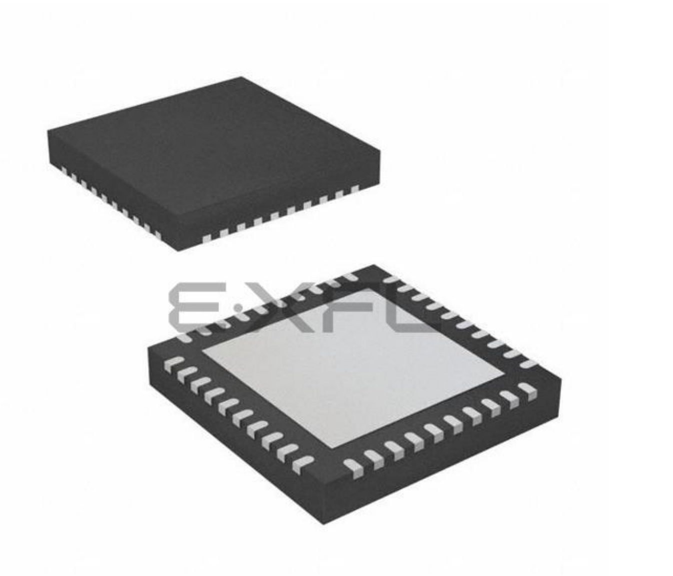
Microchip Technology - PIC18LF43K22-I/MV Datasheet



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	64MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, HLVD, POR, PWM, WDT
Number of I/O	35
Program Memory Size	8KB (4K x 16)
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 30x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	40-UFQFN Exposed Pad
Supplier Device Package	40-UQFN (5x5)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18lf43k22-i-mv

TABLE 1-2: PIC18(L)F2XK22 PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Number		Pin Name	Pin Type	Buffer Type	Description
PDIP, SOIC	QFN, UQFN				
13	10	RC2/CTPLS/P1A/CCP1/T5CKI/AN14			
		RC2	I/O	ST	Digital I/O.
		CTPLS	O	—	CTMU pulse generator output.
		P1A	O	CMOS	Enhanced CCP1 PWM output.
		CCP1	I/O	ST	Capture 1 input/Compare 1 output/PWM 1 output.
		T5CKI	I	ST	Timer5 clock input.
		AN14	I	Analog	Analog input 14.
14	11	RC3/SCK1/SCL1/AN15			
		RC3	I/O	ST	Digital I/O.
		SCK1	I/O	ST	Synchronous serial clock input/output for SPI mode (MSSP).
		SCL1	I/O	ST	Synchronous serial clock input/output for I^{2}C mode (MSSP).
		AN15	I	Analog	Analog input 15.
15	12	RC4/SDI1/SDA1/AN16			
		RC4	I/O	ST	Digital I/O.
		SDI1	I	ST	SPI data in (MSSP).
		SDA1	I/O	ST	I^{2}C data I/O (MSSP).
		AN16	I	Analog	Analog input 16.
16	13	RC5/SDO1/AN17			
		RC5	I/O	ST	Digital I/O.
		SDO1	O	—	SPI data out (MSSP).
		AN17	I	Analog	Analog input 17.
17	14	RC6/P3A/CCP3/TX1/CK1/AN18			
		RC6	I/O	ST	Digital I/O.
		P3A(2)	O	CMOS	Enhanced CCP3 PWM output.
		CCP3(2)	I/O	ST	Capture 3 input/Compare 3 output/PWM 3 output.
		TX1	O	—	EUSART asynchronous transmit.
		CK1	I/O	ST	EUSART synchronous clock (see related RXx/DTx).
		AN18	I	Analog	Analog input 18.
18	15	RC7/P3B/RX1/DT1/AN19			
		RC7	I/O	ST	Digital I/O.
		P3B	O	CMOS	Enhanced CCP3 PWM output.
		RX1	I	ST	EUSART asynchronous receive.
		DT1	I/O	ST	EUSART synchronous data (see related TXx/CKx).
		AN19	I	Analog	Analog input 19.
1	26	RE3/VPP/$\overline{\text{MCLR}}$			
		RE3	I	ST	Digital input.
		VPP	P		Programming voltage input.
		$\overline{\text{MCLR}}$	I	ST	Active-Low Master Clear (device Reset) input.

Legend: TTL = TTL compatible input CMOS = CMOS compatible input or output; ST = Schmitt Trigger input with CMOS levels; I = Input; O = Output; P = Power.

Note 1: Default pin assignment for P2B, T3CKI, CCP3 and CCP2 when Configuration bits PB2MX, T3CMX, CCP3MX and CCP2MX are set.

2: Alternate pin assignment for P2B, T3CKI, CCP3 and CCP2 when Configuration bits PB2MX, T3CMX, CCP3MX and CCP2MX are clear.

2.6.1.1 OSCTUNE Register

The HFINTOSC/MFINTOSC oscillator circuits are factory calibrated but can be adjusted in software by writing to the TUN<5:0> bits of the OSCTUNE register (Register 2-3).

The default value of the TUN<5:0> is '`000000`'. The value is a 6-bit two's complement number.

When the OSCTUNE register is modified, the HFINTOSC/MFINTOSC frequency will begin shifting to the new frequency. Code execution continues during this shift. There is no indication that the shift has occurred.

The TUN<5:0> bits in OSCTUNE do not affect the LFINTOSC frequency. Operation of features that depend on the LFINTOSC clock source frequency, such as the Power-up Timer (PWRT), Watchdog Timer (WDT), Fail-Safe Clock Monitor (FSCM) and peripherals, are *not* affected by the change in frequency.

The OSCTUNE register also implements the INTSRC and PLLEN bits, which control certain features of the internal oscillator block.

The INTSRC bit allows users to select which internal oscillator provides the clock source when the 31.25 kHz frequency option is selected. This is covered in greater detail in **Section 2.2.3 "Low Frequency Selection"**.

The PLLEN bit controls the operation of the frequency multiplier, PLL, for all primary external clock sources and internal oscillator modes. However, the PLL is intended for operation with clock sources between 4 MHz and 16 MHz. For more details about the function of the PLLEN bit, see **Section 2.8.2 "PLL in HFINTOSC Modes"**

2.7 Register Definitions: Oscillator Tuning

REGISTER 2-3: OSCTUNE: OSCILLATOR TUNING REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
INTSRC	PLLEN(1)	TUN<5:0>					
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 **INTSRC:** Internal Oscillator Low-Frequency Source Select bit
`1` = 31.25 kHz device clock derived from the MFINTOSC or HFINTOSC source
`0` = 31.25 kHz device clock derived directly from LFINTOSC internal oscillator

bit 6 **PLLEN:** Frequency Multiplier 4xPLL for HFINTOSC Enable bit(1)
`1` = PLL enabled
`0` = PLL disabled

