



Welcome to [E-XFL.COM](#)

What is "[Embedded - Microcontrollers](#)"?

"[Embedded - Microcontrollers](#)" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "[Embedded - Microcontrollers](#)"

Details

Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	64MHz
Connectivity	I ² C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LVD, POR, PWM, WDT
Number of I/O	25
Program Memory Size	64KB (32K x 16)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	3.6K 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-SSOP (0.209", 5.30mm Width)
Supplier Device Package	28-SSOP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18lf26k40-e-ss

3.3 Code Protection

Code protection allows the device to be protected from unauthorized access. Program memory protection and data memory are controlled independently. Internal access to the program memory is unaffected by any code protection setting.

3.3.1 PROGRAM MEMORY PROTECTION

The entire program memory space is protected from external reads and writes by the CP bit in Configuration Words. When $\overline{CP} = 0$, external reads and writes of program memory are inhibited and a read will return all '0's. The CPU can continue to read program memory, regardless of the protection bit settings. Self-writing the program memory is dependent upon the write protection setting. See **Section 3.4 “Write Protection”** for more information.

3.3.2 DATA MEMORY PROTECTION

The entire Data EEPROM Memory space is protected from external reads and writes by the CPD bit in the Configuration Words. When $\overline{CPD} = 0$, external reads and writes of Data EEPROM Memory are inhibited and a read will return all '0's. The CPU can continue to read Data EEPROM Memory regardless of the protection bit settings.

3.4 Write Protection

Write protection allows the device to be protected from unintended self-writes. Applications, such as boot loader software, can be protected while allowing other regions of the program memory to be modified.

The WRT<1:0> bits in Configuration Words define the size of the program memory block that is protected.

3.5 User ID

Eight words in the memory space (200000h-200000Fh) are designated as ID locations where the user can store checksum or other code identification numbers. These locations are readable and writable during normal execution. See **Section 11.2 “User ID, Device ID and Configuration Word Access”** for more information on accessing these memory locations. For more information on checksum calculation, see the “PIC18(L)F2X/4XK40 Memory Programming Specification” (DS40001772).

10.4.2 ACCESS BANK

While the use of the BSR with an embedded 8-bit address allows users to address the entire range of data memory, it also means that the user must always ensure that the correct bank is selected. Otherwise, data may be read from or written to the wrong location. This can be disastrous if a GPR is the intended target of an operation, but an SFR is written to instead. Verifying and/or changing the BSR for each read or write to data memory can become very inefficient.

To streamline access for the most commonly used data memory locations, the data memory is configured with an Access Bank, which allows users to access a mapped block of memory without specifying a BSR. The Access Bank consists of the first 96 bytes of memory (00h-5Fh) in Bank 0 and the last 160 bytes of memory (60h-FFh) in Bank 15. The lower half is known as the “Access RAM” and is composed of GPRs. This upper half is also where the device’s SFRs are mapped. These two areas are mapped contiguously in the Access Bank and can be addressed in a linear fashion by an 8-bit address (Figure 10-4).

The Access Bank is used by core PIC18 instructions that include the Access RAM bit (the ‘a’ parameter in the instruction). When ‘a’ is equal to ‘1’, the instruction uses the BSR and the 8-bit address included in the opcode for the data memory address. When ‘a’ is ‘0’, however, the instruction is forced to use the Access Bank address map; the current value of the BSR is ignored entirely.

Using this “forced” addressing allows the instruction to operate on a data address in a single cycle, without updating the BSR first. For 8-bit addresses of 60h and above, this means that users can evaluate and operate on SFRs more efficiently. The Access RAM below 60h is a good place for data values that the user might need to access rapidly, such as immediate computational results or common program variables. Access RAM also allows for faster and more code efficient context saving and switching of variables.

The mapping of the Access Bank is slightly different when the extended instruction set is enabled (XINST Configuration bit = 1). This is discussed in more detail in **Section 10.7.3 “Mapping the Access Bank in Indexed Literal Offset Mode”**.

10.4.3 GENERAL PURPOSE REGISTER FILE

PIC18 devices may have banked memory in the GPR area. This is data RAM, which is available for use by all instructions. GPRs start at the bottom of Bank 0 (address 000h) and grow upwards towards the bottom of the SFR area. GPRs are not initialized by a Power-on Reset and are unchanged on all other Resets.

10.4.4 SPECIAL FUNCTION REGISTERS

The Special Function Registers (SFRs) are registers used by the CPU and peripheral modules for controlling the desired operation of the device. These registers are implemented as static RAM. SFRs start at the top of data memory (FFFh) and extend downward to occupy the top portion of Bank 15 (F38h to FFFh). A list of these registers is given in Table 10-3 and Table 10-4.

The SFRs can be classified into two sets: those associated with the “core” device functionality (ALU, Resets and interrupts) and those related to the peripheral functions. The Reset and Interrupt registers are described in their respective chapters, while the ALU’s STATUS register is described later in this section. Registers related to the operation of a peripheral feature are described in the chapter for that peripheral.

The SFRs are typically distributed among the peripherals whose functions they control. Unused SFR locations are unimplemented and read as ‘0’s.

11.3.6 OPERATION DURING CODE-PROTECT

Data EEPROM Memory has its own code-protect bits in Configuration Words. External read and write operations are disabled if code protection is enabled.

If the Data EEPROM is write-protected or if NVMADR points an invalid address location, the WR bit is cleared without any effect. WRERR is signaled in this scenario.

11.3.7 PROTECTION AGAINST SPURIOUS WRITE

There are conditions when the user may not want to write to the Data EEPROM Memory. To protect against spurious EEPROM writes, various mechanisms have been implemented. On power-up, the WREN bit is cleared. In addition, writes to the EEPROM are blocked during the Power-up Timer period (TPWRT).

The unlock sequence and the WREN bit together help prevent an accidental write during brown-out, power glitch or software malfunction.

11.3.8 ERASING THE DATA EEPROM MEMORY

Data EEPROM Memory can be erased by writing 0xFF to all locations in the Data EEPROM Memory that needs to be erased.

