CLC2_out	CLC2_out	CLC2_out
1011	CLC1_out	CLC1_out	CLC1_out
1010	ZCD1_output	ZCD1_output	ZCD1_output
1001	NCO1_out	NCO1_out	NCO1_out
1000	CLKR	CLKR	CLKR
0111	SOSC	SOSC	SOSC

Figure 33-3. Programmable Logic Functions

Rev. 10-000122B
9/13/2016



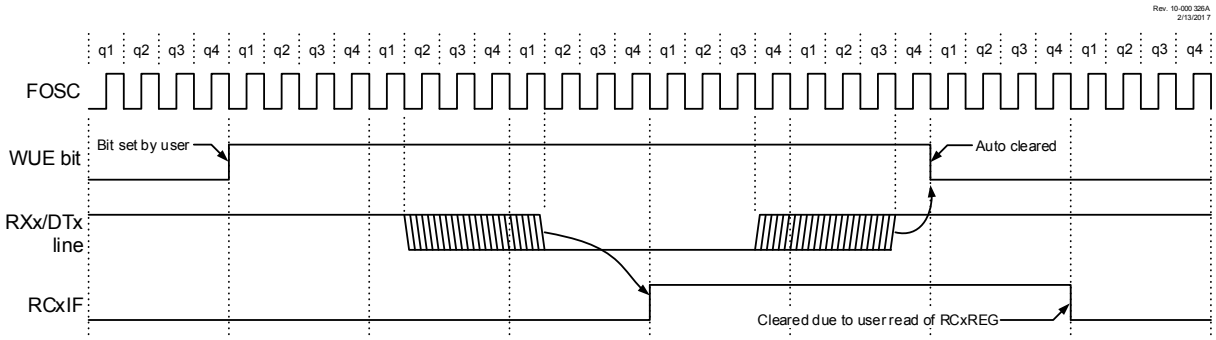
33.1.4 Output Polarity

The last stage in the Configurable Logic Cell is the output polarity. Setting the POL bit inverts the output signal from the logic stage. Changing the polarity while the interrupts are enabled will cause an interrupt for the resulting output transition.

33.2 CLC Interrupts

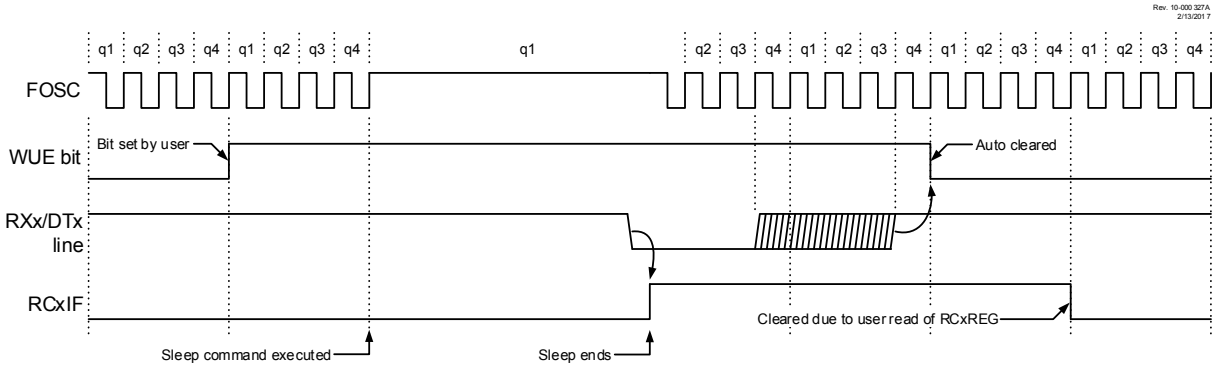
An interrupt will be generated upon a change in the output value of the CLCx when the appropriate interrupt enables are set. A rising edge detector and a falling edge detector are present in each CLC for this purpose.

Figure 36-8. Auto-Wake-up Bit (WUE) Timing During Normal Operation



Note 1: The EUSART remains in idle while the WUE bit is set.

Figure 36-9. Auto-Wake-up Bit (WUE) Timings During Sleep



Note 1: The EUSART remains in idle while the WUE bit is set.

36.2.4 Break Character Sequence

The EUSART module has the capability of sending the special Break character sequences that are required by the LIN bus standard. A Break character consists of a Start bit, followed by 12 ‘0’ bits and a Stop bit.

To send a Break character, set the SENDB and TXEN bits of the TXxSTA register. The Break character transmission is then initiated by a write to the TXxREG. The value of data written to TXxREG will be ignored and all ‘0’s will be transmitted.

The SENDB bit is automatically reset by hardware after the corresponding Stop bit is sent. This allows the user to preload the transmit FIFO with the next transmit byte following the Break character (typically, the Sync character in the LIN specification).