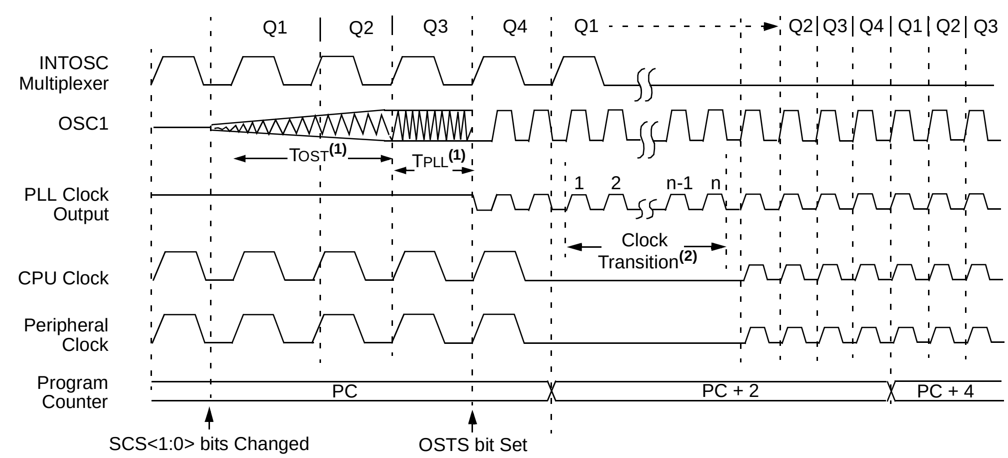
bit 5-0 **TUN<5:0>:** Frequency Tuning bits – use to adjust MFINTOSC and HFINTOSC frequencies
`011111` = Maximum frequency
`011110` =
•••
`000001` =
`000000` = Oscillator module (HFINTOSC and MFINTOSC) are running at the factory calibrated frequency.
`111111` =
•••
`100000` = Minimum frequency

Note 1: The PLLEN bit is active for all the primary clock sources (internal or external) and is designed to operate with clock frequencies between 4 MHz and 16 MHz.

TABLE 3-2: INTERNAL OSCILLATOR FREQUENCY STABILITY BITS

IRCF<2:0>	INTSRC	MFIOSEL	Selected Oscillator	Selected Oscillator Stable when:
000	0	x	LFINTOSC	LFIOFS = 1
000	1	0	HFINTOSC	HFIOFS = 1
000	1	1	MFINTOSC	MFIOFS = 1
010 or 001	x	0	HFINTOSC	HFIOFS = 1
010 or 001	x	1	MFINTOSC	MFIOFS = 1
011 – 111	x	x	HFINTOSC	HFIOFS = 1

FIGURE 3-3: TRANSITION TIMING FROM RC_RUN MODE TO PRI_RUN MODE



Note 1: T_{OST} = 1024 T_{OSC}; T_{PLL} = 2 ms (approx). These intervals are not shown to scale.

2: Clock transition typically occurs within 2-4 T_{OSC}.

3.5 Exiting Idle and Sleep Modes

An exit from Sleep mode or any of the Idle modes is triggered by any one of the following:

- an interrupt
- a Reset
- a Watchdog Time-out

This section discusses the triggers that cause exits from power-managed modes. The clocking subsystem actions are discussed in each of the power-managed modes (see **Section 3.2 "Run Modes", Section 3.3 "Sleep Mode"** and **Section 3.4 "Idle Modes"**).

3.5.1 EXIT BY INTERRUPT

Any of the available interrupt sources can cause the device to exit from an Idle mode or the Sleep mode to a Run mode. To enable this functionality, an interrupt source must be enabled by setting its enable bit in one of the INTCON or PIE registers. The exit sequence is initiated when the corresponding interrupt flag bit is set.

The instruction immediately following the `SLEEP` instruction is executed on all exits by interrupt from Idle or Sleep modes. Code execution then branches to the interrupt vector if the GIE/GIEH bit of the INTCON register is set, otherwise code execution continues without branching (see **Section 9.0 "Interrupts"**).

A fixed delay of interval T_{CSD} following the wake event is required when leaving Sleep and Idle modes. This delay is required for the CPU to prepare for execution. Instruction execution resumes on the first clock cycle following this delay.

3.5.2 EXIT BY WDT TIME-OUT

A WDT time-out will cause different actions depending on which power-managed mode the device is in when the time-out occurs.

If the device is not executing code (all Idle modes and Sleep mode), the time-out will result in an exit from the power-managed mode (see **Section 3.2 "Run Modes"** and **Section 3.3 "Sleep Mode"**). If the device is executing code (all Run modes), the time-out will result in a WDT Reset (see **Section 24.3 "Watchdog Timer (WDT)"**).

The WDT timer and postscaler are cleared by any one of the following:

- executing a `SLEEP` instruction
- executing a `CLRWDT` instruction
- the loss of the currently selected clock source when the Fail-Safe Clock Monitor is enabled
- modifying the IRCF bits in the OSCCON register when the internal oscillator block is the device clock source

3.5.3 EXIT BY RESET

Exiting Sleep and Idle modes by Reset causes code execution to restart at address '0'. See **Section 4.0 "Reset"** for more details.

The exit delay time from Reset to the start of code execution depends on both the clock sources before and after the wake-up and the type of oscillator.

3.5.4 EXIT WITHOUT AN OSCILLATOR START-UP DELAY

Certain exits from power-managed modes do not invoke the OST at all. There are two cases:

- PRI_IDLE mode, where the primary clock source is not stopped and
- the primary clock source is not any of the LP, XT, HS or HSPLL modes.

In these instances, the primary clock source either does not require an oscillator start-up delay since it is already running (PRI_IDLE), or normally does not require an oscillator start-up delay (RC, EC, INTOSC, and INTOSCIO modes). However, a fixed delay of interval T_{CSD} following the wake event is still required when leaving Sleep and Idle modes to allow the CPU to prepare for execution. Instruction execution resumes on the first clock cycle following this delay.

5.0 MEMORY ORGANIZATION

There are three types of memory in PIC18 Enhanced microcontroller devices:

- Program Memory
- Data RAM
- Data EEPROM

As Harvard architecture devices, the data and program memories use separate buses; this allows for concurrent access of the two memory spaces. The data EEPROM, for practical purposes, can be regarded as a peripheral device, since it is addressed and accessed through a set of control registers.

Additional detailed information on the operation of the Flash program memory is provided in **Section 6.0 “Flash Program Memory”**. Data EEPROM is discussed separately in **Section 7.0 “Data EEPROM Memory”**.

5.1 Program Memory Organization

PIC18 microcontrollers implement a 21-bit program counter, which is capable of addressing a 2-Mbyte program memory space. Accessing a location between the upper boundary of the physically implemented memory and the 2-Mbyte address will return all ‘`0`’s (a `NOP` instruction).

This family of devices contain the following:

- PIC18(L)F23K22, PIC18(L)F43K22: 8 Kbytes of Flash Memory, up to 4,096 single-word instructions
- PIC18(L)F24K22, PIC18(L)F44K22: 16 Kbytes of Flash Memory, up to 8,192 single-word instructions
- PIC18(L)F25K22, PIC18(L)F45K22: 32 Kbytes of Flash Memory, up to 16,384 single-word instructions
- PIC18(L)F26K22, PIC18(L)F46K22: 64 Kbytes of Flash Memory, up to 37,768 single-word instructions

PIC18 devices have two interrupt vectors. The Reset vector address is at 0000h and the interrupt vector addresses are at 0008h and 0018h.