EXAMPLE 11-7: DATA EEPROM REFRESH ROUTINE

```

        CLRF    NVMADRL          ; Clear address low byte register
        CLRF    NVMADRH          ; Clear address high byte register (if applicable)
        BCF     NVMCON1, NVMREG0  ; Set access for EEPROM
        BCF     NVMCON1, NVMREG1  ; Set access for EEPROM
        SETF    NVMDAT           ; Load 0xFF to data register
        BCF     INTCON, GIE       ; Disable interrupts
        BSF     NVMCON1, WREN     ; Enable writes
Loop:
        MOVLW   0x55             ; Initiate unlock sequence
        MOVWF   NVMCON2          ;
        MOVLW   0xAA             ;
        MOVWF   NVMCON2          ;
        BSF     NVMCON1, WR       ; Set WR bit to begin write
        BTFSC   NVMCON1, WR       ; Wait for write to complete
        BRA     $-2              ;
        INCFSZ  NVMADRL, F        ; Increment address low byte
        BRA     Loop             ; Not zero, do it again

//The following 4 lines of code are not needed if the part doesn't have NVMADRH register
        INCF    NVMADRH, F        ; Decrement address high byte
        MOVLW   0x03             ; Move 0x03 to working register
        CPFSGT  NVMADRH          ; Compare address high byte with working register
        BRA     Loop             ; Skip if greater than working register
                                ; Else go back to erase loop

        BCF     NVMCON1, WREN     ; Disable writes
        BSF     INTCON, GIE       ; Enable interrupts
    
```

REGISTER 14-5: PIR3: PERIPHERAL INTERRUPT REQUEST (FLAG) REGISTER 3

R-0/0	R-0/0	R-0/0	R-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
RC2IF	TX2IF	RC1IF	TX1IF	BCL2IF	SSP2IF	BCL1IF	SSP1IF
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	RC2IF: EUSART2 Receive Interrupt Flag bit 1 = The EUSART2 receive buffer, RC1REG, is full (cleared by reading RC2REG) 0 = The EUSART2 receive buffer is empty
bit 6	TX2IF: EUSART2 Transmit Interrupt Flag bit 1 = The EUSART2 transmit buffer, TX2REG, is empty (cleared by writing TX2REG) 0 = The EUSART2 transmit buffer is full
bit 5	RC1IF: EUSART1 Receive Interrupt Flag bit 1 = The EUSART1 receive buffer, RC1REG, is full (cleared by reading RC1REG) 0 = The EUSART1 receive buffer is empty
bit 4	TX1IF: EUSART1 Transmit Interrupt Flag bit 1 = The EUSART1 transmit buffer, TX1REG, is empty (cleared by writing TX1REG) 0 = The EUSART1 transmit buffer is full
bit 3	BCL2IF: MSSP2 Bus Collision Interrupt Flag bit 1 = A bus collision has occurred while the MSSP2 module configured in I ² C master was transmitting (must be cleared in software) 0 = No bus collision occurred
bit 2	SSP2IF: Synchronous Serial Port 2 Interrupt Flag bit 1 = The transmission/reception is complete (must be cleared in software) 0 = Waiting to transmit/receive
bit 1	BCL1IF: MSSP1 Bus Collision Interrupt Flag bit 1 = A bus collision has occurred while the MSSP1 module configured in I ² C master was transmitting (must be cleared in software) 0 = No bus collision occurred
bit 0	SSP1IF: Synchronous Serial Port 1 Interrupt Flag bit 1 = The transmission/reception is complete (must be cleared in software) 0 = Waiting to transmit/receive

17.3 Bidirectional Pins

PPS selections for peripherals with bidirectional signals on a single pin must be made so that the PPS input and PPS output select the same pin. Peripherals that have bidirectional signals include:

- EUSART (synchronous operation)
- MSSP (I²C)
- CCP module

Note: The I²C default input pins are I²C and SMBus compatible. RB1 and RB2 are additional pins. RC4 and RC3 are default MMP1 pins and are SMBus compatible. Clock and data signals can be routed to any pin, however pins without I²C compatibility will operate at standard TTL/ST logic levels as selected by the INVLV register.

17.4 PPS Lock

The PPS includes a mode in which all input and output selections can be locked to prevent inadvertent changes. PPS selections are locked by setting the PPSLOCKED bit of the PPSLOCK register. Setting and clearing this bit requires a special sequence as an extra precaution against inadvertent changes. Examples of setting and clearing the PPSLOCKED bit are shown in Example 17-1.

EXAMPLE 17-1: PPS LOCK SEQUENCE

```
; Disable interrupts:
BCF    INTCON,GIE

; Bank to PPSLOCK register
BANKSEL PPSLOCK
MOVLB   PPSLOCK
MOVLW   55h

; Required sequence, next 4 instructions
MOVWF   PPSLOCK
MOVLW   AAh
MOVWF   PPSLOCK

; Set PPSLOCKED bit to disable writes
; Only a BSF instruction will work
BSF     PPSLOCK,0

; Enable Interrupts
BSF     INTCON,GIE
```

EXAMPLE 17-2: PPS UNLOCK SEQUENCE

```
; Disable interrupts:
BCF     INTCON,GIE

; Bank to PPSLOCK register
BANKSEL PPSLOCK
MOVLB   PPSLOCK
MOVLW   55h

; Required sequence, next 4 instructions
MOVWF   PPSLOCK
MOVLW   AAh
MOVWF   PPSLOCK

; Clear PPSLOCKED bit to enable writes
; Only a BCF instruction will work
BCF     PPSLOCK,0

; Enable Interrupts
BSF     INTCON,GIE
```

17.5 PPS One-Way Lock

Using the PPS1WAY Configuration bit, the PPS settings can be locked in. When this bit is set, the PPSLOCKED bit can only be cleared and set one time after a device Reset. This allows for clearing the PPSLOCKED bit so that the input and output selections can be made during initialization. When the PPSLOCKED bit is set after all selections have been made, it will remain set and cannot be cleared until after the next device Reset event.

17.6 Operation During Sleep

PPS input and output selections are unaffected by Sleep.

