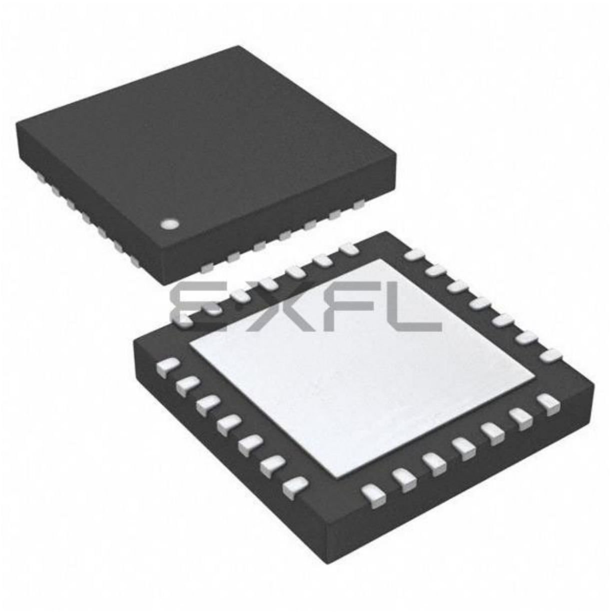
Microchip Technology - PIC16LF18854-E/ML Datasheet



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	25
Program Memory Size	7KB (4K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 24x10b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	28-VQFN Exposed Pad
Supplier Device Package	28-QFN (6x6)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf18854-e-ml

5.6 RESET Instruction

A RESET instruction will cause a device Reset. The $\overline{\text{RI}}$ bit in the PCON register will be set to '0'. See Table 5-4 for default conditions after a RESET instruction has occurred.

5.7 Stack Overflow/Underflow Reset

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

5.8 Programming Mode Exit

Upon exit of In-Circuit Serial Programming (ICSP) mode, the device will behave as if a POR had just occurred (the device does not reset upon run time self-programming/erase operations).

5.9 Power-Up Timer

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

The Power-up Timer is controlled by the $\overline{\text{PWRTE}}$ bit of the Configuration Words.

The Power-up Timer provides a nominal 64 ms time out on POR or Brown-out Reset. The device is held in Reset as long as PWRT is active. The PWRT delay allows additional time for the VDD to rise to an acceptable level. The Power-up Timer is enabled by clearing the $\overline{\text{PWRTE}}$ bit in the Configuration Words. The Power-up Timer starts after the release of the POR and BOR. For additional information, refer to Application Note AN607, "*Power-up Trouble Shooting*" (DS00607).

5.10 Start-up Sequence

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

1. Power-up Timer runs to completion (if enabled).
2. Oscillator start-up timer runs to completion (if required for oscillator source).
3. $\overline{\text{MCLR}}$ must be released (if enabled).

The total time-out will vary based on oscillator configuration and Power-up Timer Configuration. See **Section 6.0 "Oscillator Module (with Fail-Safe Clock Monitor)"** for more information.

The Power-up Timer and oscillator start-up timer run independently of $\overline{\text{MCLR}}$ Reset. If $\overline{\text{MCLR}}$ is kept low long enough, the Power-up Timer and oscillator start-up timer will expire. Upon bringing $\overline{\text{MCLR}}$ high, the device will begin execution after 10 FOSC cycles (see Figure 5-3). This is useful for testing purposes or to synchronize more than one device operating in parallel.

6.2 Clock Source Types

Clock sources can be classified as external or internal.

External clock sources rely on external circuitry for the clock source to function. Examples are: oscillator modules (ECH, ECM, ECL mode), quartz crystal resonators or ceramic resonators (LP, XT and HS modes).

Internal clock sources are contained within the oscillator module. The internal oscillator block has two internal oscillators and a dedicated Phase Lock Loop (PLL) that are used to generate internal system clock sources. The High-Frequency Internal Oscillator (HFINTOSC) can produce a range from 1 to 32 MHz. The Low-Frequency Internal Oscillator (LFINTOSC) generates a 31 kHz frequency. The external oscillator block can also be used with the PLL. See **Section 6.2.1.4 "4x PLL"** for more details.

The system clock can be selected between external or internal clock sources via the NOSC bits in the OSCCON1 register. See **Section 6.3 "Clock Switching"** for additional information.

6.2.1 EXTERNAL CLOCK SOURCES

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

- Program the RSTOSC<2:0> bits in the Configuration Words to select an external clock source that will be used as the default system clock upon a device Reset
- Write the NOSC<2:0> and NDIV<4:0> bits in the OSCCON1 register to switch the system clock source

See **Section 6.3 "Clock Switching"**for more information.

6.2.1.1 EC Mode

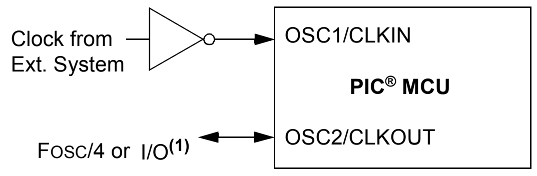
The External Clock (EC) mode allows an externally generated logic level signal to be the system clock source. When operating in this mode, an external clock source is connected to the OSC1 input. OSC2/CLKOUT is available for general purpose I/O or CLKOUT. Figure 6-2 shows the pin connections for EC mode.

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

- ECH – High power, 4-32 MHz
- ECM – Medium power, 0.1-4 MHz
- ECL – Low power, 0-0.1 MHz

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

FIGURE 6-2: EXTERNAL CLOCK (EC) MODE OPERATION



Note 1: Output depends upon $\overline{\text{CLKOUTEN}}$ bit of the Configuration Words.

6.2.1.2 LP, XT, HS Modes

The LP, XT and HS modes support the use of quartz crystal resonators or ceramic resonators connected to OSC1 and OSC2 (Figure 6-3). The three modes select a low, medium or high gain setting of the internal inverter-amplifier to support various resonator types and speed.

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

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

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

Figure 6-3 and Figure 6-4 show typical circuits for quartz crystal and ceramic resonators, respectively.

REGISTER 7-7: PIE5: PERIPHERAL INTERRUPT ENABLE REGISTER 5

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	R/W-0/0	R/W-0/0	R/W-0/0
CLC4IE	CLC3IE	CLC2IE	CLC1IE	—	TMR5GIE	TMR3GIE	TMR1GIE
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 = Hardware set

bit 7 **CLC4IE:** CLC4 Interrupt Enable bit
1 = CLC4 interrupt enabled
0 = CLC4 interrupt disabled

bit 6 **CLC3IE:** CLC3 Interrupt Enable bit
1 = CLC3 interrupt enabled
0 = CLC3 interrupt disabled

bit 5 **CLC2IE:** CLC2 Interrupt Enable bit
1 = CLC2 interrupt enabled
0 = CLC2 interrupt disabled

bit 4 **CLC1IE:** CLC1 Interrupt Enable bit
1 = CLC1 interrupt enabled
0 = CLC1 interrupt disabled

bit 3 **Unimplemented:** Read as ‘0’

bit 2 **TMR5GIE:** Timer5 Gate Interrupt Enable bit
1 = Enables the Timer5 gate acquisition interrupt
0 = Disables the Timer5 gate acquisition interrupt

bit 1 **TMR3GIE:** Timer3 Gate Interrupt Enable bit
1 = Enables the Timer3 gate acquisition interrupt
0 = Disables the Timer3 gate acquisition interrupt