The TRMT bit of the TXxSTA register indicates when the transmit operation is active or idle, just as it does during normal transmission. See Figure 36-10 for more detail.

36.2.4.1 Break and Sync Transmit Sequence

The following sequence will start a message frame header made up of a Break, followed by an auto-baud Sync byte. This sequence is typical of a LIN bus master.

1. Configure the EUSART for the desired mode.
2. Set the TXEN and SENDB bits to enable the Break sequence.
3. Load the TXxREG with a dummy character to initiate transmission (the value is ignored).

36.6.3 BAUDxCON

Name: BAUDxCON
Address: 0x011F

Baud Rate Control Register

Bit	7	6	5	4	3	2	1	0
	ABDOVF	RCIDL		SCKP	BRG16		WUE	ABDEN
Access	RO	RO		RW	RW		RW	RW
Reset	0	0		0	0		0	0

Bit 7 – ABDOVF Auto-Baud Detect Overflow bit

Value	Condition	Description
1	SYNC=0	Auto-baud timer overflowed
0	SYNC=0	Auto-baud timer did not overflow
X	SYNC=1	Don't care

Bit 6 – RCIDL Receive Idle Flag bit

Value	Condition	Description
1	SYNC=0	Receiver is Idle
0	SYNC=0	Start bit has been received and the receiver is receiving
X	SYNC=1	Don't care

Bit 4 – SCKP Synchronous Clock Polarity Select bit

Value	Condition	Description
1	SYNC=0	Idle state for transmit (TX) is a low level (transmit data inverted)
0	SYNC=0	Idle state for transmit (TX) is a high level (transmit data is non-inverted)
1	SYNC=1	Data is clocked on rising edge of the clock
0	SYNC=1	Data is clocked on falling edge of the clock

Bit 3 – BRG16 16-bit Baud Rate Generator Select bit

Value	Description
1	16-bit Baud Rate Generator is used
0	8-bit Baud Rate Generator is used

Bit 1 – WUE Wake-up Enable bit

Value	Condition	Description
1	SYNC=0	Receiver is waiting for a falling edge. Upon falling edge no character will be received and flag RCxIF will be set. WUE will automatically clear after RCxIF is set.
0	SYNC=0	Receiver is operating normally
X	SYNC=1	Don't care

Bit 0 – ABDEN Auto-Baud Detect Enable bit

Figure 37-3. Gated Timer Mode, Repeat Acquisition Timing Diagram

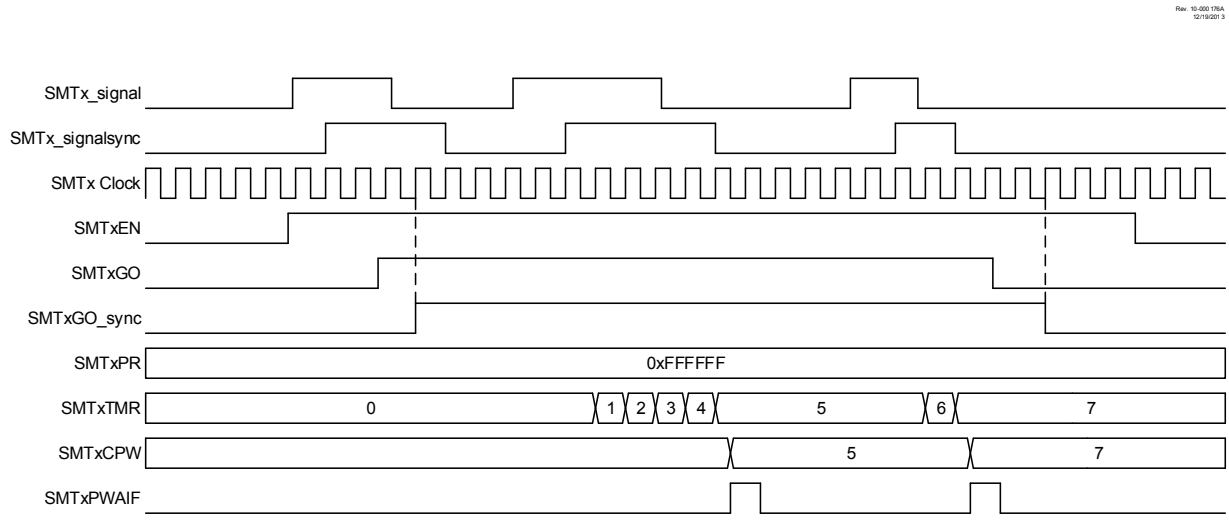
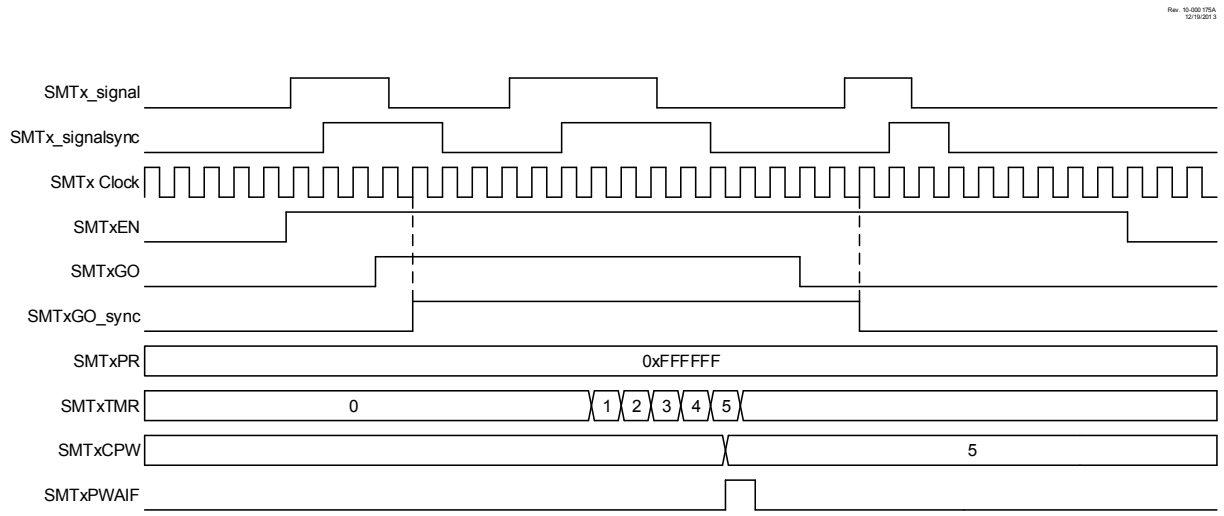