17.7 Effects of a Reset

A device Power-on-Reset (POR) clears all PPS input and output selections to their default values. All other Resets leave the selections unchanged. Default input selections are shown in the **Section “Pin Allocation Tables”**. The PPS one-way is also removed.

19.4 Timer1/3/5 Prescaler

Timer1/3/5 has four prescaler options allowing 1, 2, 4 or 8 divisions of the clock input. The CKPS bits of the TxCON register control the prescale counter. The prescale counter is not directly readable or writable; however, the prescaler counter is cleared upon a write to TMRxH or TMRxL.

19.5 Secondary Oscillator

A secondary low-power 32.768 kHz oscillator circuit is built-in between pins SOSCI (input) and SOSCO (amplifier output). This internal circuit is to be used in conjunction with an external 32.768 kHz crystal. The secondary oscillator is not dedicated only to Timer1/3/5; it can also be used by other modules.

The oscillator circuit is enabled by setting the SOSSEN bit of the OSCEN register (Register 4-7). This can be used as the clock source to the Timer using the TMRxCLK bits. The oscillator will continue to run during Sleep.

Note: The oscillator requires a start-up and stabilization time before use. Thus, the SOSSEN bit of the OSCEN register should be set and a suitable delay observed prior to enabling Timer1/3/5. A software check can be performed to confirm if the secondary oscillator is enabled and ready to use. This is done by polling the SOR bit of the OSCSTAT (Register 4-4).

19.6 Timer1/3/5 Operation in Asynchronous Counter Mode

If control bit $\overline{\text{SYNC}}$ of the TxCON register is set, the external clock input is not synchronized. The timer increments asynchronously to the internal phase clocks. If external clock source is selected then the timer will continue to run during Sleep and can generate an interrupt on overflow, which will wake-up the processor. However, special precautions in software are needed to read/write the timer (see **Section 19.6.1 “Reading and Writing Timer1/3/5 in Asynchronous Counter Mode”**).

Note: When switching from synchronous to asynchronous operation, it is possible to skip an increment. When switching from asynchronous to synchronous operation, it is possible to produce an additional increment.

19.6.1 READING AND WRITING TIMER1/3/5 IN ASYNCHRONOUS COUNTER MODE

Reading TMRxH or TMRxL while the timer is running from an external asynchronous clock will ensure a valid read (taken care of in hardware). However, the user should keep in mind that reading the 16-bit timer in two 8-bit values itself, poses certain problems, since the timer may overflow between the reads. For writes, it is recommended that the user simply stop the timer and write the desired values. A write contention may occur by writing to the timer registers, while the register is incrementing. This may produce an unpredictable value in the TMRxH:TMRxL register pair.

24.0 COMPLEMENTARY WAVEFORM GENERATOR (CWG) MODULE

The Complementary Waveform Generator (CWG) produces half-bridge, full-bridge, and steering of PWM waveforms. It is backwards compatible with previous CCP functions. The PIC18(L)F2x/4xK40 family has one instance of the CWG module.

The CWG has the following features:

- Six operating modes:
 - Synchronous Steering mode
 - Asynchronous Steering mode
 - Full-Bridge mode, Forward
 - Full-Bridge mode, Reverse
 - Half-Bridge mode
 - Push-Pull mode
- Output polarity control
- Output steering
- Independent 6-bit rising and falling event dead-band timers
 - Clocked dead band
 - Independent rising and falling dead-band enables
- Auto-shutdown control with:
 - Selectable shutdown sources
 - Auto-restart option
 - Auto-shutdown pin override control

24.1 Fundamental Operation

The CWG generates two output waveforms from the selected input source.

The off-to-on transition of each output can be delayed from the on-to-off transition of the other output, thereby, creating a time delay immediately where neither output is driven. This is referred to as dead time and is covered in **Section 24.6 “Dead-Band Control”**.

It may be necessary to guard against the possibility of circuit faults or a feedback event arriving too late or not at all. In this case, the active drive must be terminated before the Fault condition causes damage. This is referred to as auto-shutdown and is covered in **Section 24.10 “Auto-Shutdown”**.

24.2 Operating Modes

The CWG module can operate in six different modes, as specified by the MODE<2:0> bits of the CWG1CON0 register:

- Half-Bridge mode
- Push-Pull mode
- Asynchronous Steering mode
- Synchronous Steering mode
- Full-Bridge mode, Forward
- Full-Bridge mode, Reverse

All modes accept a single pulse data input, and provide up to four outputs as described in the following sections.

All modes include auto-shutdown control as described in **Section 24.10 “Auto-Shutdown”**

Note: Except as noted for Full-bridge mode (**Section 24.2.3 “Full-Bridge Modes”**), mode changes should only be performed while EN = 0 (Register 24-1).

24.2.1 HALF-BRIDGE MODE

In Half-Bridge mode, two output signals are generated as true and inverted versions of the input as illustrated in Figure 24-2. A non-overlap (dead-band) time is inserted between the two outputs to prevent shoot through current in various power supply applications. Dead-band control is described in **Section 24.6 “Dead-Band Control”**. The output steering feature cannot be used in this mode. A basic block diagram of this mode is shown in Figure 24-1.

The unused outputs CWG1C and CWG1D drive similar signals, with polarity independently controlled by the POLC and POLD bits of the CWG1CON1 register, respectively.

PIC18LF26/45/46K40

REGISTER 25-3: MDCARH: MODULATION HIGH CARRIER CONTROL REGISTER

U-0	U-0	U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—	—	—	CHS<2:0> ⁽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 7-3 **Unimplemented:** Read as '0'
bit 2-0 **CHS<2:0>:** Modulator Carrier High Selection bits
See Table 25-2 for signal list

REGISTER 25-4: MDCARL: MODULATION LOW CARRIER CONTROL REGISTER

U-0	U-0	U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—	—	—	CLS<2:0> ⁽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 7-3 **Unimplemented:** Read as '0'
bit 2-0 **CLS<2:0>:** Modulator Carrier Low Input Selection bits
See Table 25-2 for signal list