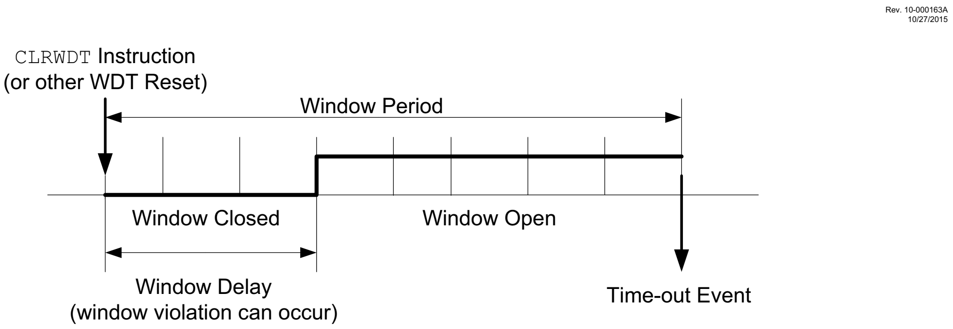
bit 0 **TMR1GIE:** Timer1 Gate Interrupt Enable bit
1 = Enables the Timer1 gate acquisition interrupt
0 = Disables the Timer1 gate acquisition interrupt

Note: Bit PEIE of the INTCON register must be set to enable any peripheral interrupt controlled by registers PIE1-PIE8.

TABLE 9-2: WDT CLEARING CONDITIONS

Conditions	WDT
WDTE<1:0> = 00	Cleared
WDTE<1:0> = 01 and SEN = 0	
WDTE<1:0> = 10 and enter Sleep	
CLRWDT Command	
Oscillator Fail Detected	
Exit Sleep + System Clock = T1OSC, EXTRC, INTOSC, EXTCLK	
Change INTOSC divider (IRCF bits)	Unaffected

FIGURE 9-2: WINDOW PERIOD AND DELAY



REGISTER 9-3: WDTPSL: WDT PRESCALE SELECT LOW BYTE REGISTER (READ-ONLY)

R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0
PSCNT<7:0>[1]							
bit 7							bit 0

Legend:

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

bit 7-0 **PSCNT<7:0>**: Prescale Select Low Byte bits[1]

Note 1: The 18-bit WDT prescale value, PSCNT<17:0> includes the WDTPSL, WDTPSH and the lower bits of the WDTTMR registers. PSCNT<17:0> is intended for debug operations and should be read during normal operation.

REGISTER 9-4: WDTPSH: WDT PRESCALE SELECT HIGH BYTE REGISTER (READ-ONLY)

R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0
PSCNT<15:8>[1]							
bit 7							bit 0

Legend:

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

bit 7-0 **PSCNT<15:8>**: Prescale Select High Byte bits[1]

Note 1: The 18-bit WDT prescale value, PSCNT<17:0> includes the WDTPSL, WDTPSH and the lower bits of the WDTTMR registers. PSCNT<17:0> is intended for debug operations and should be read during normal operation.

REGISTER 9-5: WDTTMR: WDT TIMER REGISTER (READ-ONLY)

U-0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0
—	WDTTMR<3:0>				STATE	PSCNT<17:16>[1]	
bit 7							bit 0

Legend:

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

bit 7 **Unimplemented:** Read as ‘0’

bit 6-3 **WDTTMR<3:0>**: Watchdog Timer Value bits

bit 2 **STATE**: WDT Armed Status bit
1 = WDT is armed
0 = WDT is not armed

bit 1-0 **PSCNT<17:16>**: Prescale Select Upper Byte bits[1]

Note 1: The 18-bit WDT prescale value, PSCNT<17:0> includes the WDTPSL, WDTPSH and the lower bits of the WDTTMR registers. PSCNT<17:0> is intended for debug operations and should be read during normal operation.

EXAMPLE 10-4: WRITING TO PROGRAM FLASH MEMORY (PFM)

```
; 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 loaded in ADDRH:ADDRL
; 4. ADDRH and ADDRL are located in common RAM (locations 0x70 - 0x7F)
; 5. NVM interrupts are not taken into account

        BANKSEL       NVMADRH
        MOVF          ADDRH,W
        MOVWF         NVMADRH                 ; Load initial address
        MOVF          ADDRL,W
        MOVWF         NVMADRL
        MOVLW         LOW DATA_ADDR           ; Load initial data address
        MOVWF         FSR0L
        MOVLW         HIGH DATA_ADDR
        MOVWF         FSR0H
        BCF           NVMCON1,NVMREGS         ; Set Program Flash Memory as write location
        BSF           NVMCON1,WREN            ; Enable writes
        BSF           NVMCON1,LWLO            ; Load only write latches

LOOP
        MOVIW         FSR0++
        MOVWF         NVMDATL                 ; Load first data byte
        MOVIW         FSR0++
        MOVWF         NVMDATH                 ; Load second data byte
        MOVF          NVMADRL,W
        XORLW         0x1F                    ; Check if lower bits of address are 00000
        ANDLW         0x1F                    ; and if on last of 32 addresses
        BTFSC         STATUS,Z                ; Last of 32 words?
        GOTO          START_WRITE             ; If so, go write latches into memory
        CALL          UNLOCK_SEQ              ; If not, go load latch
        INCF          NVMADRL,F               ; Increment address
        GOTO          LOOP

START_WRITE
        BCF           NVMCON1,LWLO            ; Latch writes complete, now write memory
        CALL          UNLOCK_SEQ              ; Perform required unlock sequence
        BCF           NVMCON1,WREN            ; Disable writes

UNLOCK_SEQ
        MOVLW         55h
        BCF           INTCON,GIE              ; Disable interrupts
        MOVWF         NVMCON2                 ; Begin unlock sequence
        MOVLW         AAh
        MOVWF         NVMCON2
        BSF           NVMCON1,WR
        BSF           INTCON,GIE              ; Unlock sequence complete, re-enable interrupts
        return
```

REGISTER 10-5: NVMCON1: NONVOLATILE MEMORY CONTROL 1 REGISTER

U-0	R/W-0/0	R/W-0/0	R/W/HC-0/0	R/W/HC-x/q	R/W-0/0	R/S/HC-0/0	R/S/HC-0/0
—	NVMREGS	LWLO	FREE	WRERR(1,2,3)	WREN	WR(4,5,6)	RD(7)
bit 7							bit 0