The program memory map for PIC18(L)F2X/4XK22 devices is shown in Figure 5-1. Memory block details are shown in Figure 20-2.

5.3.3 INSTRUCTIONS IN PROGRAM MEMORY

The program memory is addressed in bytes. Instructions are stored as either two bytes or four bytes in program memory. The Least Significant Byte of an instruction word is always stored in a program memory location with an even address (LSb = `0`). To maintain alignment with instruction boundaries, the PC increments in steps of two and the LSb will always read '`0`' (see **Section 5.1.1 "Program Counter"**).

Figure 5-4 shows an example of how instruction words are stored in the program memory.

The `CALL` and `GOTO` instructions have the absolute program memory address embedded into the instruction. Since instructions are always stored on word boundaries, the data contained in the instruction is a word address. The word address is written to PC<20:1>, which accesses the desired byte address in program memory. Instruction #2 in Figure 5-4 shows how the instruction `GOTO 0006h` is encoded in the program memory. Program branch instructions, which encode a relative address offset, operate in the same manner. The offset value stored in a branch instruction represents the number of single-word instructions that the PC will be offset by. **Section 25.0 "Instruction Set Summary"** provides further details of the instruction set.

FIGURE 5-4: INSTRUCTIONS IN PROGRAM MEMORY

			LSB = 1	LSB = 0	Word Address ↓
Program Memory Byte Locations →					000000h
					000002h
					000004h
					000006h
Instruction 1:	`MOVLW`	`055h`	0Fh	55h	000008h
Instruction 2:	`GOTO`	`0006h`	EFh	03h	00000Ah
			F0h	00h	00000Ch
Instruction 3:	`MOVFF`	`123h, 456h`	C1h	23h	00000Eh
			F4h	56h	000010h
					000012h
					000014h

5.3.4 TWO-WORD INSTRUCTIONS

The standard PIC18 instruction set has four two-word instructions: `CALL`, `MOVFF`, `GOTO` and `LSFR`. In all cases, the second word of the instruction always has '`1111`' as its four Most Significant bits; the other 12 bits are literal data, usually a data memory address.

The use of '`1111`' in the 4 MSbs of an instruction specifies a special form of `NOP`. If the instruction is executed in proper sequence – immediately after the first word – the data in the second word is accessed and used by the instruction sequence.

If the first word is skipped for some reason and the second word is executed by itself, a `NOP` is executed instead. This is necessary for cases when the two-word instruction is preceded by a conditional instruction that changes the PC. Example 5-4 shows how this works.

> **Note:** See **Section 5.8 "PIC18 Instruction Execution and the Extended Instruction Set"** for information on two-word instructions in the extended instruction set.

EXAMPLE 5-4: TWO-WORD INSTRUCTIONS

CASE 1:

Object Code	Source Code
`0110 0110 0000 0000`	`TSTFSZ REG1 ; is RAM location 0?`
`1100 0001 0010 0011`	`MOVFF REG1, REG2 ; No, skip this word`
`1111 0100 0101 0110`	`; Execute this word as a NOP`
`0010 0100 0000 0000`	`ADDWF REG3 ; continue code`

CASE 2:

Object Code	Source Code
`0110 0110 0000 0000`	`TSTFSZ REG1 ; is RAM location 0?`
`1100 0001 0010 0011`	`MOVFF REG1, REG2 ; Yes, execute this word`
`1111 0100 0101 0110`	`; 2nd word of instruction`
`0010 0100 0000 0000`	`ADDWF REG3 ; continue code`

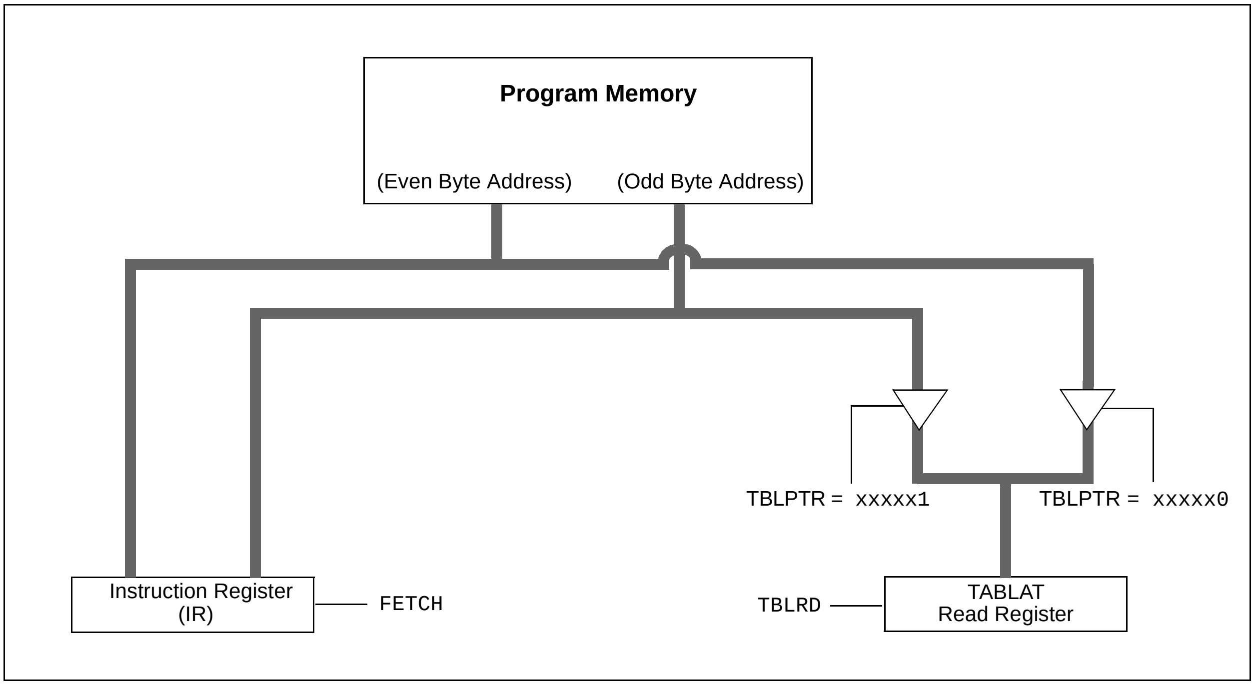
6.4 Reading the Flash Program Memory

The TBLRD instruction retrieves data from program memory and places it into data RAM. Table reads from program memory are performed one byte at a time.

TBLPTR points to a byte address in program space. Executing TBLRD places the byte pointed to into TABLAT. In addition, TBLPTR can be modified automatically for the next table read operation.

The internal program memory is typically organized by words. The Least Significant bit of the address selects between the high and low bytes of the word. Figure 6-4 shows the interface between the internal program memory and the TABLAT.