TABLE 25-2: MDCARH/MDCARL SELECTION MUX CONNECTIONS

MDCARH			MDCARL		
CHS<2:0>		Connection	CLS<2:0>		Connection
111	7	PWM4 OUT	111	7	PWM4 OUT
110	6	PWM3 OUT	110	6	PWM3 OUT
101	5	CCP2 OUT	101	5	CCP2 OUT
100	4	CCP1 OUT	100	4	CCP1 OUT
011	3	CLKREF output	011	3	CLKREF output
010	2	HFINTOSC	010	2	HFINTOSC
001	1	FOSC (system clock)	001	1	FOSC (system clock)
000	0	Pin selected by MDCARHPPS	000	0	Pin selected by MDCARLPPS

REGISTER 26-11: SSPxADD: MSSP ADDRESS REGISTER (I²C MASTER MODE)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADD<7:0>							
bit 7				bit 0			

Legend:

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

Master mode: I²C mode

bit 7-0 Baud Rate Clock Divider bits⁽¹⁾

$$\text{SCK/SCL pin clock period} = ((\text{SSPxADD}<7:0> + 1) * 4) / \text{Fosc}$$

10-Bit Slave mode – Most Significant Address Byte:

bit 7-3 **Not used:** Unused for Most Significant Address Byte. Bit state of this register is a don't care. Bit pattern sent by master is fixed by I²C specification and must be equal to, '11110'. However, those bits are compared by hardware and are not affected by the value in this register.

bit 2-1 **ADD<9:8>:** Two Most Significant bits of 10-bit Address

bit 0 **Not used:** Unused in this mode. Bit state is a don't care.

10-Bit Slave mode – Least Significant Address Byte:

bit 7-0 **ADD<7:0>:** Eight Least Significant bits of 10-bit Address

7-Bit Slave mode:

bit 7-1 7-bit Slave Address

bit 0 **Not used:** Unused in this mode. Bit state is a don't care.

Note 1: Values of 0x00, 0x01 and 0x02 are not valid for SSPxADD when used as a Baud Rate Generator for I²C. This is an implementation limitation.

REGISTER 26-12: SSPxMSK: MSSPx ADDRESS MASK REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
MSK<7:1>							MSK0
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 7-1 **MSK<7:1>:** Mask bits

1 = The received address bit n is compared to SSPxADDn to detect I²C address match

0 = The received address bit n is not used to detect I²C address match

bit 0 **MSK0:** Mask bit for I²C Slave mode, 10-bit Address

I²C Slave mode, 10-bit address (SSPM<3:0> = 0111 or 1111):

1 = The received address bit 0 is compared to SSPxADD0 to detect I²C address match

0 = The received address bit 0 is not used to detect I²C address match

I²C Slave mode, 7-bit address, the bit is ignored.

27.6 EUSART Operation During Sleep

The EUSART will remain active during Sleep only in the Synchronous Slave mode. All other modes require the system clock and therefore cannot generate the necessary signals to run the Transmit or Receive Shift registers during Sleep.

Synchronous Slave mode uses an externally generated clock to run the Transmit and Receive Shift registers.

27.6.1 SYNCHRONOUS RECEIVE DURING SLEEP

To receive during Sleep, all the following conditions must be met before entering Sleep mode:

- RCxSTA and TXxSTA Control registers must be configured for Synchronous Slave Reception (see **Section 27.5.2.4 “Synchronous Slave Reception Setup:”**).
- If interrupts are desired, set the RCxIE bit of the PIE3 register and the GIE and PEIE bits of the INTCON register.
- The RCxIF interrupt flag must be cleared by reading RCxREG to unload any pending characters in the receive buffer.

Upon entering Sleep mode, the device will be ready to accept data and clocks on the RXx/DTx and TXx/CKx pins, respectively. When the data word has been completely clocked in by the external device, the RCxIF interrupt flag bit of the PIR3 register will be set. Thereby, waking the processor from Sleep.

Upon waking from Sleep, the instruction following the SLEEP instruction will be executed. If the Global Interrupt Enable (GIE) bit of the INTCON register is also set, then the Interrupt Service Routine at address 004h will be called.

27.6.2 SYNCHRONOUS TRANSMIT DURING SLEEP

To transmit during Sleep, all the following conditions must be met before entering Sleep mode:

- The RCxSTA and TXxSTA Control registers must be configured for synchronous slave transmission (see **Section 27.5.2.2 “Synchronous Slave Transmission Setup”**).
- The TXxIF interrupt flag must be cleared by writing the output data to the TXxREG, thereby filling the TSR and transmit buffer.
- If interrupts are desired, set the TXxIE bit of the PIE3 register and the PEIE bit of the INTCON register.
- Interrupt enable bits TXxIE of the PIE3 register and PEIE of the INTCON register must set.

Upon entering Sleep mode, the device will be ready to accept clocks on TXx/CKx pin and transmit data on the RXx/DTx pin. When the data word in the TSR has been completely clocked out by the external device, the pending byte in the TXxREG will transfer to the TSR and the TXxIF flag will be set. Thereby, waking the processor from Sleep. At this point, the TXxREG is available to accept another character for transmission, which will clear the TXxIF flag.

Upon waking from Sleep, the instruction following the SLEEP instruction will be executed. If the Global Interrupt Enable (GIE) bit is also set then the Interrupt Service Routine at address 0004h will be called.

PIC18(L)F26/45/46K40

REGISTER 30-2: DAC1CON1: DAC DATA REGISTER

U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—	DAC1R<4:0>				
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 7-5

Unimplemented: Read as '0'

bit 4-0

DAC1R<4:0>: Data Input Register for DAC bits

TABLE 30-2: SUMMARY OF REGISTERS ASSOCIATED WITH THE DAC MODULE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
DAC1CON0	EN	—	OE1	OE2	PSS<1:0>		—	NSS	428
DAC1CON1	—	—	—	DAC1R<4:0>					429
FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDAFVR<1:0>		ADFVR<1:0>		423

Legend: — = Unimplemented location, read as '0'. Shaded cells are not used with the DAC module.

31.5.8 CONTINUOUS SAMPLING MODE

Setting the ADCONT bit in the ADCON0 register automatically retriggers a new conversion cycle after updating the ADACC register. That means the ADGO bit is set to generate automatic retriggering, until the device Reset occurs or the A/D Stop-on-interrupt bit (ADSOI in the ADCON3 register) is set (correct logic).