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

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

bit 6 **NVMREGS:** Configuration Select bit
1 = Access EEPROM, Configuration, User ID and Device ID Registers
0 = Access PFM

bit 5 **LWLO:** Load Write Latches Only bit
When FREE = 0:
1 = The next WR command updates the write latch for this word within the row; no memory operation is initiated.
0 = The next WR command writes data or erases
Otherwise: The bit is ignored

bit 4 **FREE:** PFM Erase Enable bit
When NVMREGS:NVMADR points to a PFM location:
1 = Performs an erase operation with the next WR command; the 32-word pseudo-row containing the indicated address is erased (to all 1s) to prepare for writing.
0 = All write operations have completed normally

bit 3 **WRERR:** Program/Erase Error Flag bit(1,2,3)
This bit is normally set by hardware.
1 = A write operation was interrupted by a Reset, interrupted unlock sequence, or WR was written to one while NVMADR points to a write-protected address.
0 = The program or erase operation completed normally

bit 2 **WREN:** Program/Erase Enable bit
1 = Allows program/erase cycles
0 = Inhibits programming/erasing of program Flash

bit 1 **WR:** Write Control bit(4,5,6)
When NVMREG:NVMADR points to a EEPROM location:
1 = Initiates an erase/program cycle at the corresponding EEPROM location
0 = NVM program/erase operation is complete and inactive
When NVMREG:NVMADR points to a PFM location:
1 = Initiates the operation indicated by Table 10-4
0 = NVM program/erase operation is complete and inactive
Otherwise: This bit is ignored

bit 0 **RD:** Read Control bit(7)
1 = Initiates a read at address = NVMADR1, and loads data to NVMDAT Read takes one instruction cycle and the bit is cleared when the operation is complete. The bit can only be set (not cleared) in software.
0 = NVM read operation is complete and inactive

Note 1: Bit is undefined while WR = 1 (during the EEPROM write operation it may be '0' or '1').
2: Bit must be cleared by software; hardware will not clear this bit.
3: Bit may be written to '1' by software in order to implement test sequences.
4: This bit can only be set by following the unlock sequence of **Section 10.4.2 "NVM Unlock Sequence"**.
5: Operations are self-timed, and the WR bit is cleared by hardware when complete.
6: Once a write operation is initiated, setting this bit to zero will have no effect.
7: Reading from EEPROM loads only NVMDATL<7:0> (Register 10-1).

REGISTER 12-34: INLVLE: PORTE INPUT LEVEL CONTROL REGISTER

U-0	U-0	U-0	U-0	R/W-1/1	U-0	U-0	U-0
—	—	—	—	INLVLE3	—	—	—
bit 7							bit 0

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

bit 7-4 **Unimplemented:** Read as ‘0’

bit 3 **INLVLE3:** PORTE Input Level Select bits
For RE3 pin,
1 = ST input used for PORT reads and interrupt-on-change
0 = TTL input used for PORT reads and interrupt-on-change

bit 2-0 **Unimplemented:** Read as ‘0’

TABLE 12-5: SUMMARY OF REGISTERS ASSOCIATED WITH PORTE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
PORTE	—	—	—	—	RE3	—	—	—	206
WPUE	—	—	—	—	WPUE3	—	—	—	206
INLVLE	—	—	—	—	INLVLE3	—	—	—	207

Legend: x = unknown, u = unchanged, – = unimplemented locations read as ‘0’. Shaded cells are not used by PORTE.

TABLE 12-6: SUMMARY OF CONFIGURATION WORD WITH PORTE

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
CONFIG2	13:8	—	—	$\overline{\text{DEBUG}}$	STVREN	PPS1WAY	ZCDDIS	BORV	—	74
	7:0	BOREN<1:0>		$\overline{\text{LPBOREN}}$	—	—	—	$\overline{\text{PWRTE}}$	MCLRE	

Legend: — = unimplemented location, read as ‘0’. Shaded cells are not used by PORTE.

REGISTER 14-1: PMD0: PMD CONTROL REGISTER 0

R/W-0/0	R/W-0/0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
SYSCMD	FVRMD	—	CRCMD	SCANMD	NVMMD	CLKRMD	IOCMD
7							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 **SYSCMD:** Disable Peripheral System Clock Network bit
See description in **Section 14.4 “System Clock Disable”**.
1 = System clock network disabled (a.k.a. FOSC)
0 = System clock network enabled

bit 6 **FVRMD:** Disable Fixed Voltage Reference (FVR) bit
1 = FVR module disabled
0 = FVR module enabled

bit 5 **Unimplemented:** Read as ‘0’

bit 4 **CRCMD:** CRC module disable bit
1 = CRC module disabled
0 = CRC module enabled

bit 3 **SCANMD:** Program Memory Scanner Module Disable bit
1 = Scanner module disabled
0 = Scanner module enabled

bit 2 **NVMMD:** NVM Module Disable bit[(1)]
1 = User memory and EEPROM reading and writing is disabled; NVMCON registers cannot be written; FSR access to these locations returns zero.
0 = NVM module enabled

bit 1 **CLKRMD:** Disable Clock Reference CLKR bit
1 = CLKR module disabled
0 = CLKR module enabled

bit 0 **IOCMD:** Disable Interrupt-on-Change bit, All Ports
1 = IOC module(s) disabled
0 = IOC module(s) enabled

Note 1: When enabling NVM, a delay of up to 1 μs may be required before accessing data.

17.0 TEMPERATURE INDICATOR MODULE

This family of devices is equipped with a temperature circuit designed to measure the operating temperature of the silicon die. The circuit's range of operating temperature falls between -40°C and +85°C. The output is a voltage that is proportional to the device temperature. The output of the temperature indicator is internally connected to the device ADC.

The circuit may be used as a temperature threshold detector or a more accurate temperature indicator, depending on the level of calibration performed. A one-point calibration allows the circuit to indicate a temperature closely surrounding that point. A two-point calibration allows the circuit to sense the entire range of temperature more accurately. Reference Application Note AN1333, "*Use and Calibration of the Internal Temperature Indicator*" (DS01333) for more details regarding the calibration process.

17.1 Circuit Operation

Figure 17-1 shows a simplified block diagram of the temperature circuit. The proportional voltage output is achieved by measuring the forward voltage drop across multiple silicon junctions.

Equation 17-1 describes the output characteristics of the temperature indicator.

EQUATION 17-1: VOUT RANGES

High Range: VOUT = VDD - 4VT

Low Range: VOUT = VDD - 2VT

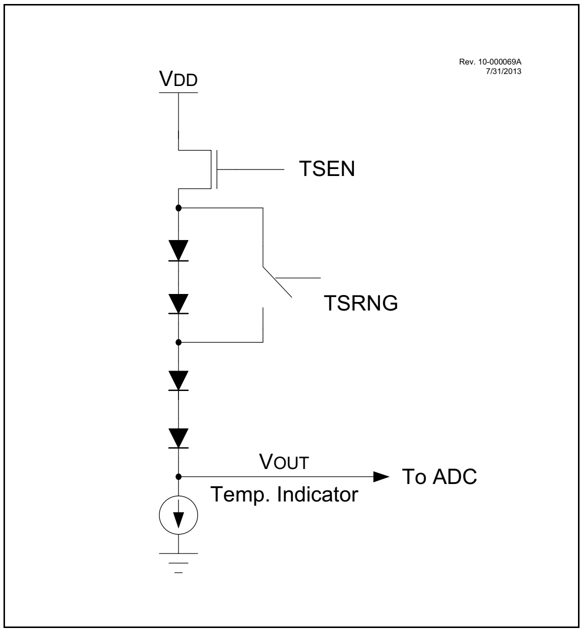
The temperature sense circuit is integrated with the Fixed Voltage Reference (FVR) module. See **16.0 "Fixed Voltage Reference (FVR)"** for more information.