Figure 37-4. Gated Timer Mode, Single Acquisition Timing Diagram



37.1.6.3 Period and Duty Cycle Measurement Mode

In this mode, either the duty cycle or period (depending on polarity) of the input signal can be acquired relative to the SMT clock. The CPW register is updated on a falling edge of the signal, and the CPR register is updated on a rising edge of the signal, along with the SMTxTMR resetting to 0x000001. In addition, the SMTxGO bit is reset on a rising edge when the SMT is in single acquisition mode. See figures below.

Figure 37-13. Time of Flight Mode, Repeat Acquisition Timing Diagram

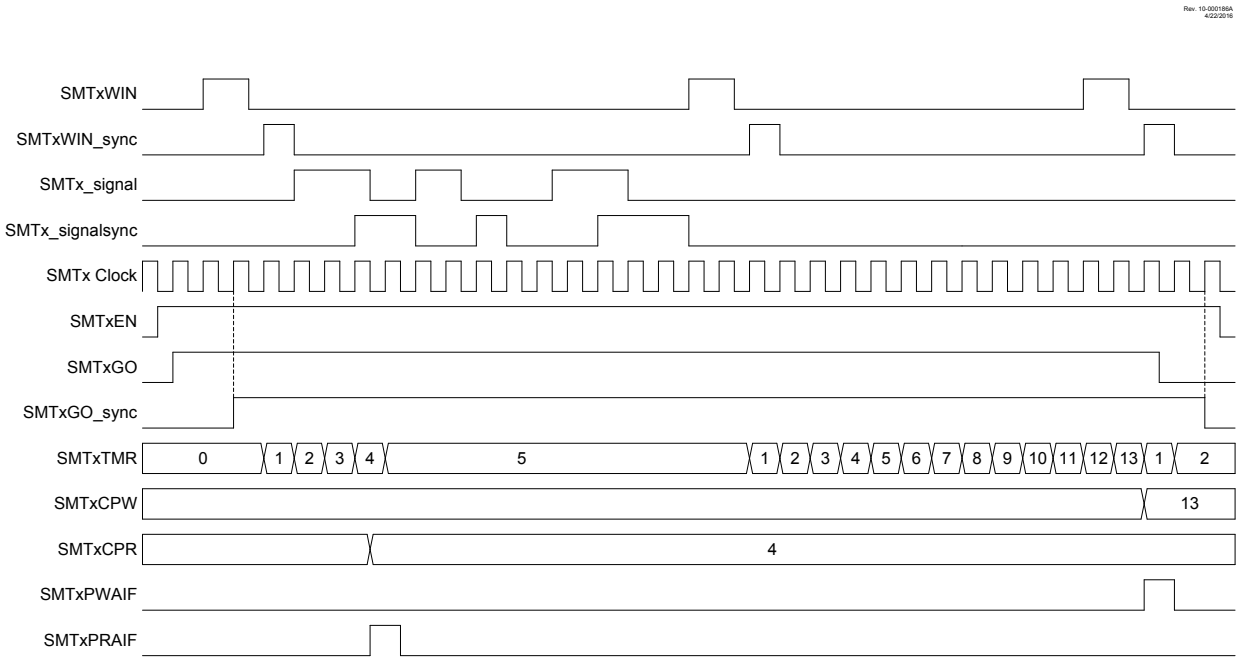
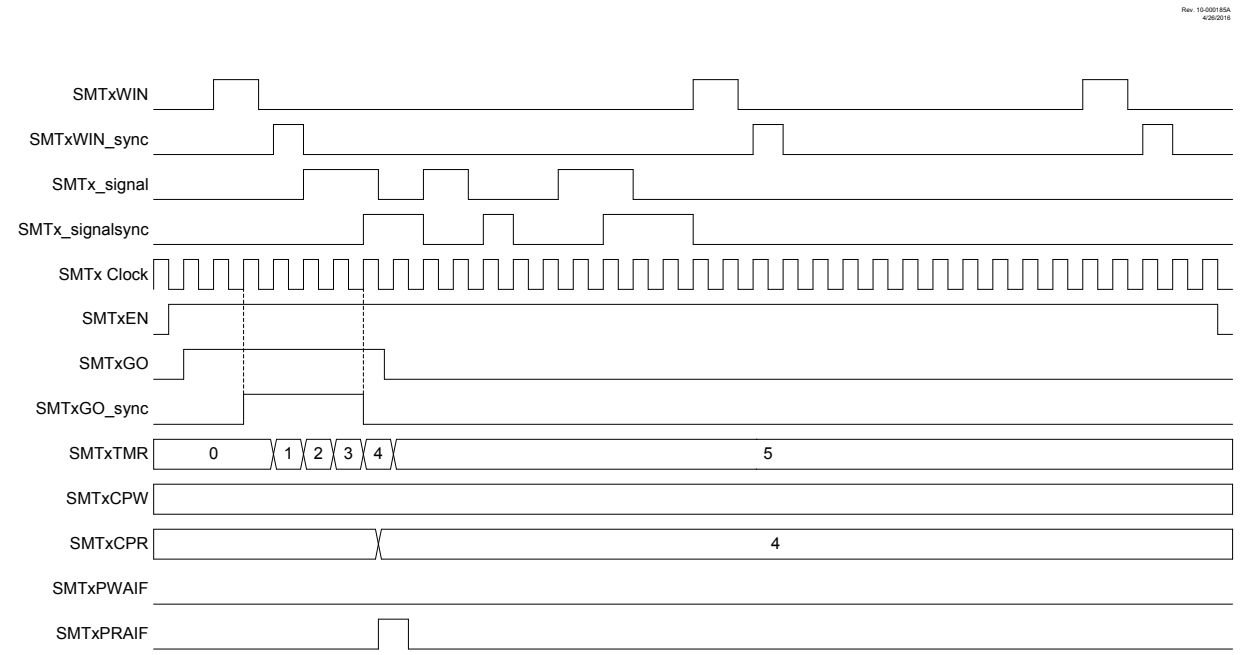


Figure 37-14. Time of Flight Mode, Single Acquisition Timing Diagram



37.1.6.8 Capture Mode

This mode captures the Timer value based on a rising or falling edge on the window 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 window signal, and updates the value of the SMTxCPW register on each falling edge of the window signal. The timer is not reset by any hardware conditions in this mode and must be reset by software, if desired. See figures below.

Offset	Name	Bit Pos.								
0x0A89	WREG	7:0	WREG[7:0]							
0x0A8A	PCLATH	7:0		PCLATH[6:0]						
0x0A8B	INTCON	7:0	GIE	PEIE						INTEDG