31.5.9 DOUBLE SAMPLE CONVERSION

Double sampling is enabled by setting the ADDSEN bit of the ADCON1 register. When this bit is set, two conversions are required before the module will calculate threshold error (each conversion must still be triggered separately). The first conversion will set the ADMATH bit of the ADSTAT register and update ADACC, but will not calculate ADERR or trigger ADTIF. When the second conversion completes, the first value is transferred to ADPREV (depending on the setting of ADPSIS) and the value of the second conversion is placed into ADRES. Only upon the completion of the second conversion is ADERR calculated and ADTIF triggered (depending on the value of ADCALC).

31.6 Register Definitions: ADC Control

REGISTER 31-1: ADCON0: ADC CONTROL REGISTER 0

R/W-0/0	R/W-0/0	U-0	R/W-0/0	U-0	R/W-0/0	U-0	R/W/HC-0
ADON	ADCONT	—	ADCS	—	ADFM	—	ADGO
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

u = Bit is unchanged

x = Bit is unknown

-n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set

'0' = Bit is cleared

HC = Bit is cleared by hardware

bit 7	ADON: ADC Enable bit 1 = ADC is enabled 0 = ADC is disabled
bit 6	ADCONT: ADC Continuous Operation Enable bit 1 = ADGO is retriggered upon completion of each conversion trigger until ADTIF is set (if ADSOI is set) or until ADGO is cleared (regardless of the value of ADSOI) 0 = ADC is cleared upon completion of each conversion trigger
bit 5	Unimplemented: Read as '0'
bit 4	ADCS: ADC Clock Selection bit 1 = Clock supplied from FRC dedicated oscillator 0 = Clock supplied by FOSC, divided according to ADCLK register
bit 3	Unimplemented: Read as '0'
bit 2	ADFM: ADC results Format/alignment Selection 1 = ADRES and ADPREV data are right-justified 0 = ADRES and ADPREV data are left-justified, zero-filled
bit 1	Unimplemented: Read as '0'
bit 0	ADGO: ADC Conversion Status bit 1 = ADC conversion cycle in progress. Setting this bit starts an ADC conversion cycle. The bit is cleared by hardware as determined by the A/DCONT bit 0 = ADC conversion completed/not in progress

PIC18(L)F26/45/46K40

REGISTER 31-24: ADSTPTH: ADC THRESHOLD SETPOINT REGISTER HIGH

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
ADSTPT<15:8>							
bit 7				bit 0			

Legend:

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

bit 7-0 **ADSTPT<15:8>**: ADC Threshold Setpoint MSB. Upper byte of ADC threshold setpoint, depending on ADCALC, may be used to determine ADERR, see Register 23-1 for more details.

REGISTER 31-25: ADSTPTL: ADC THRESHOLD SETPOINT REGISTER LOW

R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x
ADSTPT<7:0>							
bit 7				bit 0			

Legend:

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

bit 7-0 **ADSTPT<7:0>**: ADC Threshold Setpoint LSB. Lower byte of ADC threshold setpoint, depending on ADCALC, may be used to determine ADERR, see Register 23-1 for more details.

REGISTER 31-26: ADERRH: ADC SETPOINT ERROR REGISTER HIGH

R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
ADERR<7:0>							
bit 7				bit 0			

Legend:

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

bit 7-0 **ADERR<7:0>**: ADC Setpoint Error MSB. Upper byte of ADC Setpoint Error. Setpoint Error calculation is determined by ADCALC bits of ADCON3, see Register 23-1 for more details.

PIC18(L)F26/45/46K40

REGISTER 31-32: ADOACT: ADC AUTO CONVERSION TRIGGER CONTROL REGISTER

U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—	ADOACT<4:0>				
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 7-5 **Unimplemented:** Read as '0'