The circuit is enabled by setting the TSEN bit of the FVRCON register. When disabled, the circuit draws no current.

The circuit operates in either high or low range. The high range, selected by setting the TSRNG bit of the FVRCON register, provides a wider output voltage. This provides more resolution over the temperature range, but may be less consistent from part to part. This range requires a higher bias voltage to operate and thus, a higher VDD is needed.

The low range is selected by clearing the TSRNG bit of the FVRCON register. The low range generates a lower voltage drop and thus, a lower bias voltage is needed to operate the circuit. The low range is provided for low voltage operation.

FIGURE 17-1: TEMPERATURE CIRCUIT DIAGRAM



17.2 Minimum Operating VDD

When the temperature circuit is operated in low range, the device may be operated at any operating voltage that is within specifications.

When the temperature circuit is operated in high range, the device operating voltage, VDD, must be high enough to ensure that the temperature circuit is correctly biased.

Table 17-1 shows the recommended minimum VDD vs. range setting.

TABLE 17-1: RECOMMENDED VDD VS. RANGE

Min. VDD, TSRNG = 1	Min. VDD, TSRNG = 0
3.6V	1.8V

17.3 Temperature Output

The output of the circuit is measured using the internal Analog-to-Digital Converter. A channel is reserved for the temperature circuit output. Refer to **Section 23.0 "Analog-to-Digital Converter With Computation (ADC2) Module"** for detailed information.

REGISTER 27-1: T0CON0: TIMER0 CONTROL REGISTER 0

R/W-0/0	U-0	R-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
T0EN	—	T0OUT	T016BIT	T0OUTPS<3:0>			
bit 7							bit 0

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

bit 7 **T0EN:** TMR0 Enable bit
1 = The module is enabled and operating
0 = The module is disabled and in the lowest power mode

bit 6 **Unimplemented:** Read as ‘0’

bit 5 **T0OUT:** TMR0 Output bit (read-only)
TMR0 output bit

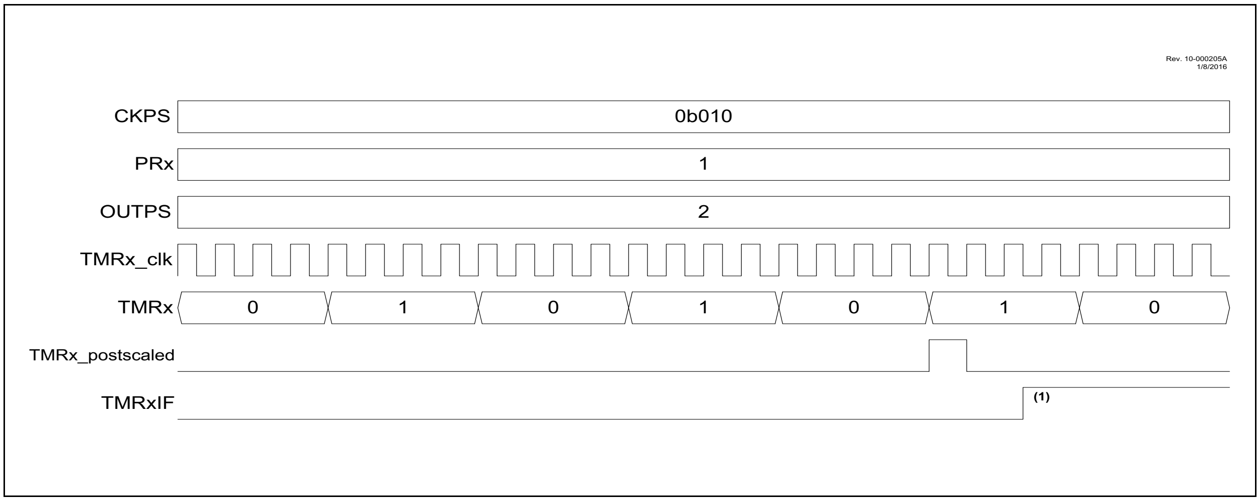
bit 4 **T016BIT:** TMR0 Operating as 16-bit Timer Select bit
1 = TMR0 is a 16-bit timer
0 = TMR0 is an 8-bit timer

bit 3-0 **T0OUTPS<3:0>:** TMR0 output postscaler (divider) select bits
1111 = 1:16 Postscaler
1110 = 1:15 Postscaler
1101 = 1:14 Postscaler
1100 = 1:13 Postscaler
1011 = 1:12 Postscaler
1010 = 1:11 Postscaler
1001 = 1:10 Postscaler
1000 = 1:9 Postscaler
0111 = 1:8 Postscaler
0110 = 1:7 Postscaler
0101 = 1:6 Postscaler
0100 = 1:5 Postscaler
0011 = 1:4 Postscaler
0010 = 1:3 Postscaler
0001 = 1:2 Postscaler
0000 = 1:1 Postscaler

29.4 Timer2 Interrupt

Timer2 can also generate a device interrupt. The interrupt is generated when the postscaler counter matches one of 16 postscale options (from 1:1 through 1:16), which are selected with the postscaler control bits, OUTPS<3:0> of the T2CON register. The interrupt is enabled by setting the TMR2IE interrupt enable bit of the PIE4 register. Interrupt timing is illustrated in Figure 29-3.

FIGURE 29-3: TIMER2 PRESCALER, POSTSCALER, AND INTERRUPT TIMING DIAGRAM



31.2.3 SPI MASTER MODE

The master can initiate the data transfer at any time because it controls the SCK line. The master determines when the slave (Processor 2, Figure 31-5) is to broadcast data by the software protocol.

In Master mode, the data is transmitted/received as soon as the SSPxBUF register is written to. If the SPI is only going to receive, the SDO output could be disabled (programmed as an input). The SSPxSR register will continue to shift in the signal present on the SDI pin at the programmed clock rate. As each byte is received, it will be loaded into the SSPxBUF register as if a normal received byte (interrupts and Status bits appropriately set).