FIGURE 6-4: READS FROM FLASH PROGRAM MEMORY



EXAMPLE 6-1: READING A FLASH PROGRAM MEMORY WORD

```
          MOVLW     CODE_ADDR_UPPER        ; Load TBLPTR with the base
          MOVWF     TBLPTRU                ; address of the word
          MOVLW     CODE_ADDR_HIGH
          MOVWF     TBLPTRH
          MOVLW     CODE_ADDR_LOW
          MOVWF     TBLPTRL
READ_WORD
          TBLRD*+                          ; read into TABLAT and increment
          MOVF      TABLAT, W              ; get data
          MOVWF     WORD_EVEN
          TBLRD*+                          ; read into TABLAT and increment
          MOVFW     TABLAT, W              ; get data
          MOVF      WORD_ODD
```

REGISTER 9-10: PIE2: PERIPHERAL INTERRUPT ENABLE (FLAG) REGISTER 2

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
OSCFIE	C1IE	C2IE	EEIE	BCL1IE	HLVDIE	TMR3IE	CCP2IE
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 **OSCFIE:** Oscillator Fail Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 6 **C1IE:** Comparator C1 Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 5 **C2IE:** Comparator C2 Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 4 **EEIE:** Data EEPROM/Flash Write Operation Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 3 **BCL1IE:** MSSP1 Bus Collision Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 2 **HLVDIE:** Low-Voltage Detect Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 1 **TMR3IE:** TMR3 Overflow Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 0 **CCP2IE:** CCP2 Interrupt Enable bit
1 = Enabled
0 = Disabled

TABLE 10-11: PORTD I/O SUMMARY (CONTINUED)

Pin Name	Function	TRIS Setting	ANSEL setting	Pin Type	Buffer Type	Description
RD6/P1C/TX2/CK2/AN26	RD6	0	0	O	DIG	LATD<6> data output; not affected by analog input.
		1	0	I	ST	PORTD<6> data input; disabled when analog input enabled.
	P1C	0	0	O	DIG	Enhanced CCP1 PWM output 3.
	TX2	1	0	O	DIG	EUSART asynchronous transmit data output.
	CK2	1	0	O	DIG	EUSART synchronous serial clock output.
		1	0	I	ST	EUSART synchronous serial clock input.
	AN26	1	1	I	AN	Analog input 26.
RD7/P1D/RX2/DT2/AN27	RD7	0	0	O	DIG	LATD<7> data output; not affected by analog input.
		1	0	I	ST	PORTD<7> data input; disabled when analog input enabled.
	P1D	0	0	O	DIG	Enhanced CCP1 PWM output 4.
	RX2	1	0	I	ST	EUSART asynchronous receive data in.
	DT2	1	0	O	DIG	EUSART synchronous serial data output.
		1	0	I	ST	EUSART synchronous serial data input.
	AN27	1	1	I	AN	Analog input 27.

Legend: AN = Analog input or output; TTL = TTL compatible input; HV = High Voltage; OD = Open Drain; XTAL = Crystal; CMOS = CMOS compatible input or output; ST = Schmitt Trigger input with CMOS levels; I^2C = Schmitt Trigger input with I^2C.

Note 1: Default pin assignment for P2B, T3CKI, CCP3 and CCP2 when Configuration bits PB2MX, T3CMX, CCP3MX and CCP2MX are set.

FIGURE 15-9: SPI MODE WAVEFORM (SLAVE MODE WITH CKE = 0)

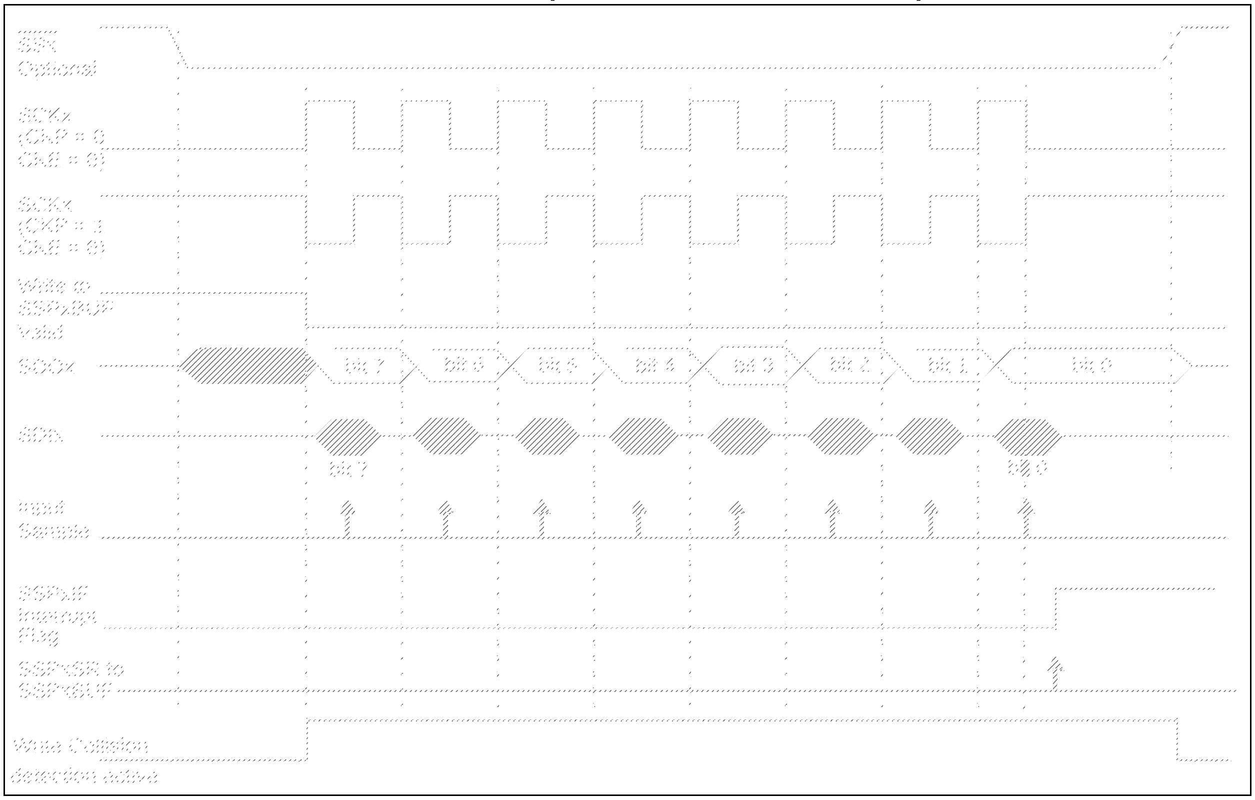


FIGURE 15-10: SPI MODE WAVEFORM (SLAVE MODE WITH CKE = 1)