bit 4-0 **ADOACT<4:0>:** Auto-Conversion Trigger Select Bits

11111 = Software write to ADPCH

11110 = Reserved, do not use

11101 = Software read of ADRESH

11100 = Software read of ADERRH

11011 = Reserved, do not use

•

•

•

10000 = Reserved, do not use

01111 = Interrupt-on-change Interrupt Flag

01110 = C2_out

01101 = C1_out

01100 = PWM4_out

01011 = PWM3_out

01010 = CCP2_trigger

01001 = CCP1_trigger

01000 = TMR6_postscaled

00111 = TMR5_overflow

00110 = TMR4_postscaled

00101 = TMR3_overflow

00100 = TMR2_postscaled

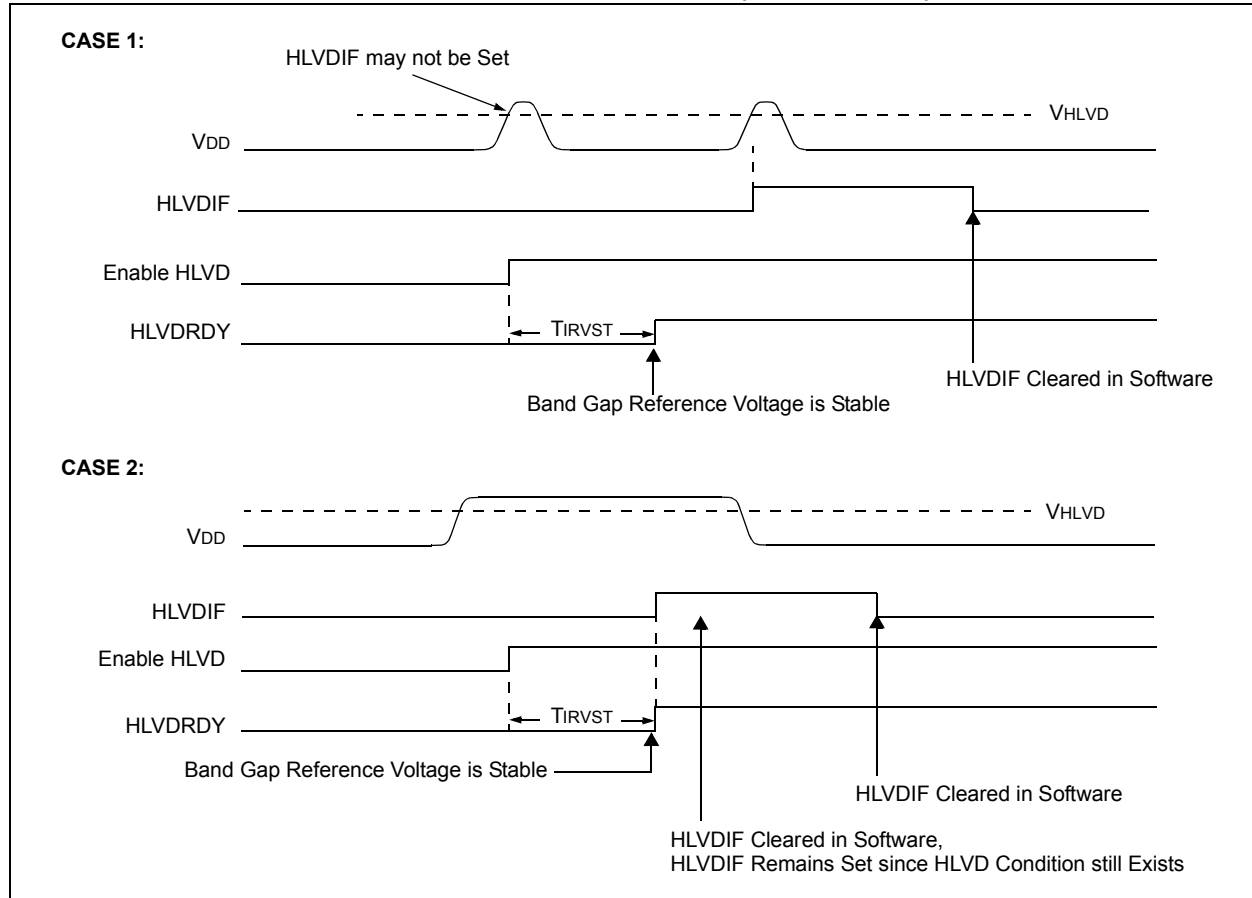
00011 = TMR1_overflow

00010 = TMR0_overflow

00001 = Pin selected by ADOACTPPS

00000 = External Trigger Disabled

FIGURE 33-3: HIGH-VOLTAGE DETECT OPERATION (HLVDINTH = 1)

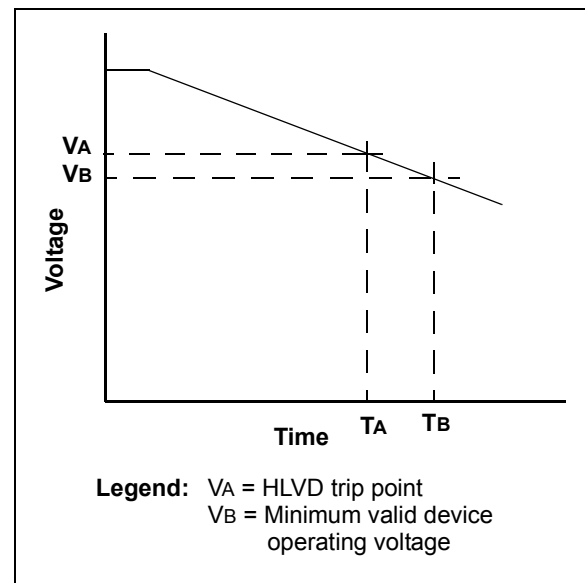


33.5 Applications

In many applications, it is desirable to detect a drop below, or rise above, a particular voltage threshold. For example, the HLVD module could be periodically enabled to detect Universal Serial Bus (USB) attach or detach. This assumes the device is powered by a lower voltage source than the USB when detached. An attach would indicate a High-Voltage Detect from, for example, 3.3V to 5V (the voltage on USB) and vice versa for a detach. This feature could save a design a few extra components and an attach signal (input pin).

For general battery applications, Figure 33-4 shows a possible voltage curve. Over time, the device voltage decreases. When the device voltage reaches voltage, V_A , the HLVD logic generates an interrupt at time, T_A . The interrupt could cause the execution of an Interrupt Service Routine (ISR), which would allow the application to perform "housekeeping tasks" and a controlled shutdown before the device voltage exits the valid operating range at T_B . This would give the application a time window, represented by the difference between T_A and T_B , to safely exit.

FIGURE 33-4: TYPICAL LOW-VOLTAGE DETECT APPLICATION



PIC18(L)F26/45/46K40

REGISTER 33-2: HLVDCON0: HIGH/LOW-VOLTAGE DETECT CONTROL REGISTER 0

R/W-0/0	U-0	R-x	R-x	U-0	U-0	R/W-0/0	R/W-0/0
EN	—	OUT	RDY	—	—	INTH	INTL
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

- bit 7 **EN:** High/Low-voltage Detect Power Enable bit
1 = Enables HLVD, powers up HLVD circuit and supporting reference circuitry
0 = Disables HLVD, powers down HLVD and supporting circuitry
- bit 6 **Unimplemented:** Read as '0'
- bit 5 **OUT:** HLVD Comparator Output bit
1 = Voltage \leq selected detection limit (HLVDL<3:0>)
0 = Voltage \geq selected detection limit (HLVDL<3:0>)
- bit 4 **RDY:** Band Gap Reference Voltages Stable Status Flag bit
1 = Indicates HLVD Module is ready and output is stable
0 = Indicates HLVD Module is not ready
- bit 3-2 **Unimplemented:** Read as '0'
- bit 1 **INTH:** HLVD Positive going (High Voltage) Interrupt Enable
1 = HLVDIF will be set when voltage \geq selected detection limit (HLVDSEL<3:0>)
0 = HLVDIF will not be set
- bit 0 **INTL:** HLVD Negative going (Low Voltage) Interrupt Enable
1 = HLVDIF will be set when voltage \leq selected detection limit (HLVDSEL<3:0>)
0 = HLVDIF will not be set

TABLE 33-2: REGISTERS ASSOCIATED WITH HIGH/LOW-VOLTAGE DETECT MODULE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
HLVDCON0	EN	—	OUT	RDY	—	—	INTH	INTL	482
HLVDCON1	—	—	—	—	SEL<3:0>				481
INTCON	GIE/GIEH	PEIE/GIEL	IPEN	—	—	INT2EDG	INT1EDG	INT0EDG	170
PIR2	HLVDIF	ZCDIF	—	—	—	—	C2IF	C1IF	173
PIE2	HLVDIE	ZCDIE	—	—	—	—	C2IE	C1IE	181
IPR2	HLVDIP	ZCDIP	—	—	—	—	C2IP	C1IP	189
PMD0	SYSCMD	FVRMD	HLVDM	CRCMD	SCANMD	NVMMD	CLKRMD	IOCMD	68

Legend: — = unimplemented, read as '0'. Shaded cells are unused by the HLVD module.