0x0A8C ... 0x0AFF	Reserved									
0x0B00	INDF0	7:0	INDF0[7:0]							
0x0B01	INDF1	7:0	INDF1[7:0]							
0x0B02	PCL	7:0	PCL[7:0]							
0x0B03	STATUS	7:0				$\overline{TO}$	$\overline{PD}$	Z	DC	C
0x0B04	FSR0	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x0B06	FSR1	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x0B08	BSR	7:0			BSR[5:0]					
0x0B09	WREG	7:0	WREG[7:0]							
0x0B0A	PCLATH	7:0		PCLATH[6:0]						
0x0B0B	INTCON	7:0	GIE	PEIE						INTEDG
0x0B0C ... 0x0B7F	Reserved									
0x0B80	INDF0	7:0	INDF0[7:0]							
0x0B81	INDF1	7:0	INDF1[7:0]							
0x0B82	PCL	7:0	PCL[7:0]							
0x0B83	STATUS	7:0				$\overline{TO}$	$\overline{PD}$	Z	DC	C
0x0B84	FSR0	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x0B86	FSR1	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x0B88	BSR	7:0			BSR[5:0]					
0x0B89	WREG	7:0	WREG[7:0]							
0x0B8A	PCLATH	7:0		PCLATH[6:0]						
0x0B8B	INTCON	7:0	GIE	PEIE						INTEDG
0x0B8C ... 0x0BFF	Reserved									
0x0C00	INDF0	7:0	INDF0[7:0]							
0x0C01	INDF1	7:0	INDF1[7:0]							
0x0C02	PCL	7:0	PCL[7:0]							
0x0C03	STATUS	7:0				$\overline{TO}$	$\overline{PD}$	Z	DC	C
0x0C04	FSR0	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x0C06	FSR1	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x0C08	BSR	7:0			BSR[5:0]					
0x0C09	WREG	7:0	WREG[7:0]							