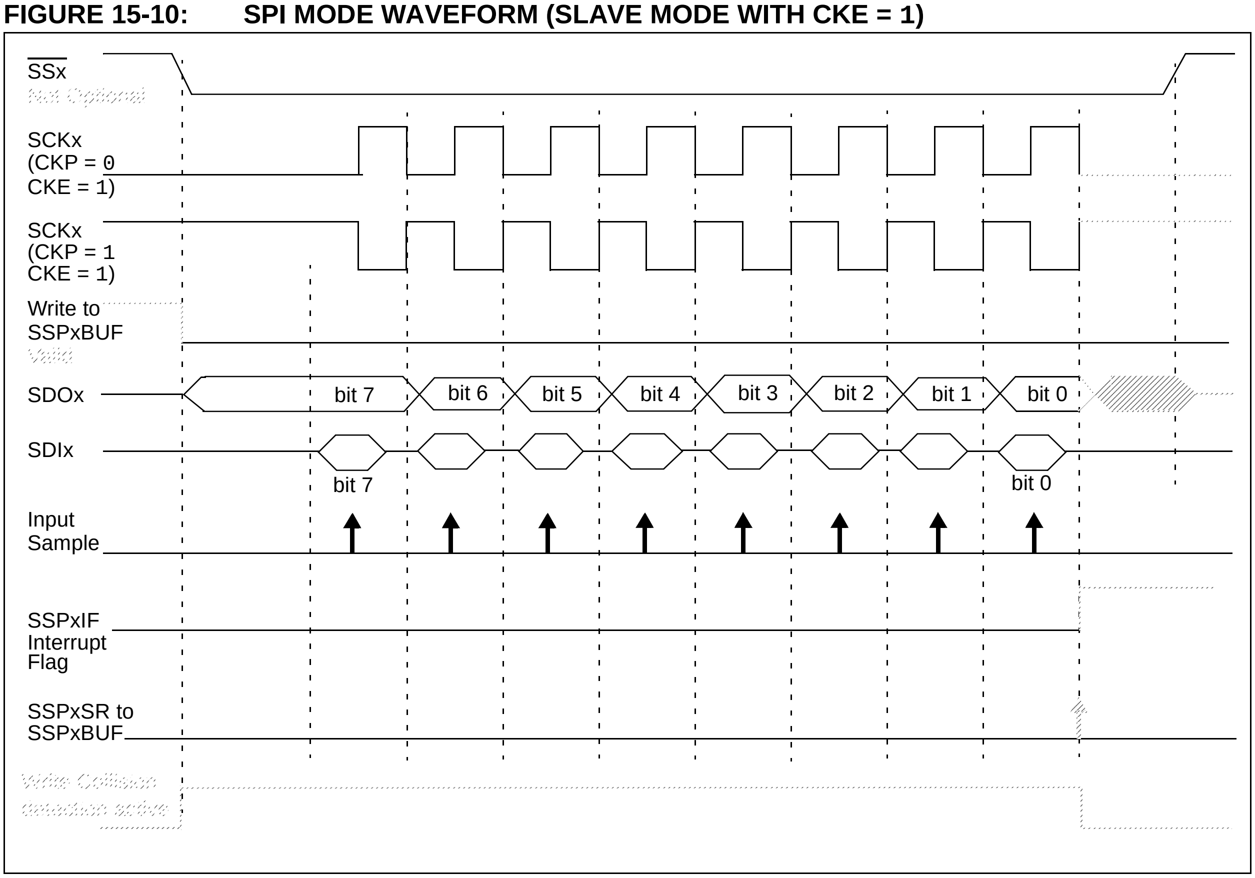
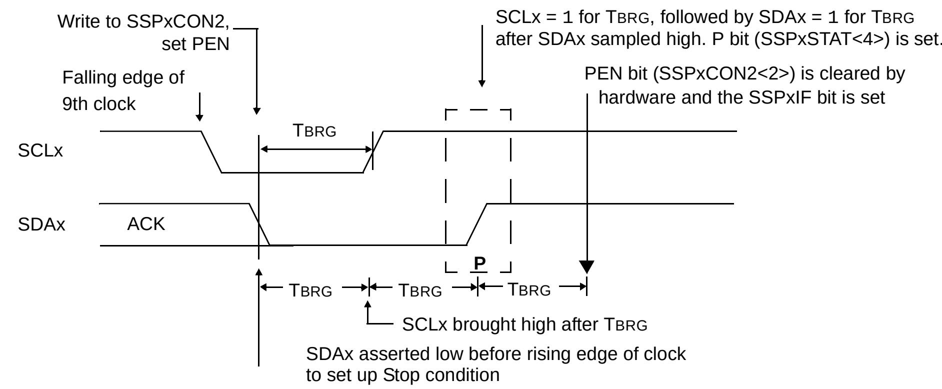


FIGURE 15-31: STOP CONDITION RECEIVE OR TRANSMIT MODE



Note: TBRG = one Baud Rate Generator period.

15.6.10 SLEEP OPERATION

While in Sleep mode, the I²C slave module can receive addresses or data and when an address match or complete byte transfer occurs, wake the processor from Sleep (if the MSSPx interrupt is enabled).

15.6.11 EFFECTS OF A RESET

A Reset disables the MSSPx module and terminates the current transfer.

15.6.12 MULTI-MASTER MODE

In Multi-Master mode, the interrupt generation on the detection of the Start and Stop conditions allows the determination of when the bus is free. The Stop (P) and Start (S) bits are cleared from a Reset or when the MSSPx module is disabled. Control of the I²C bus may be taken when the P bit of the SSPxSTAT register is set, or the bus is Idle, with both the S and P bits clear. When the bus is busy, enabling the SSPx interrupt will generate the interrupt when the Stop condition occurs.

In multi-master operation, the SDAx line must be monitored for arbitration to see if the signal level is the expected output level. This check is performed by hardware with the result placed in the BCLxIF bit.

The states where arbitration can be lost are:

- Address Transfer
- Data Transfer
- A Start Condition
- A Repeated Start Condition
- An Acknowledge Condition

16.1.1.5 TSR Status

The TRMT bit of the TXSTAx register indicates the status of the TSR register. This is a read-only bit. The TRMT bit is set when the TSR register is empty and is cleared when a character is transferred to the TSR register from the TXREGx. The TRMT bit remains clear until all bits have been shifted out of the TSR register. No interrupt logic is tied to this bit, so the user needs to poll this bit to determine the TSR status.

Note:	The TSR register is not mapped in data memory, so it is not available to the user.

16.1.1.6 Transmitting 9-Bit Characters

The EUSART supports 9-bit character transmissions. When the TX9 bit of the TXSTAx register is set the EUSART will shift nine bits out for each character transmitted. The TX9D bit of the TXSTAx register is the ninth, and Most Significant, data bit. When transmitting 9-bit data, the TX9D data bit must be written before writing the eight Least Significant bits into the TXREGx. All nine bits of data will be transferred to the TSR shift register immediately after the TXREGx is written.

A special 9-bit Address mode is available for use with multiple receivers. See **Section 16.1.2.8 “Address Detection”** for more information on the Address mode.

16.1.1.7 Asynchronous Transmission Setup:

1. Initialize the SPBRGHx:SPBRGx register pair and the BRGH and BRG16 bits to achieve the desired baud rate (see **Section 16.4 “EUSART Baud Rate Generator (BRG)”**).
2. Set the RXx/DTx and TXx/CKx TRIS controls to ‘1’.
3. Enable the asynchronous serial port by clearing the SYNC bit and setting the SPEN bit.
4. If 9-bit transmission is desired, set the TX9 control bit. A set ninth data bit will indicate that the eight Least Significant data bits are an address when the receiver is set for address detection.
5. Set the CKTXP control bit if inverted transmit data polarity is desired.
6. Enable the transmission by setting the TXEN control bit. This will cause the TXxIF interrupt bit to be set.
7. If interrupts are desired, set the TXxIE interrupt enable bit. An interrupt will occur immediately provided that the GIE/GIEH and PEIE/GIEL bits of the INTCON register are also set.
8. If 9-bit transmission is selected, the ninth bit should be loaded into the TX9D data bit.
9. Load 8-bit data into the TXREGx register. This will start the transmission.