Note 1: PORTA<7:6> and their direction bits are individually configured as port pins based on various primary oscillator modes. When disabled, these bits read as '0'.

PIC18(L)F26/45/46K40

TABLE 35-2: INSTRUCTION SET (CONTINUED)

Mnemonic, Operands		Description	Cycles	16-Bit Instruction Word				Status Affected	Notes
				MSb		LSb			
BIT-ORIENTED OPERATIONS									
BCF	f, b, a	Bit Clear f	1	1001	bbba	ffff	ffff	None	1, 2
BSF	f, b, a	Bit Set f	1	1000	bbba	ffff	ffff	None	1, 2
BTFS	f, b, a	Bit Test f, Skip if Clear	1 (2 or 3)	1011	bbba	ffff	ffff	None	3, 4
BTFS	f, b, a	Bit Test f, Skip if Set	1 (2 or 3)	1010	bbba	ffff	ffff	None	3, 4
BTG	f, b, a	Bit Toggle f	1	0111	bbba	ffff	ffff	None	1, 2
CONTROL OPERATIONS									
BC	n	Branch if Carry	1 (2)	1110	0010	nnnn	nnnn	None	4
BN	n	Branch if Negative	1 (2)	1110	0110	nnnn	nnnn	None	
BNC	n	Branch if Not Carry	1 (2)	1110	0011	nnnn	nnnn	None	
BNN	n	Branch if Not Negative	1 (2)	1110	0111	nnnn	nnnn	None	
BNOV	n	Branch if Not Overflow	1 (2)	1110	0101	nnnn	nnnn	None	
BNZ	n	Branch if Not Zero	1 (2)	1110	0001	nnnn	nnnn	None	
BOV	n	Branch if Overflow	1 (2)	1110	0100	nnnn	nnnn	None	
BRA	n	Branch Unconditionally	2	1101	0nnn	nnnn	nnnn	None	
BZ	n	Branch if Zero	1 (2)	1110	0000	nnnn	nnnn	None	
CALL	k, s	Call subroutine 1st word	2	1110	110s	kkkk	kkkk	None	
		2nd word		1111	kkkk	kkkk	kkkk		
CLRWD	—	Clear Watchdog Timer	1	0000	0000	0000	0100	\overline{TO} , \overline{PD}	
DAW	—	Decimal Adjust WREG	1	0000	0000	0000	0111	C	
GOTO	k	Go to address 1st word	2	1110	1111	kkkk	kkkk	None	
		2nd word		1111	kkkk	kkkk	kkkk		
NOP	—	No Operation	1	0000	0000	0000	0000	None	
NOP	—	No Operation	1	1111	xxxx	xxxx	xxxx	None	
POP	—	Pop top of return stack (TOS)	1	0000	0000	0000	0110	None	
PUSH	—	Push top of return stack (TOS)	1	0000	0000	0000	0101	None	
RCALL	n	Relative Call	2	1101	1nnn	nnnn	nnnn	None	
RESET		Software device Reset	1	0000	0000	1111	1111	All	
RETFIE	s	Return from interrupt enable	2	0000	0000	0001	000s	GIE/GIEH, PEIE/GIEL	
RETLW	k	Return with literal in WREG	2	0000	1100	kkkk	kkkk	None	
RETURN	s	Return from Subroutine	2	0000	0000	0001	001s	None	
SLEEP	—	Go into Standby mode	1	0000	0000	0000	0011	\overline{TO} , \overline{PD}	

- Note 1:** When a PORT register is modified as a function of itself (e.g., `MOVF PORTB, 1, 0`), the value used will be that value present on the pins themselves. For example, if the data latch is '1' for a pin configured as input and is driven low by an external device, the data will be written back with a '0'.
- 2:** If this instruction is executed on the TMR0 register (and where applicable, 'd' = 1), the prescaler will be cleared if assigned.
- 3:** If Program Counter (PC) is modified or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a `NOP`.
- 4:** Some instructions are two-word instructions. The second word of these instructions will be executed as a `NOP` unless the first word of the instruction retrieves the information embedded in these 16 bits. This ensures that all program memory locations have a valid instruction.

PIC18(L)F26/45/46K40

TABLE 37-6: THERMAL CHARACTERISTICS

Standard Operating Conditions (unless otherwise stated)					
Param No.	Sym.	Characteristic	Typ.	Units	Conditions
TH01	θ_{JA}	Thermal Resistance Junction to Ambient	60	°C/W	28-pin SPDIP package
			80	°C/W	28-pin SOIC package
			90	°C/W	28-pin SSOP package
			27.5	°C/W	28-pin UQFN 4x4 mm package
			27.5	°C/W	28-pin QFN 6x6mm package
			47.2	°C/W	40-pin PDIP package
			46	°C/W	44-pin TQFP package
			24.4	°C/W	44-pin QFN 8x8mm package
TH02	θ_{JC}	Thermal Resistance Junction to Case	31.4	°C/W	28-pin SPDIP package
			24	°C/W	28-pin SOIC package
			24	°C/W	28-pin SSOP package
			24	°C/W	28-pin UQFN 4x4mm package
			24	°C/W	28-pin QFN 6x6mm package
			24.7	°C/W	40-pin PDIP package
			14.5	°C/W	44-pin TQFP package
			20	°C/W	44-pin QFN 8x8mm package
TH03	T_{JMAX}	Maximum Junction Temperature	150	°C	
TH04	PD	Power Dissipation	—	W	$PD = P_{INTERNAL} + P_{I/O}^{(3)}$
TH05	$P_{INTERNAL}$	Internal Power Dissipation	—	W	$P_{INTERNAL} = I_{DD} \times V_{DD}^{(1)}$
TH06	$P_{I/O}$	I/O Power Dissipation	—	W	$P_{I/O} = \sum (I_{OL} \times V_{OL}) + \sum (I_{OH} \times (V_{DD} - V_{OH}))$
TH07	P_{DER}	Derated Power	—	W	$P_{DER} = P_{DMAX} (T_J - T_A) / \theta_{JA}^{(2)}$

Note 1: I_{DD} is current to run the chip alone without driving any load on the output pins.