MOVF	Move f
Syntax:	[*label*] MOVF f, d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	f → dest
Status Affected:	Z
Description:	The contents of register f is moved to a destination dependent upon the status of d. If d = 0, destination is W register. If d = 1, the destination is file register f itself. d = 1 is useful to test a file register since status flag Z is affected.
Words:	1
Cycles:	1

Example:	`MOVF FSR, 0`
After Instruction W = value in FSR register Z = 1	

MOVIW	Move INDFn to W
Syntax:	[*label*] MOVIW ++FSRn [*label*] MOVIW --FSRn [*label*] MOVIW FSRn++ [*label*] MOVIW FSRn-- [*label*] MOVIW k[FSRn]
Operands:	n ∈ [0,1] mm ∈ [00,01,10,11] -32 ≤ k ≤ 31
Operation:	INDFn → (W) Effective address is determined by • FSR + 1 (preincrement) • FSR - 1 (predecrement) • FSR + k (relative offset) After the Move, the FSR value will be either: • FSR + 1 (all increments) • FSR - 1 (all decrements) • Unchanged

MOVIW	Move INDFn to W		
Status Affected:	Z		
	MODE	SYNTAX	mm
	Preincrement	++FSRn	00
	Predecrement	--FSRn	01
	Postincrement	FSRn++	10
	Postdecrement	FSRn--	11
Description:	This instruction is used to move data between W and one of the indirect registers (INDFn). Before/after this move, the pointer (FSRn) is updated by pre/post incrementing/ decrementing it. The INDFn registers are not physical registers. Any instruction that accesses an INDFn register actually accesses the register at the address specified by the FSRn. FSRn is limited to the range 0000h - FFFFh. Incrementing/decrementing it beyond these bounds will cause it to wrap-around.		

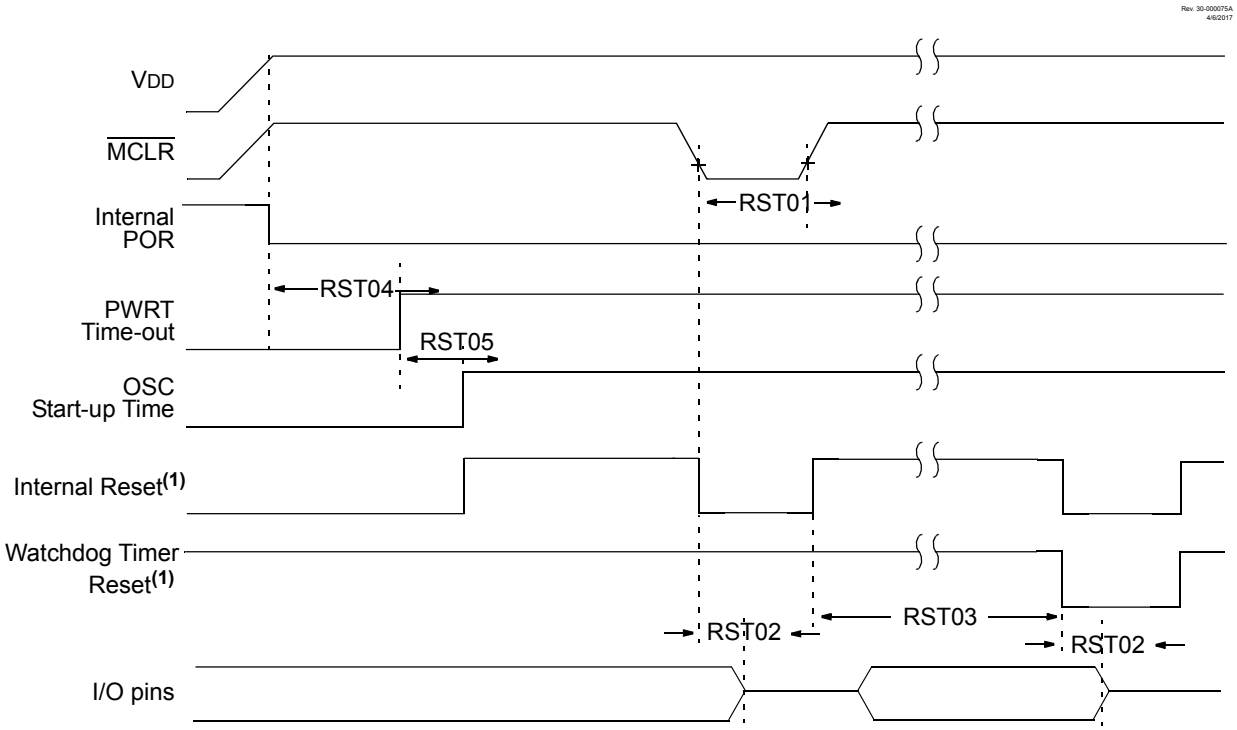
MOVLB	Move literal to BSR
Syntax:	[*label*] MOVLB k
Operands:	0 ≤ k ≤ 127
Operation:	k → BSR
Status Affected:	None
Description:	The 6-bit literal ‘k’ is loaded into the Bank Select Register (BSR).