FIGURE 16-3: ASYNCHRONOUS TRANSMISSION

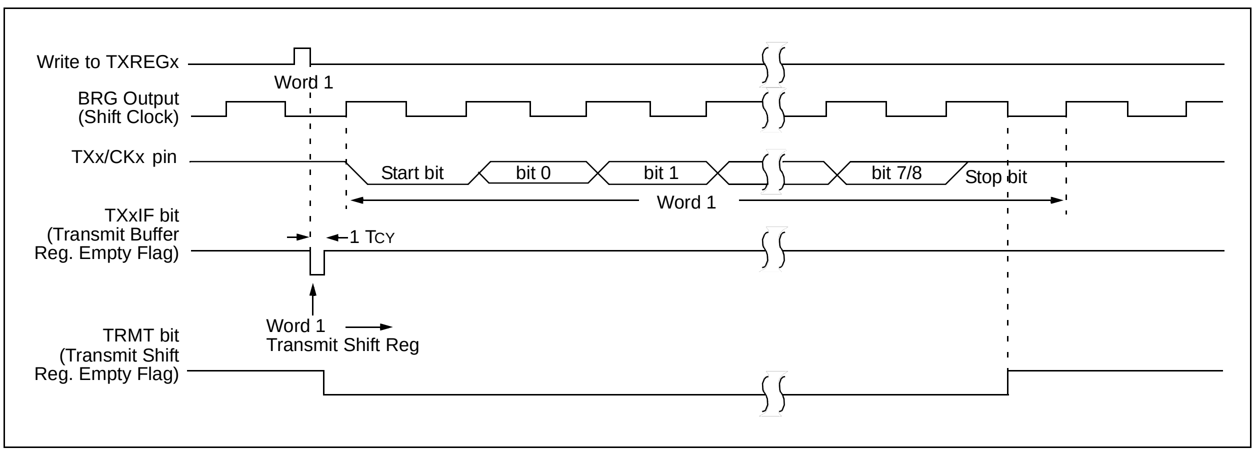
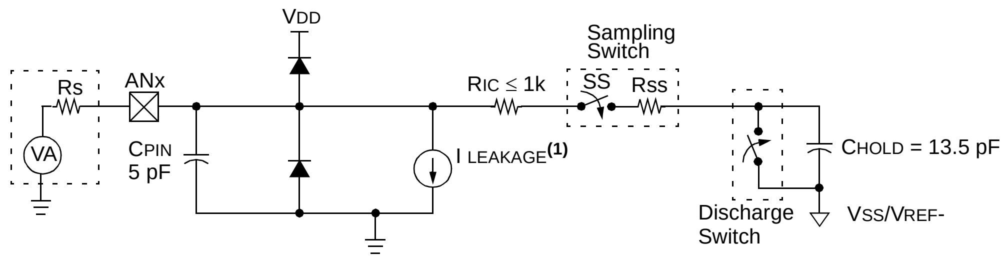
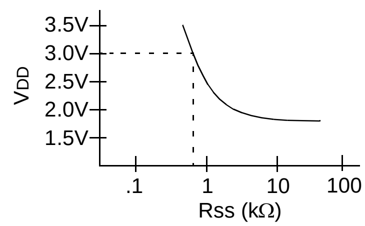


FIGURE 17-5: ANALOG INPUT MODEL

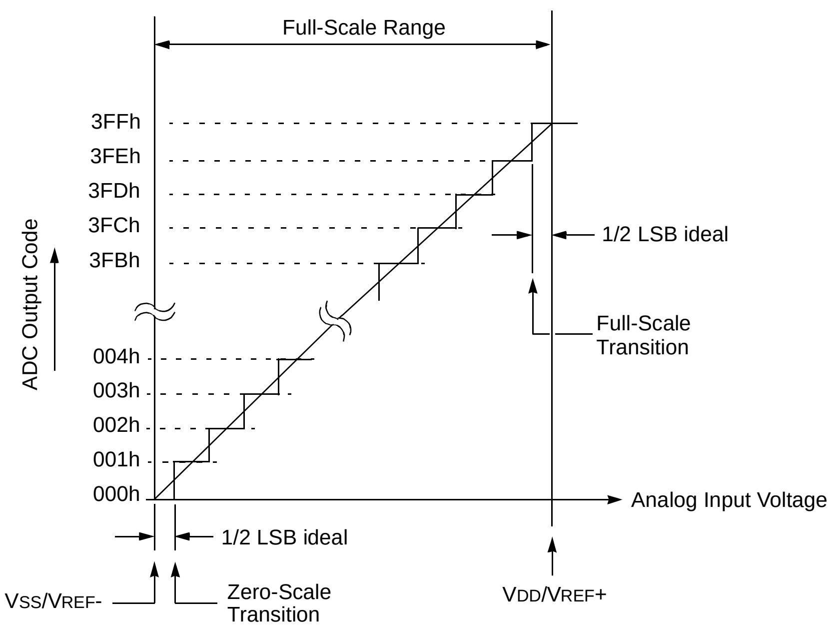




Legend:
- C_{PIN} = Input Capacitance
- $I_{LEAKAGE}$ = Leakage current at the pin due to various junctions
- R_{IC} = Interconnect Resistance
- SS = Sampling Switch
- C_{HOLD} = Sample/Hold Capacitance

Note 1: See **Section 27.0 “Electrical Specifications”**.

FIGURE 17-6: ADC TRANSFER FUNCTION



19.4 Measuring Capacitance with the CTMU

There are two separate methods of measuring capacitance with the CTMU. The first is the absolute method, in which the actual capacitance value is desired. The second is the relative method, in which the actual capacitance is not needed, rather an indication of a change in capacitance is required.

19.4.1 ABSOLUTE CAPACITANCE MEASUREMENT

For absolute capacitance measurements, both the current and capacitance calibration steps found in **Section 19.3 "Calibrating the CTMU Module"** should be followed. Capacitance measurements are then performed using the following steps:

1. Initialize the A/D Converter.
2. Initialize the CTMU.
3. Set EDG1STAT.
4. Wait for a fixed delay, *T*.
5. Clear EDG1STAT.
6. Perform an A/D conversion.
7. Calculate the total capacitance, $C_{TOTAL} = (I * T)/V$, where *I* is known from the current source measurement step (see **Section 19.3.1 "Current Source Calibration"**), *T* is a fixed delay and *V* is measured by performing an A/D conversion.
8. Subtract the stray and A/D capacitance (C_{OFFSET} from **Section 19.3.2 "Capacitance Calibration"**) from C_{TOTAL} to determine the measured capacitance.