2: T_A = Ambient Temperature, T_J = Junction Temperature

3: See absolute maximum ratings for total power dissipation.

PIC18(L)F26/45/46K40

TABLE 37-11: RESET, WDT, OSCILLATOR START-UP TIMER, POWER-UP TIMER, BROWN-OUT RESET AND LOW-POWER BROWN-OUT RESET SPECIFICATIONS

Standard Operating Conditions (unless otherwise stated)							
Param No.	Sym.	Characteristic	Min.	Typ†	Max.	Units	Conditions
RST01*	TMCLR	MCLR Pulse Width Low to ensure Reset	2	—	—	μs	
RST02*	TIOZ	I/O high-impedance from Reset detection	—	—	2	μs	
RST03	TWDT	Watchdog Timer Time-out Period	—	16	—	ms	1:512 Prescaler
RST04*	TPWRT	Power-up Timer Period	—	65	—	ms	
RST05	TOST	Oscillator Start-up Timer Period ^(1,2)	—	1024	—	Tosc	
RST06	VBOR	Brown-out Reset Voltage ⁽⁴⁾	2.7	2.85	3.0	V	BORV = 00
			2.55	2.7	2.85	V	BORV = 01
			2.3	2.45	2.6	V	BORV = 10
			2.3	2.45	2.6	V	BORV = 11 (PIC18Fxxx)
			1.8	1.9	2.1	V	BORV = 11 (PIC18LFxxx)
RST07	VBORHYS	Brown-out Reset Hysteresis	—	40	—	mV	
RST08	TBORDC	Brown-out Reset Response Time	—	3	—	μs	
RST09	VLPBOR	Low-Power Brown-out Reset Voltage	1.8	1.9	2.5	V	

* These parameters are characterized but not tested.

† 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: By design, the Oscillator Start-up Timer (OST) counts the first 1024 cycles, independent of frequency.

Note 2: To ensure these voltage tolerances, VDD and VSS must be capacitively decoupled as close to the device as possible. 0.1 μF and 0.01 μF values in parallel are recommended.

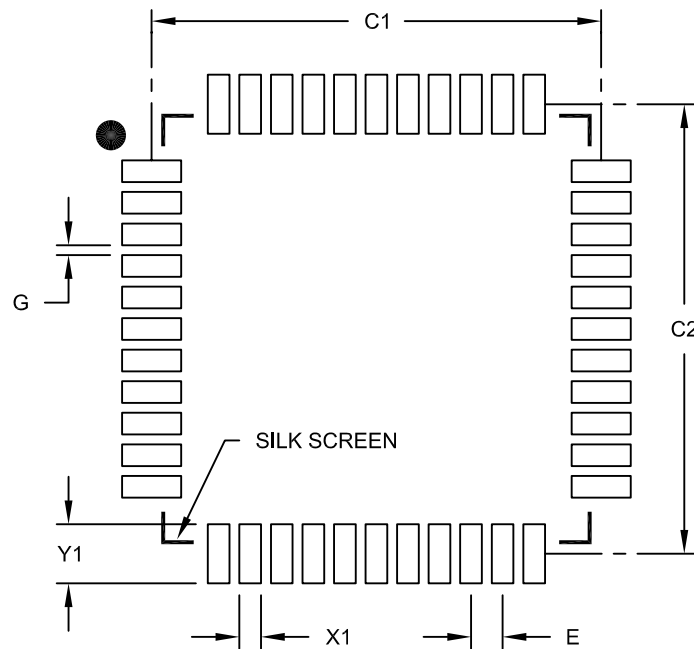
TABLE 37-12: HIGH/LOW-VOLTAGE DETECT CHARACTERISTICS

Standard Operating Conditions (unless otherwise stated)							
Param. No.	Symbol	Characteristic	Min.	Typ†	Max.	Units	Conditions
HLVD01	V _{DET}	Voltage Detection	—	1.90	—	V	HLVDSEL<3:0>=0000
			—	2.10	—	V	HLVDSEL<3:0>=0001
			—	2.25	—	V	HLVDSEL<3:0>=0010
			—	2.50	—	V	HLVDSEL<3:0>=0011
			—	2.60	—	V	HLVDSEL<3:0>=0100
			—	2.75	—	V	HLVDSEL<3:0>=0101
			—	2.90	—	V	HLVDSEL<3:0>=0110
			—	3.15	—	V	HLVDSEL<3:0>=0111
			—	3.35	—	V	HLVDSEL<3:0>=1000
			—	3.60	—	V	HLVDSEL<3:0>=1001
			—	3.75	—	V	HLVDSEL<3:0>=1010
			—	4.00	—	V	HLVDSEL<3:0>=1011
			—	4.20	—	V	HLVDSEL<3:0>=1100
			—	4.35	—	V	HLVDSEL<3:0>=1101
			—	4.65	—	V	HLVDSEL<3:0>=1110

PIC18(L)F26/45/46K40

44-Lead Plastic Thin Quad Flatpack (PT) 10X10X1 mm Body, 2.00 mm Footprint [TQFP]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E	0.80 BSC		
Contact Pad Spacing	C1		11.40	
Contact Pad Spacing	C2		11.40	
Contact Pad Width (X44)	X1			0.55
Contact Pad Length (X44)	Y1			1.50
Distance Between Pads	G	0.25		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2076B