MOVLP	Move literal to PCLATH
Syntax:	[*label*] MOVLP k
Operands:	0 ≤ k ≤ 127
Operation:	k → PCLATH
Status Affected:	None
Description:	The 7-bit literal ‘k’ is loaded into the PCLATH register.

MOVLW	Move literal to W
Syntax:	[*label*] MOVLW k
Operands:	0 ≤ k ≤ 255
Operation:	k → (W)
Status Affected:	None

42.4.5 Reset, WDT, Oscillator Start-up Timer, Power-up Timer, Brown-Out Reset and Low-Power Brown-Out Reset Specifications

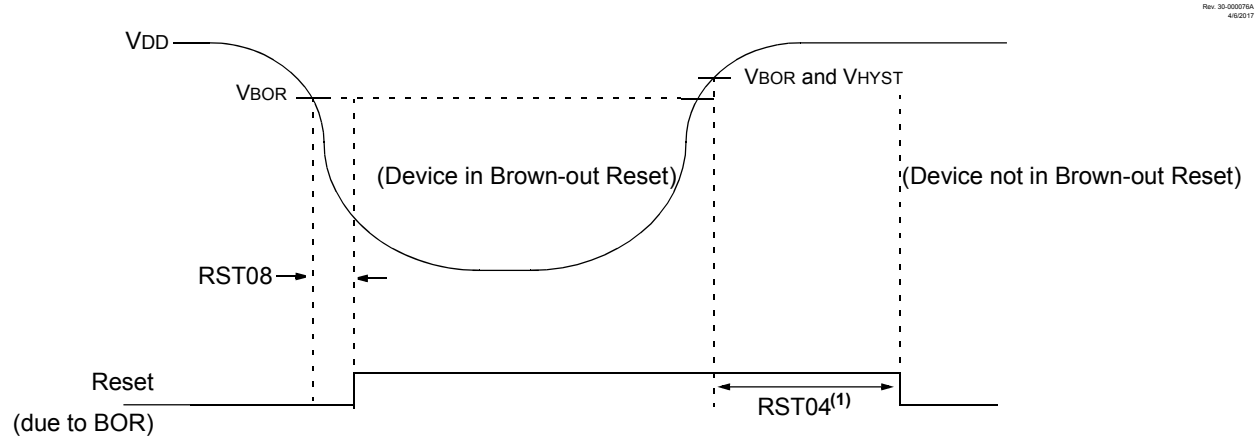
Figure 42-8. Reset, Watchdog Timer, Oscillator Start-up Timer and Power-up Timer Timing



Note:

1. Asserted low.

Figure 42-9. Brown-out Reset Timing and Characteristics



Note:

1. Only if $\overline{\text{PWRTE}}$ bit in the Configuration Word register is programmed to '1'; 2 ms delay if $\overline{\text{PWRTE}}$ = 0.