19.4.2 RELATIVE CHARGE MEASUREMENT

An application may not require precise capacitance measurements. For example, when detecting a valid press of a capacitance-based switch, detecting a relative change of capacitance is of interest. In this type of application, when the switch is open (or not touched), the total capacitance is the capacitance of the combination of the board traces, the A/D Converter, etc. A larger voltage will be measured by the A/D Converter. When the switch is closed (or is touched), the total capacitance is larger due to the addition of the capacitance of the human body to the above listed capacitances, and a smaller voltage will be measured by the A/D Converter.

Detecting capacitance changes is easily accomplished with the CTMU using these steps:

1. Initialize the A/D Converter and the CTMU.
2. Set EDG1STAT.
3. Wait for a fixed delay.
4. Clear EDG1STAT.
5. Perform an A/D conversion.

The voltage measured by performing the A/D conversion is an indication of the relative capacitance. Note that in this case, no calibration of the current source or circuit capacitance measurement is needed. See Example 19-4 for a sample software routine for a capacitive touch switch.

REGISTER 19-3: CTMUICON: CTMU CURRENT CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ITRIM<5:0>						IRNG<1:0>	
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 7-2 **ITRIM<5:0>:** Current Source Trim bits
`011111` = Maximum positive change from nominal current
`011110`
.
.
.
`000001` = Minimum positive change from nominal current
`000000` = Nominal current output specified by IRNG<1:0>
`111111` = Minimum negative change from nominal current
.
.
.
`100010`
`100001` = Maximum negative change from nominal current

bit 1-0 **IRNG<1:0>:** Current Source Range Select bits (see Table 27-4)
`11` = 100 × Base current
`10` = 10 × Base current
`01` = Base current level
`00` = Current source disabled

TABLE 19-1: REGISTERS ASSOCIATED WITH CTMU MODULE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset Values on Page
CTMUCONH	CTMUEN	—	CTMUSIDL	TGEN	EDGEN	EDGSEQEN	IDISSEN	CTTRIG	323
CTMUCONL	EDG2POL	EDG2SEL<1:0>		EDG1POL	EDG1SEL<1:0>		EDG2STAT	EDG1STAT	324
CTMUICON	ITRIM<5:0>						IRNG<1:0>		325
IPR3	SSP2IP	BCL2IP	RC2IP	TX2IP	CTMUIP	TMR5GIP	TMR3GIP	TMR1GIP	123
PIE3	SSP2IE	BCL2IE	RC2IE	TX2IE	CTMUIE	TMR5GIE	TMR3GIE	TMR1GIE	119
PIR3	SSP2IF	BCL2IF	RC2IF	TX2IF	CTMUIF	TMR5GIF	TMR3GIF	TMR1GIF	114
PMD2	—	—	—	—	CTMUMD	CMP2MD	CMP1MD	ADCMD	54

Legend: — = unimplemented, read as ‘0’. Shaded bits are not used during CTMU operation.

23.3 HLVD Setup

To set up the HLVD module:

1. Select the desired HLVD trip point by writing the value to the HLVDL<3:0> bits.
2. Set the VDIRMAG bit to detect high voltage (VDIRMAG = 1) or low voltage (VDIRMAG = 0).
3. Enable the HLVD module by setting the HLVDEN bit.
4. Clear the HLVD interrupt flag (PIR2<2>), which may have been set from a previous interrupt.
5. If interrupts are desired, enable the HLVD interrupt by setting the HLVDIE and GIE/GIEH bits (PIE2<2> and INTCON<7>, respectively).

   An interrupt will not be generated until the IRVST bit is set.

Note:	Before changing any module settings (VDIRMAG, HLVDL<3:0>), first disable the module (HLVDEN = 0), make the changes and re-enable the module. This prevents the generation of false HLVD events.

23.4 Current Consumption

When the module is enabled, the HLVD comparator and voltage divider are enabled and consume static current. The total current consumption, when enabled, is specified in **Section 27.0 "Electrical Specifications"**. Depending on the application, the HLVD module does not need to operate constantly. To reduce current requirements, the HLVD circuitry may only need to be enabled for short periods where the voltage is checked. After such a check, the module could be disabled.

23.5 HLVD Start-up Time

The internal reference voltage of the HLVD module, specified in **Section 27.0 "Electrical Specifications"**, may be used by other internal circuitry, such as the programmable Brown-out Reset. If the HLVD or other circuits using the voltage reference are disabled to lower the device's current consumption, the reference voltage circuit will require time to become stable before a low or high-voltage condition can be reliably detected. This start-up time, TIRVST, is an interval that is independent of device clock speed.

The HLVD interrupt flag is not enabled until TIRVST has expired and a stable reference voltage is reached. For this reason, brief excursions beyond the set point may not be detected during this interval (see Figure 23-2 or Figure 23-3).

REGISTER 24-9: CONFIG6H: CONFIGURATION REGISTER 6 HIGH

R/C-1	R/C-1	R-1	U-0	U-0	U-0	U-0	U-0
WRTD	WRTB	WRTC(1)	—	—	—	—	—
bit 7							bit 0

Legend:	
R = Readable bit	U = Unimplemented bit, read as ‘0’
-n = Value when device is unprogrammed	C = Clearable only bit

bit 7 **WRTD:** Data EEPROM Write Protection bit
1 = Data EEPROM not write-protected
0 = Data EEPROM write-protected

bit 6 **WRTB:** Boot Block Write Protection bit
1 = Boot Block not write-protected
0 = Boot Block write-protected

bit 5 **WRTC:** Configuration Register Write Protection bit(1)
1 = Configuration registers not write-protected
0 = Configuration registers write-protected

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

Note 1: This bit is read-only in normal execution mode; it can be written only in Program mode.

REGISTER 24-10: CONFIG7L: CONFIGURATION REGISTER 7 LOW

U-0	U-0	U-0	U-0	R/C-1	R/C-1	R/C-1	R/C-1
—	—	—	—	EBTR3(1)	EBTR2(1)	EBTR1	EBTR0
bit 7							bit 0

Legend:	
R = Readable bit	U = Unimplemented bit, read as ‘0’
-n = Value when device is unprogrammed	C = Clearable only bit

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

bit 3 **EBTR3:** Table Read Protection bit(1)
1 = Block 3 not protected from table reads executed in other blocks
0 = Block 3 protected from table reads executed in other blocks

bit 2 **EBTR2**: Table Read Protection bit(1)
1 = Block 2 not protected from table reads executed in other blocks
0 = Block 2 protected from table reads executed in other blocks

bit 1 **EBTR1:** Table Read Protection bit
1 = Block 1 not protected from table reads executed in other blocks
0 = Block 1 protected from table reads executed in other blocks

bit 0 **EBTR0:** Table Read Protection bit
1 = Block 0 not protected from table reads executed in other blocks
0 = Block 0 protected from table reads executed in other blocks

Note 1: Available on PIC18(L)FX5K22 and PIC18(L)FX6K22s devices.