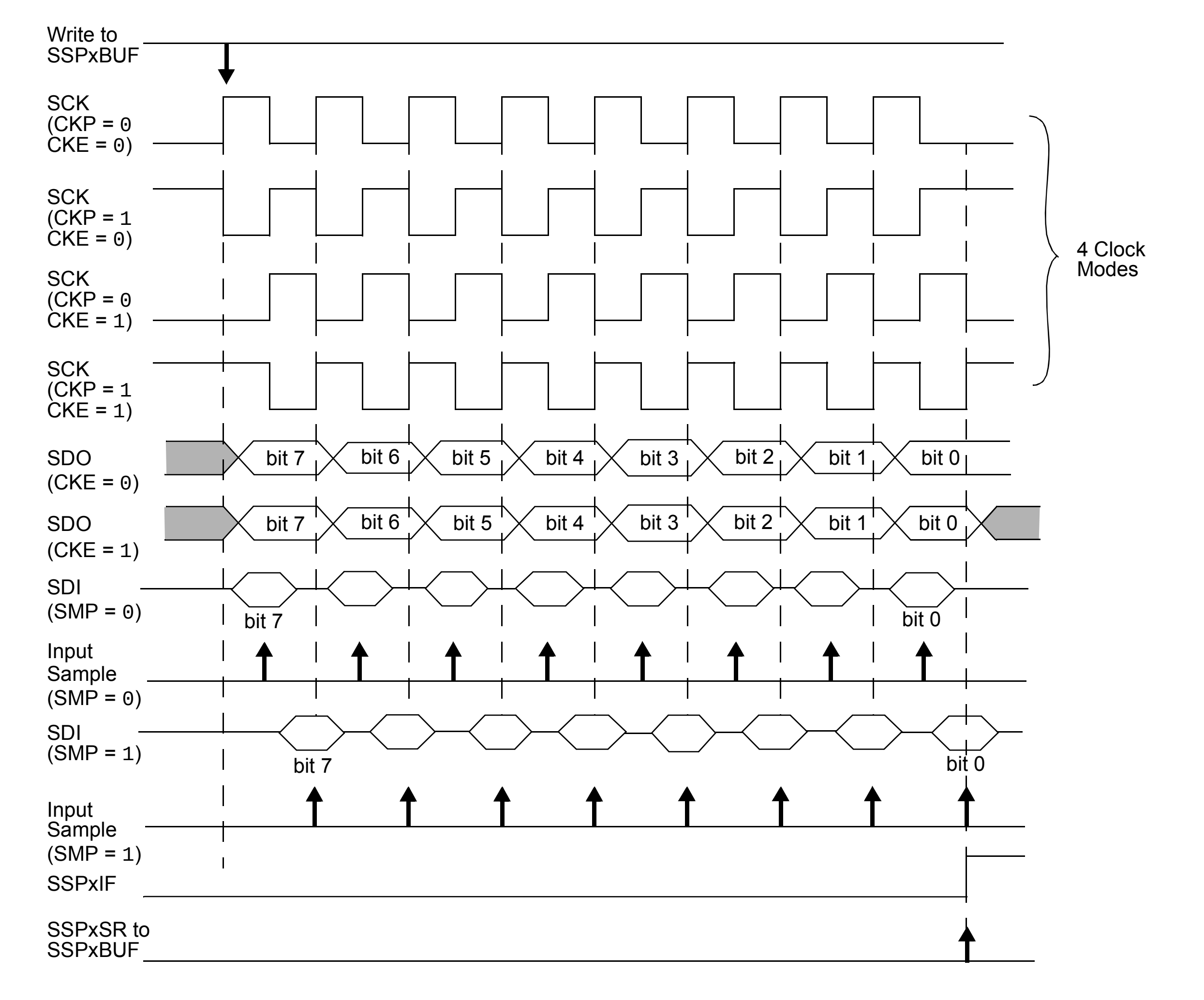
The clock polarity is selected by appropriately programming the CKP bit of the SSPxCON1 register and the CKE bit of the SSPxSTAT register. This then, would give waveforms for SPI communication as shown in Figure 31-6, Figure 31-8, Figure 31-9 and Figure 31-10, where the MSB is transmitted first. In Master mode, the SPI clock rate (bit rate) is user programmable to be one of the following:

- FOSC/4 (or TCY)
- FOSC/16 (or 4 * TCY)
- FOSC/64 (or 16 * TCY)
- Timer2 output/2
- FOSC/(4 * (SSPxADD + 1))

Figure 31-6 shows the waveforms for Master mode.

When the CKE bit is set, the SDO data is valid before there is a clock edge on SCK. The change of the input sample is shown based on the state of the SMP bit. The time when the SSPxBUF is loaded with the received data is shown.

FIGURE 31-6: SPI MODE WAVEFORM (MASTER MODE)



31.5.4 SLAVE MODE 10-BIT ADDRESS RECEPTION

This section describes a standard sequence of events for the MSSP module configured as an I^{2}C slave in 10-bit Addressing mode.

Figure 31-20 is used as a visual reference for this description.

This is a step by step process of what must be done by slave software to accomplish I^{2}C communication.

1. Bus starts Idle.
2. Master sends Start condition; S bit of SSPxSTAT is set; SSPxIF is set if interrupt on Start detect is enabled.
3. Master sends matching high address with R/$\overline{\text{W}}$ bit clear; UA bit of the SSPxSTAT register is set.
4. Slave sends $\overline{\text{ACK}}$ and SSPxIF is set.
5. Software clears the SSPxIF bit.
6. Software reads received address from SSPxBUF clearing the BF flag.
7. Slave loads low address into SSPxADD, releasing SCL.
8. Master sends matching low address byte to the slave; UA bit is set.

> **Note:** Updates to the SSPxADD register are not allowed until after the $\overline{\text{ACK}}$ sequence.

9. Slave sends $\overline{\text{ACK}}$ and SSPxIF is set.

> **Note:** If the low address does not match, SSPxIF and UA are still set so that the slave software can set SSPxADD back to the high address. BF is not set because there is no match. CKP is unaffected.

10. Slave clears SSPxIF.
11. Slave reads the received matching address from SSPxBUF clearing BF.
12. Slave loads high address into SSPxADD.
13. Master clocks a data byte to the slave and clocks out the slaves $\overline{\text{ACK}}$ on the ninth SCL pulse; SSPxIF is set.
14. If SEN bit of SSPxCON2 is set, CKP is cleared by hardware and the clock is stretched.
15. Slave clears SSPxIF.
16. Slave reads the received byte from SSPxBUF clearing BF.
17. If SEN is set the slave sets CKP to release the SCL.
18. Steps 13-17 repeat for each received byte.
19. Master sends Stop to end the transmission.

31.5.5 10-BIT ADDRESSING WITH ADDRESS OR DATA HOLD

Reception using 10-bit addressing with AHEN or DHEN set is the same as with 7-bit modes. The only difference is the need to update the SSPxADD register using the UA bit. All functionality, specifically when the CKP bit is cleared and SCL line is held low are the same. Figure 31-21 can be used as a reference of a slave in 10-bit addressing with AHEN set.

Figure 31-22 shows a standard waveform for a slave transmitter in 10-bit Addressing mode.