Figure 42-10. ADC Conversion Timing (ADC Clock F_{OSC}-Based)

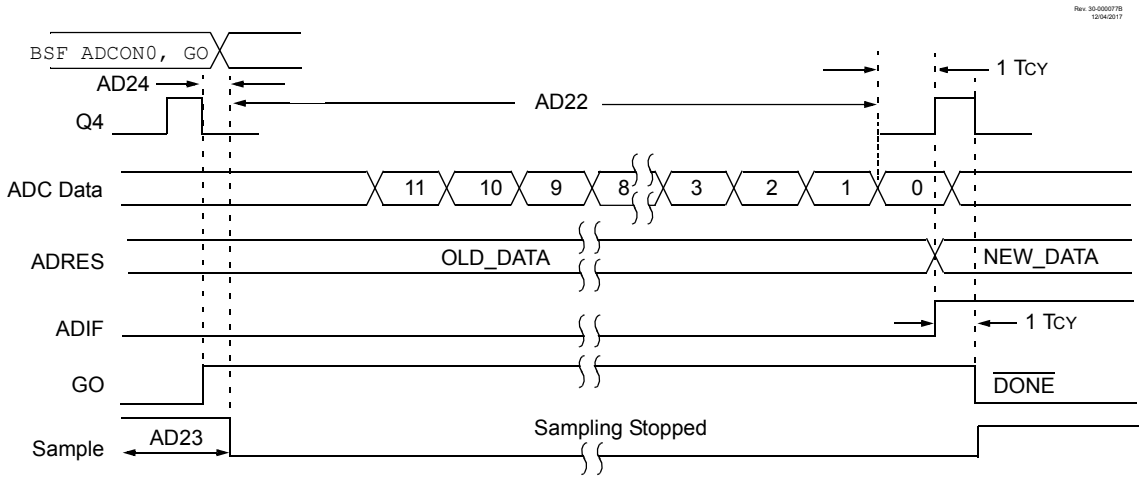
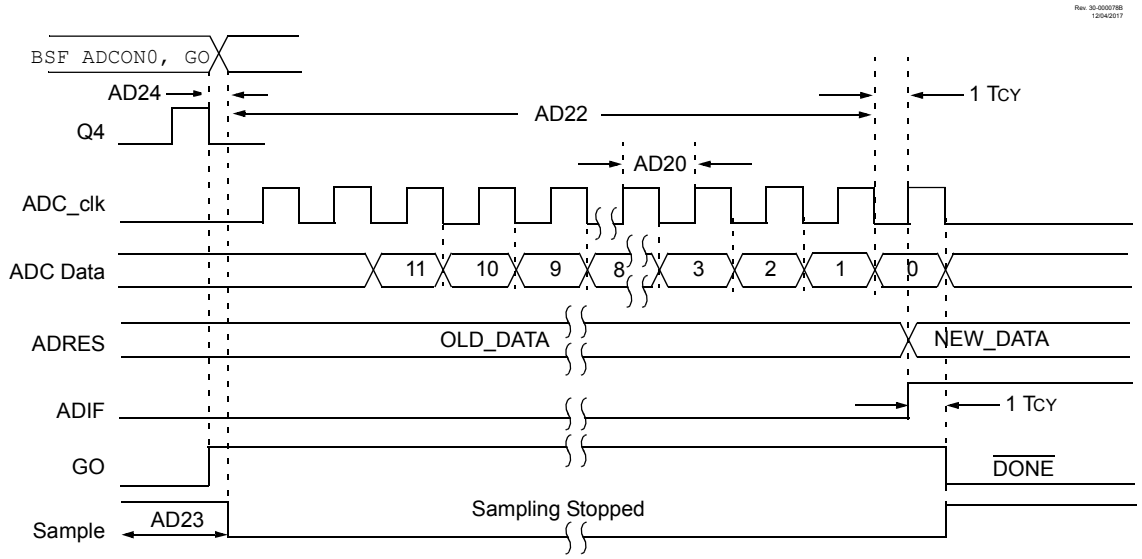