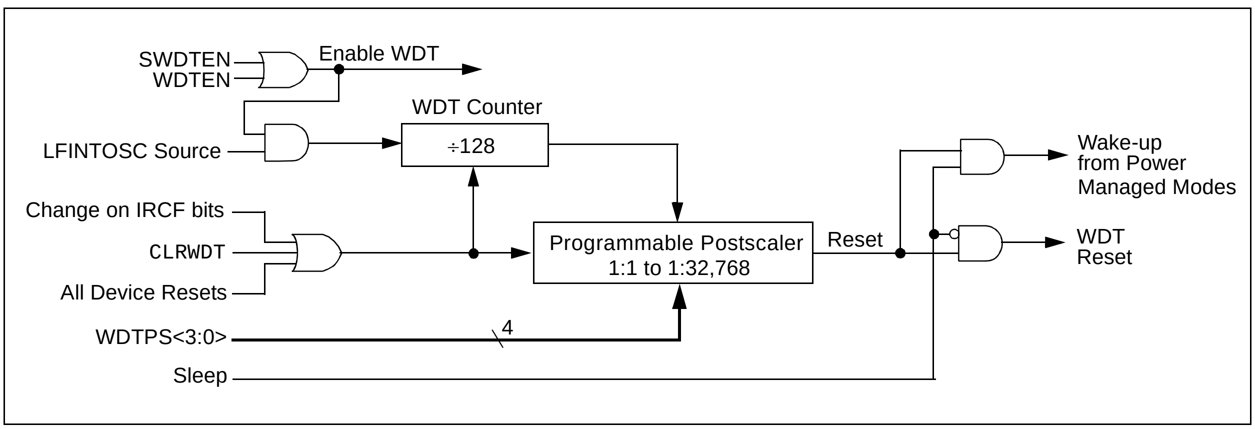
24.3 Watchdog Timer (WDT)

For PIC18(L)F2X/4XK22 devices, the WDT is driven by the LFINTOSC source. When the WDT is enabled, the clock source is also enabled. The nominal WDT period is 4 ms and has the same stability as the LFINTOSC oscillator.

The 4 ms period of the WDT is multiplied by a 16-bit postscaler. Any output of the WDT postscaler is selected by a multiplexer, controlled by bits in Configuration Register 2H. Available periods range from 4 ms to 131.072 seconds (2.18 minutes). The WDT and postscaler are cleared when any of the following events occur: a `SLEEP` or `CLRWDT` instruction is executed, the IRCF bits of the OSCCON register are changed or a clock failure has occurred.

> **Note 1:** The `CLRWDT` and `SLEEP` instructions clear the WDT and postscaler counts when executed.
>
> **2:** Changing the setting of the IRCF bits of the OSCCON register clears the WDT and postscaler counts.
>
> **3:** When a `CLRWDT` instruction is executed, the postscaler count will be cleared.

FIGURE 24-1: WDT BLOCK DIAGRAM



27.9 Memory Programming Requirements

DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ T_A ≤ +125°C				
Param No.	**Sym**	**Characteristic**	**Min**	**Typ†**	**Max**	**Units**	**Conditions**
		Internal Program Memory Programming Specifications(1)					
D170	V_{PP}	Voltage on $\overline{MCLR}$/V_{PP} pin	8	—	9	V	**(Note 3), (Note 4)**
D171	I_{DDP}	Supply Current during Programming	—	—	10	mA	
		Data EEPROM Memory					
D172	E_D	Byte Endurance	100K	—	—	E/W	-40°C to +85°C
D173	V_{DRW}	V_{DD} for Read/Write	V_{DDMIN}	—	V_{DDMAX}	V	Using EECON to read/ write
D175	T_{DEW}	Erase/Write Cycle Time	—	3	4	ms	
D176	T_{RETD}	Characteristic Retention	—	40	—	Year	Provided no other specifications are violated
D177	T_{REF}	Number of Total Erase/Write Cycles before Refresh(2)	1M	10M	—	E/W	-40°C to +85°C
		Program Flash Memory					
D178	E_P	Cell Endurance	10K	—	—	E/W	-40°C to +85°C **(Note 5)**
D179	V_{PR}	V_{DD} for Read	V_{DDMIN}	—	V_{DDMAX}	V	
D181	V_{IW}	V_{DD} for Row Erase or Write	2.2	—	V_{DDMAX}	V	PIC18LF24K22
D182	V_{IW}		V_{DDMIN}	—	V_{DDMAX}	V	PIC18(L)F26K22
D183	T_{IW}	Self-timed Write Cycle Time	—	2	—	ms	
D184	T_{RETD}	Characteristic Retention	—	40	—	Year	Provided no other specifications are violated

† Data in “Typ” column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: These specifications are for programming the on-chip program memory through the use of table write instructions.

2: Refer to **Section 7.8 “Using the Data EEPROM”** for a more detailed discussion on data EEPROM endurance.

3: Required only if single-supply programming is disabled.

4: The MPLAB ICD 2 does not support variable V_{PP} output. Circuitry to limit the MPLAB ICD 2 V_{PP} voltage must be placed between the MPLAB ICD 2 and target system when programming or debugging with the MPLAB ICD 2.

5: Self-write and Block Erase.

27.10 Analog Characteristics

TABLE 27-1: COMPARATOR SPECIFICATIONS

Operating Conditions: 1.8V < VDD < 5.5V, -40°C < TA < +125°C (unless otherwise stated)							
Param No.	**Sym**	**Characteristics**	**Min**	**Typ**	**Max**	**Units**	**Comments**
CM01	VIOFF	Input Offset Voltage	—	3	40	mV	High-Power mode VREF = VDD/2
			—	4	60	mV	Low-Power mode VREF = VDD/2
CM02	VICM	Input Common-mode Voltage	VSS	—	VDD	V	
CM04*	TRESP	Response Time(1)	—	200	400	ns	High-Power mode
			—	600	3500	ns	Low-Power mode
CM05*	TMC2OV	Comparator Mode Change to Output Valid	—	—	10	μs	

* These parameters are characterized but not tested.

Note 1: Response time measured with one comparator input at VDD/2, while the other input transitions from VSS to VDD.

TABLE 27-2: DIGITAL-TO-ANALOG CONVERTER (DAC) SPECIFICATIONS

Operating Conditions: 2.0V < VDD < 5.5V, -40°C < TA < +125°C (unless otherwise stated)							
Param No.	**Sym**	**Characteristics**	**Min**	**Typ**	**Max**	**Units**	**Comments**
CV01*	CLSB	Step Size(2)	—	VDD/32	—	V	
CV02*	CACC	Absolute Accuracy	—	—	± 1/2	LSb	ΔVSRC ≥ 2.0V
CV03*	CR	Unit Resistor Value (R)	—	5k	—	Ω	
CV04*	CST	Settling Time(1)	—	—	10	μs	
CV05*	VSRC+	DAC Positive Reference	VSRC- +2	—	VDD	V	
CV06*	VSRC-	DAC Negative Reference	VSS	—	VSRC+ -2	V	
CV07*	ΔVSRC	DAC Reference Range (VSRC+ - VSRC-)	2	—	VDD	V	

* These parameters are characterized but not tested.

Note 1: Settling time measured while CVRR = `1` and CVR3:CVR0 transitions from ‘`0000`’ to ‘`1111`’.

2: See **Section 22.0 “Digital-to-Analog Converter (DAC) Module”** for more information.