FIGURE 31-34: BUS COLLISION DURING START CONDITION (SCL = 0)

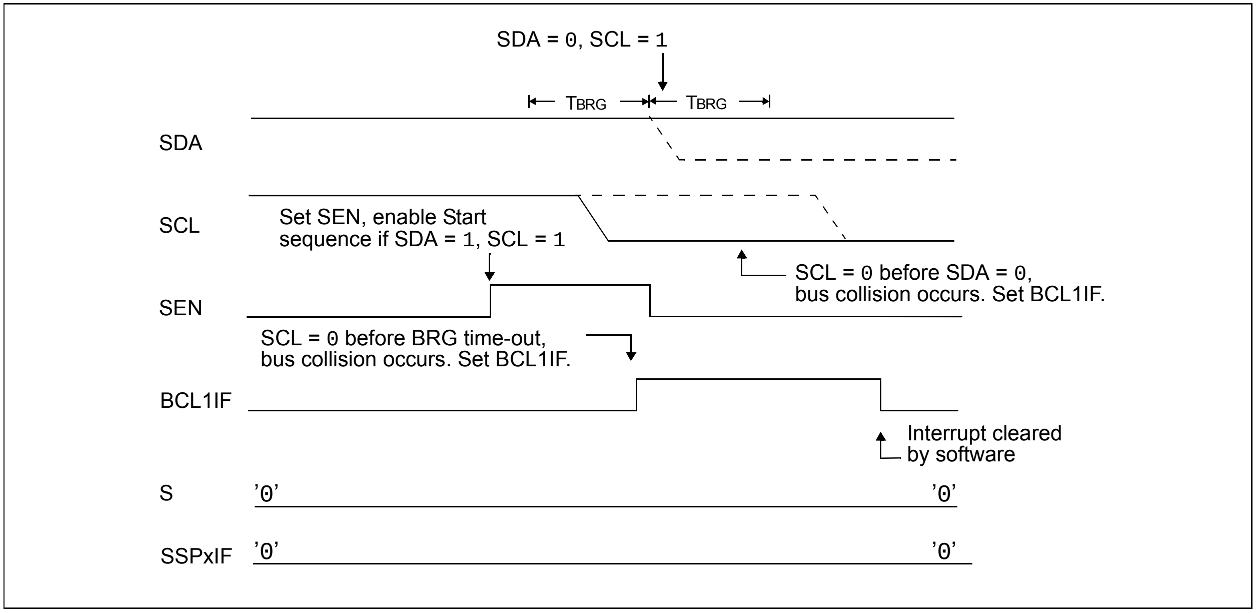

SDA = 0, SCL = 1
TBRG
TBRG
SDA
SCL
Set SEN, enable Start sequence if SDA = 1, SCL = 1
SCL = 0 before SDA = 0, bus collision occurs. Set BCL1IF.
SEN
SCL = 0 before BRG time-out, bus collision occurs. Set BCL1IF.
BCL1IF
Interrupt cleared by software
S
'0'
'0'
SSPxIF
'0'
'0'

FIGURE 31-35: BRG RESET DUE TO SDA ARBITRATION DURING START CONDITION

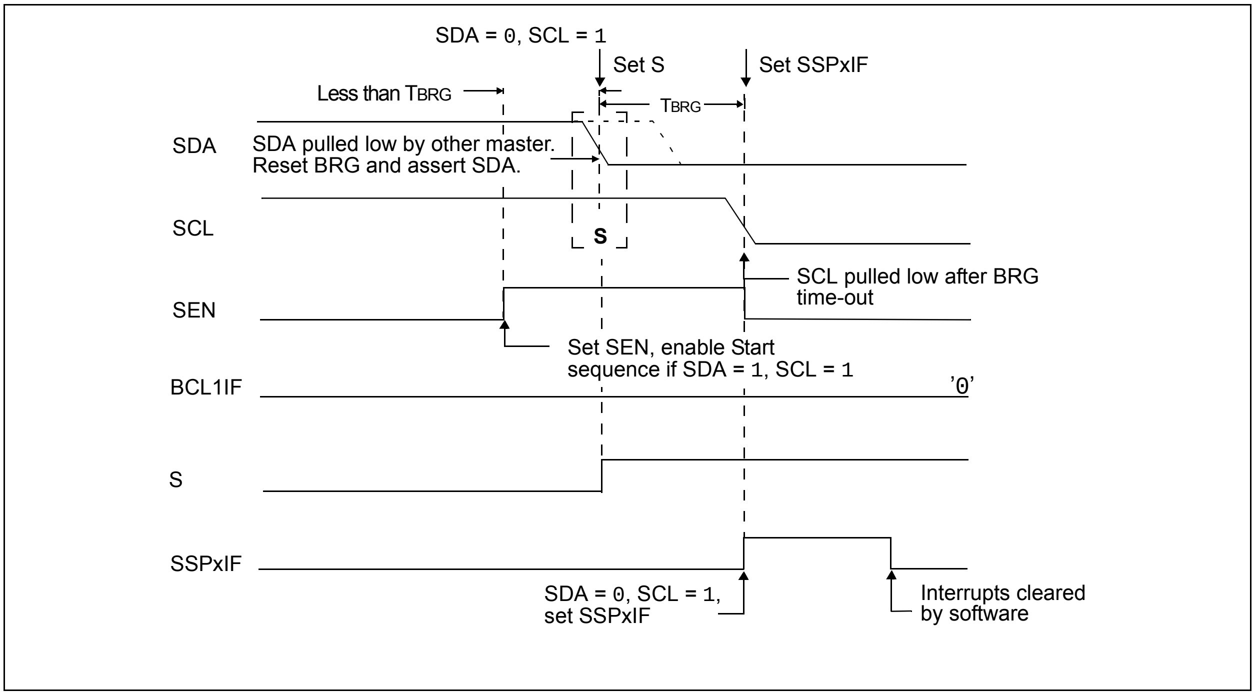

SDA = 0, SCL = 1
Set S
Set SSPxIF
Less than TBRG
TBRG
SDA
SDA pulled low by other master. Reset BRG and assert SDA.
SCL
S
SCL pulled low after BRG time-out
SEN
Set SEN, enable Start sequence if SDA = 1, SCL = 1
BCL1IF
'0'
S
SSPxIF
SDA = 0, SCL = 1, set SSPxIF
Interrupts cleared by software