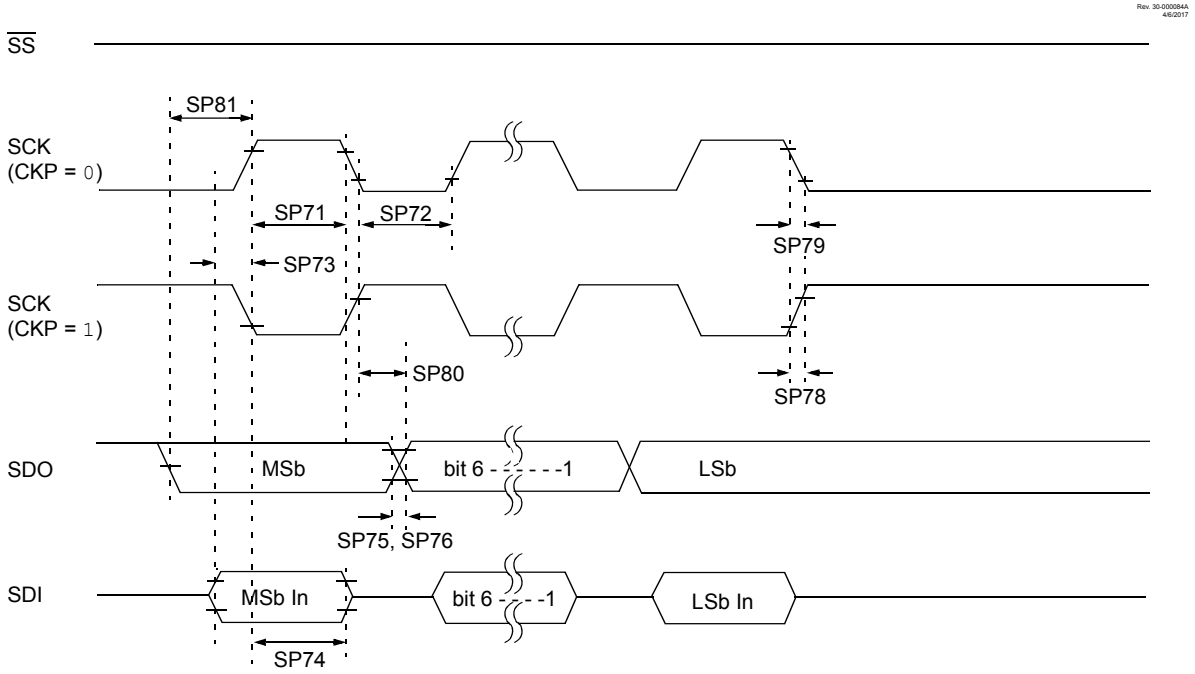
Figure 42-11. ADC Conversion Timing (ADC Clock from ADCRC)



Note:

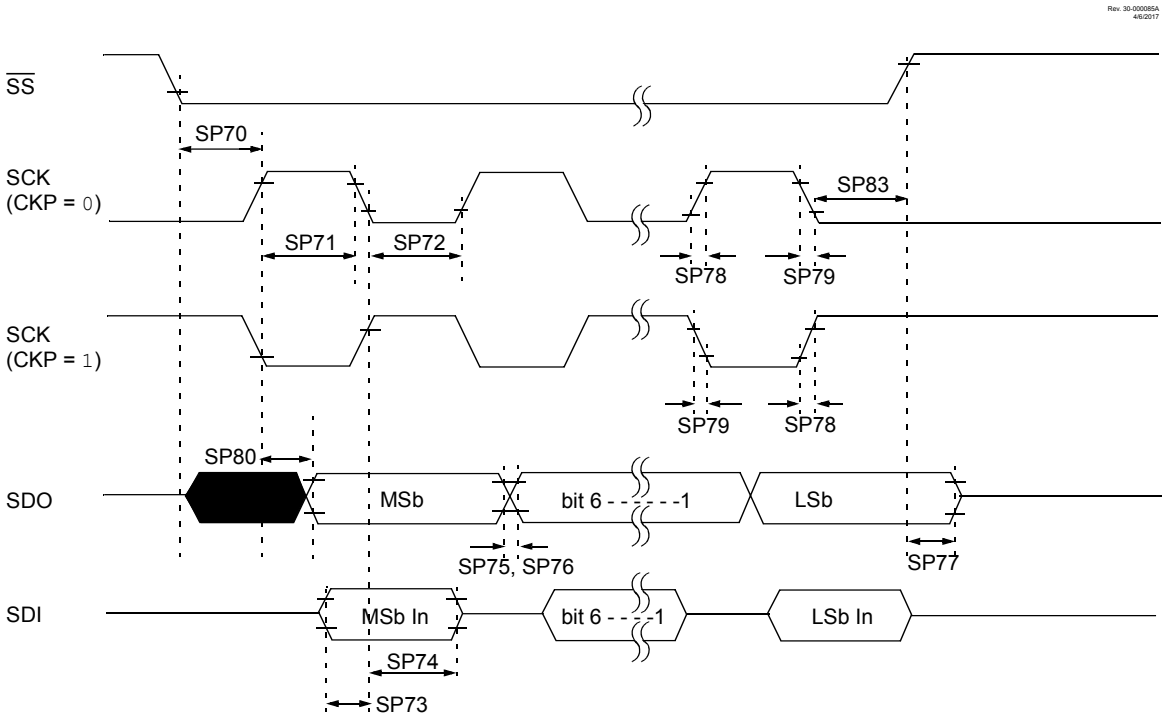
1. If the ADC clock source is selected as ADCRC, a time of T_{CY} is added before the ADC clock starts. This allows the `SLEEP` instruction to be executed.

Figure 42-18. SPI Master Mode Timing (CKE = 1, SMP = 1)



Note: Refer to Figure 42-4 for load conditions.

Figure 42-19. SPI Slave Mode Timing (CKE = 0)



Note: Refer to Figure 42-4 for load conditions.