FIGURE 32-1: SMT BLOCK DIAGRAM

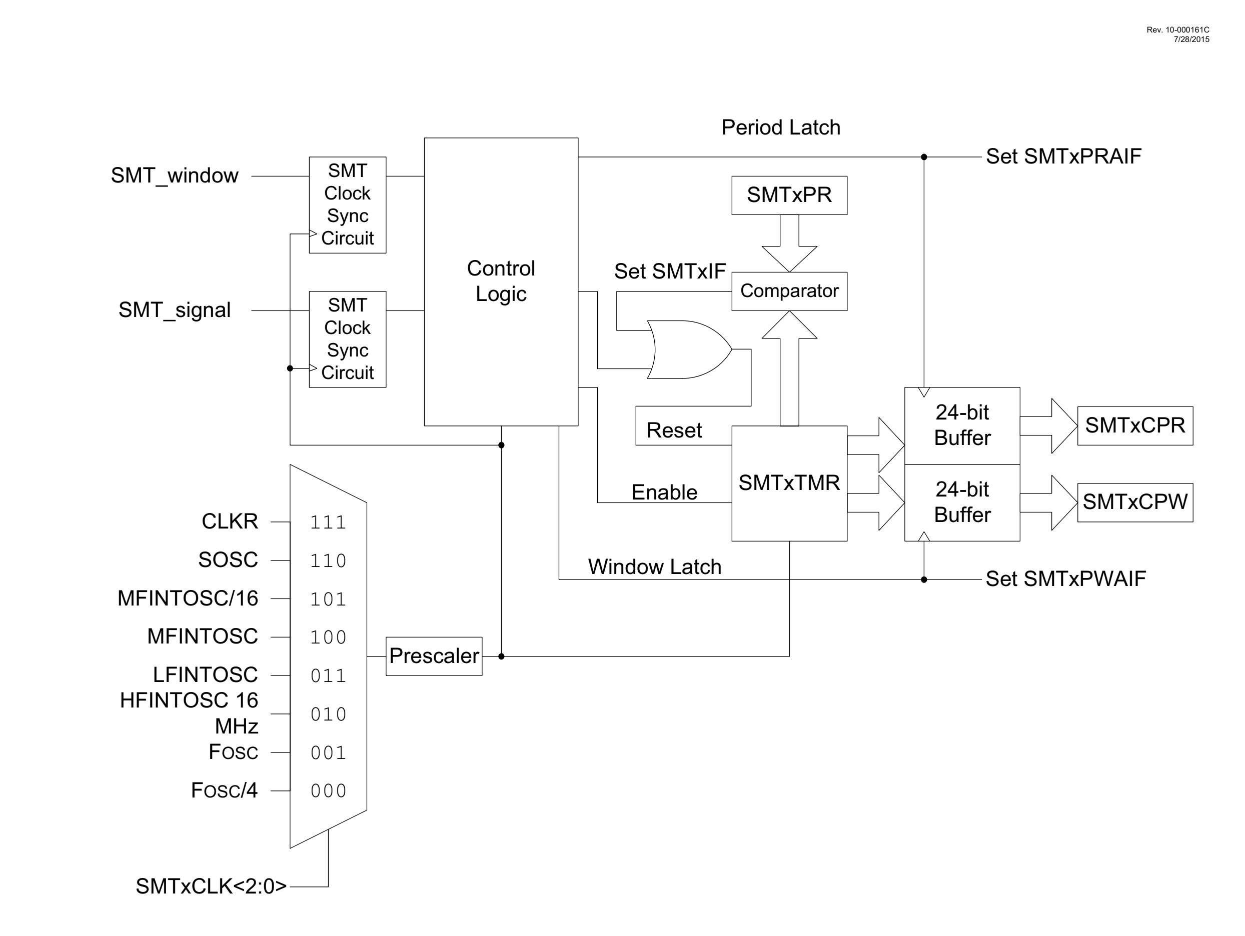
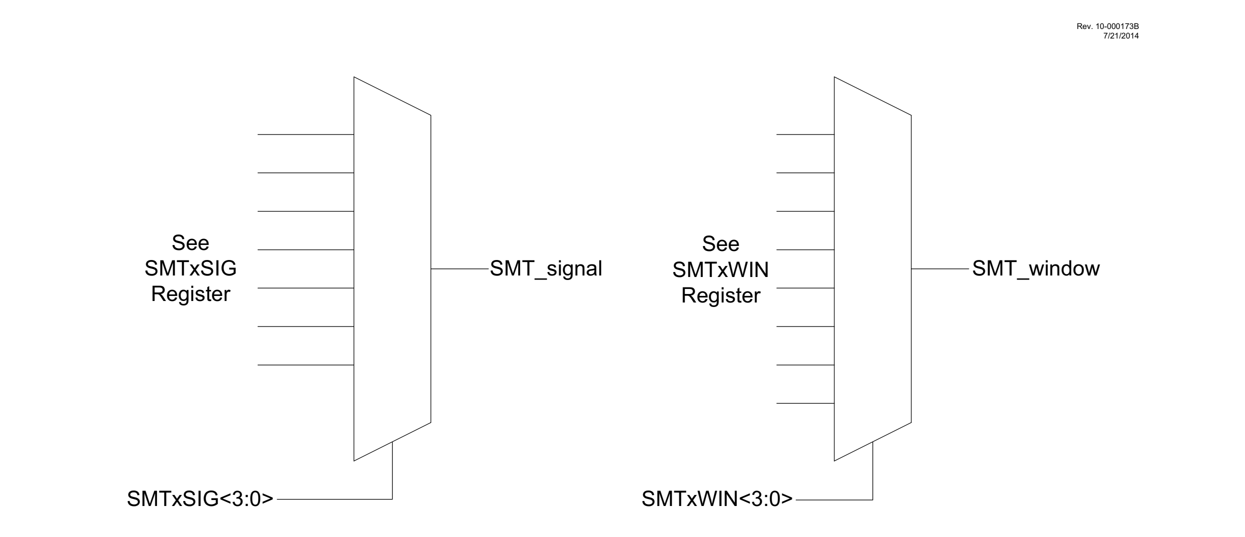


FIGURE 32-2: SMT SIGNAL AND WINDOW BLOCK DIAGRAM



32.6.8 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 32-16 and Figure 32-17.

34.0 REFERENCE CLOCK OUTPUT MODULE

The Reference Clock Output module provides the ability to send a clock signal to the clock reference output pin (CLKR). The Reference Clock Output can also be used as a signal for other peripherals, such as the Data Signal Modulator (DSM).

The Reference Clock Output module has the following features:

- Selectable input clock
- Programmable clock divider
- Selectable duty cycle

34.1 CLOCK SOURCE

The Reference Clock Output module has a selectable clock source. The CLKRCLK register (Register 34-2) controls which input is used.

34.1.1 CLOCK SYNCHRONIZATION

Once the reference clock enable (CLKREN) is set, the module is ensured to be glitch-free at start-up.

When the Reference Clock Output is disabled, the output signal will be disabled immediately.

Clock dividers and clock duty cycles can be changed while the module is enabled, but glitches may occur on the output. To avoid possible glitches, clock dividers and clock duty cycles should be changed only when the CLKREN is clear.

34.2 PROGRAMMABLE CLOCK DIVIDER

The module takes the system clock input and divides it based on the value of the CLKRDIV<2:0> bits of the CLKRCON register (Register 34-1).

The following configurations can be made based on the CLKRDIV<2:0> bits:

- Base FOSC value
- FOSC divided by 2
- FOSC divided by 4
- FOSC divided by 8
- FOSC divided by 16
- FOSC divided by 32
- FOSC divided by 64
- FOSC divided by 128

The clock divider values can be changed while the module is enabled; however, in order to prevent glitches on the output, the CLKRDIV<2:0> bits should only be changed when the module is disabled (CLKREN = 0).

34.3 SELECTABLE DUTY CYCLE

The CLKRDC<1:0> bits of the CLKRCON register can be used to modify the duty cycle of the output clock. A duty cycle of 25%, 50%, or 75% can be selected for all clock rates, with the exception of the undivided base FOSC value.

The duty cycle can be changed while the module is enabled; however, in order to prevent glitches on the output, the CLKRDC<1:0> bits should only be changed when the module is disabled (CLKREN = 0).

Note:	The CLKRDC1 bit is reset to ‘1’. This makes the default duty cycle 50% and not 0%.

34.4 OPERATION IN SLEEP MODE

The Reference Clock Output module clock is based on the system clock. When the device goes to Sleep, the module outputs will remain in their current state. This will have a direct effect on peripherals using the Reference Clock Output as an input signal.

CALL	**Call Subroutine**
Syntax:	[*label*] CALL k
Operands:	0 ≤ k ≤ 2047
Operation:	(PC)+ 1→ TOS, k → PC<10:0>, (PCLATH<6:3>) → PC<14:11>
Status Affected:	None
Description:	Call Subroutine. First, return address (PC + 1) is pushed onto the stack. The 11-bit immediate address is loaded into PC bits <10:0>. The upper bits of the PC are loaded from PCLATH. `CALL` is a 2-cycle instruction.

CALLW	**Subroutine Call With W**
Syntax:	[*label*] CALLW
Operands:	None
Operation:	(PC) +1 → TOS, (W) → PC<7:0>, (PCLATH<6:0>) → PC<14:8>
Status Affected:	None
Description:	Subroutine call with W. First, the return address (PC + 1) is pushed onto the return stack. Then, the contents of W is loaded into PC<7:0>, and the contents of PCLATH into PC<14:8>. `CALLW` is a 2-cycle instruction.

CLRF	**Clear f**
Syntax:	[*label*] CLRF f
Operands:	0 ≤ f ≤ 127
Operation:	00h → (f) `1` → Z
Status Affected:	Z
Description:	The contents of register ‘f’ are cleared and the Z bit is set.

CLRW	**Clear W**
Syntax:	[*label*] CLRW
Operands:	None
Operation:	00h → (W) `1` → Z
Status Affected:	Z
Description:	W register is cleared. Zero bit (Z) is set.

CLRWDT	**Clear Watchdog Timer**
Syntax:	[*label*] CLRWDT
Operands:	None
Operation:	00h → WDT `0` → WDT prescaler, `1` → $\overline{TO}$ `1` → $\overline{PD}$
Status Affected:	$\overline{TO}$, $\overline{PD}$
Description:	`CLRWDT` instruction resets the Watchdog Timer. It also resets the prescaler of the WDT. Status bits $\overline{TO}$ and $\overline{PD}$ are set.

COMF	**Complement f**
Syntax:	[*label*] COMF f,d
Operands:	0 ≤ f ≤ 127 d ∈ [`0`,`1`]
Operation:	($\bar{f}$) → (destination)
Status Affected:	Z
Description:	The contents of register ‘f’ are complemented. If ‘d’ is ‘`0`’, the result is stored in W. If ‘d’ is ‘`1`’, the result is stored back in register ‘f’.

DECF	**Decrement f**
Syntax:	[*label*] DECF f,d
Operands:	0 ≤ f ≤ 127 d ∈ [`0`,`1`]
Operation:	(f) - 1 → (destination)
Status Affected:	Z
Description:	Decrement register ‘f’. If ‘d’ is ‘`0`’, the result is stored in the W register. If ‘d’ is ‘`1`’, the result is stored back in register ‘f’.

FIGURE 37-7: CLKOUT AND I/O TIMING

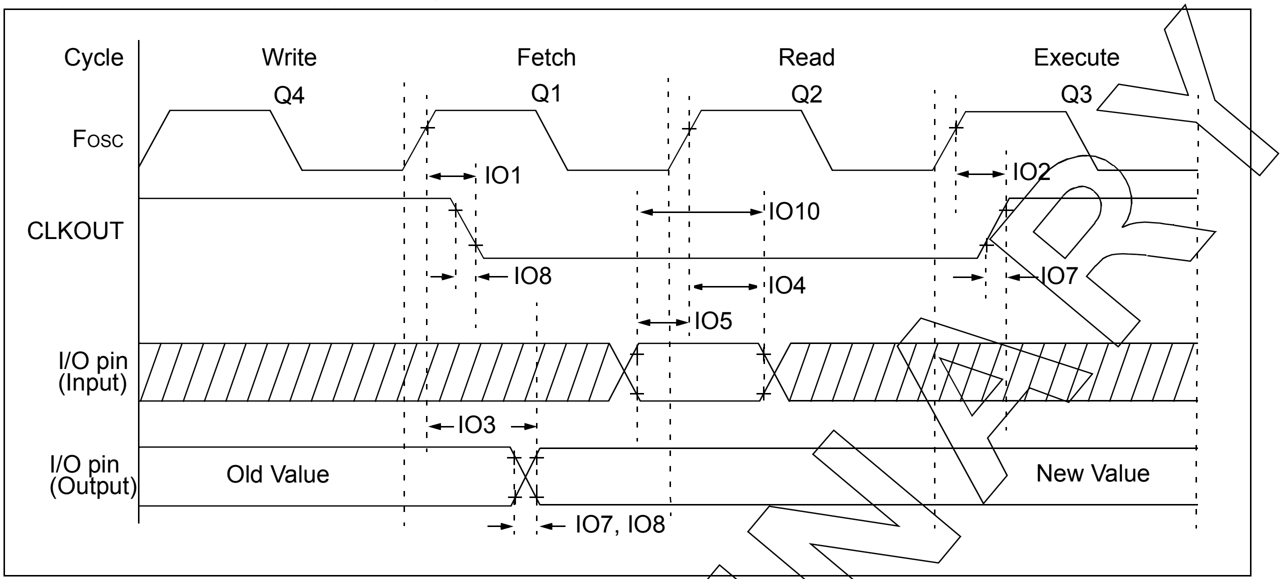


TABLE 37-10: I/O AND CLKOUT TIMING SPECIFICATIONS

Standard Operating Conditions (unless otherwise stated)							
Param. No.	**Sym.**	**Characteristic**	**Min.**	**Typ†**	**Max.**	**Units**	**Conditions**
IO1*	$T_{CLKOUTH}$	CLKOUT rising edge delay (rising edge Fosc (Q1 cycle) to falling edge CLKOUT	—	—	70	ns	
IO2*	$T_{CLKOUTL}$	CLKOUT falling edge delay (rising edge Fosc (Q3 cycle) to rising edge CLKOUT	—	—	72	ns	
IO3*	T_{IO_VALID}	Port output valid time (rising edge Fosc (Q1 cycle) to port valid)	—	50	70	ns	
IO4*	T_{IO_SETUP}	Port input setup time (Setup time before rising edge Fosc – Q2 cycle)	20	—	—	ns	
IO5*	T_{IO_HOLD}	Port input hold time (Hold time after rising edge Fosc – Q2 cycle)	50	—	—	ns	
IO6*	T_{IOR_SLREN}	Port I/O rise time, slew rate enabled	—	25	—	ns	VDD = 3.0V
IO7*	T_{IOR_SLRDIS}	Port I/O rise time, slew rate disabled	—	5	—	ns	VDD = 3.0V
IO8*	T_{IOF_SLREN}	Port I/O fall time, slew rate enabled	—	25	—	ns	VDD = 3.0V
IO9*	T_{IOF_SLRDIS}	Port I/O fall time, slew rate disabled	—	5	—	ns	VDD = 3.0V
IO10*	T_{INT}	INT pin high or low time to trigger an interrupt	25	—	—	ns	
IO11*	T_{IOC}	Interrupt-on-Change minimum high or low time to trigger interrupt	25	—	—	ns	

*These parameters are characterized but not tested.