Welcome to E-XFL.COM

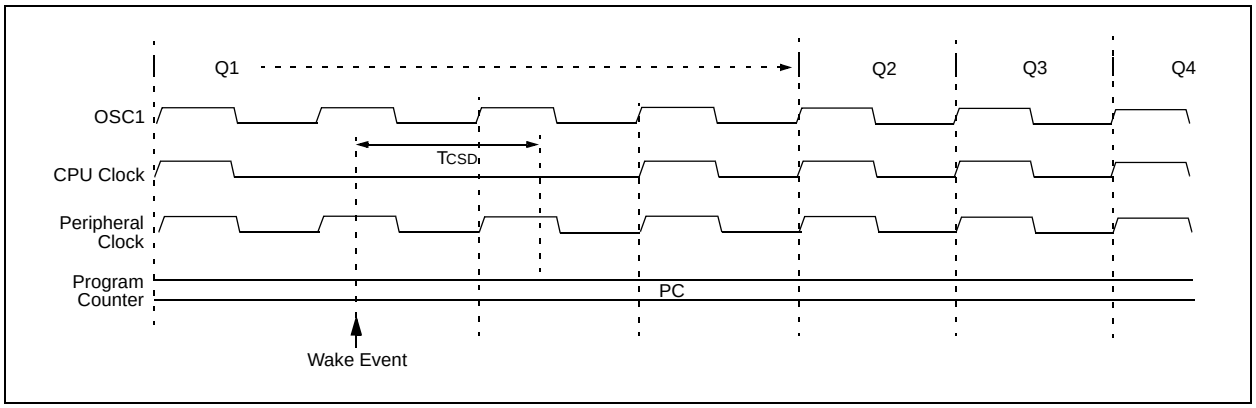
What is "Embedded - Microcontrollers"?

"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	48MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, HLVD, POR, PWM, WDT
Number of I/O	24
Program Memory Size	32KB (16K x 16)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	1.5K x 8
Voltage - Supply (Vcc/Vdd)	2.3V ~ 5.5V
Data Converters	A/D 19x10b
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/pic18f25k22-e-ss

FIGURE 3-7: TRANSITION TIMING FOR WAKE FROM IDLE TO RUN MODE



3.4.3 RC_IDLE MODE

In RC_IDLE mode, the CPU is disabled but the peripherals continue to be clocked from the internal oscillator block from the HFINTOSC multiplexer output. This mode allows for controllable power conservation during Idle periods.

From RC_RUN, this mode is entered by setting the IDLEN bit and executing a SLEEP instruction. If the device is in another Run mode, first set IDLEN, then set the SCS1 bit and execute SLEEP. It is recommended that SCS0 also be cleared, although its value is ignored, to maintain software compatibility with future devices. The HFINTOSC multiplexer may be used to select a higher clock frequency by modifying the IRCF bits before executing the SLEEP instruction. When the clock source is switched to the HFINTOSC multiplexer, the primary oscillator is shut down and the OSTS bit is cleared.

If the IRCF bits are set to any non-zero value, or either the INTSRC or MFIOSEL bits are set, the HFINTOSC output is enabled. Either the HFIOFS or the MFIOFS bits become set, after the HFINTOSC output stabilizes after an interval of TIOBST. For information on the HFIOFS and MFIOFS bits, see Table 3-2.

Clocks to the peripherals continue while the HFINTOSC source stabilizes. The HFIOFS and MFIOFS bits will remain set if the IRCF bits were previously set at a non-zero value or if INTSRC was set before the SLEEP instruction was executed and the HFINTOSC source was already stable. If the IRCF bits and INTSRC are all clear, the HFINTOSC output will not be enabled, the HFIOFS and MFIOFS bits will remain clear and there will be no indication of the current clock source.

When a wake event occurs, the peripherals continue to be clocked from the HFINTOSC multiplexer output. After a delay of TCSD following the wake event, the CPU begins executing code being clocked by the HFINTOSC multiplexer. The IDLEN and SCS bits are not affected by the wake-up. The LFINTOSC source will continue to run if either the WDT or the Fail-Safe Clock Monitor is enabled.

5.4.2 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 Block 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 (Figures 5-5 through 5-7).

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 5.7.3 “Mapping the Access Bank in Indexed Literal Offset Mode”**.

5.4.3 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.

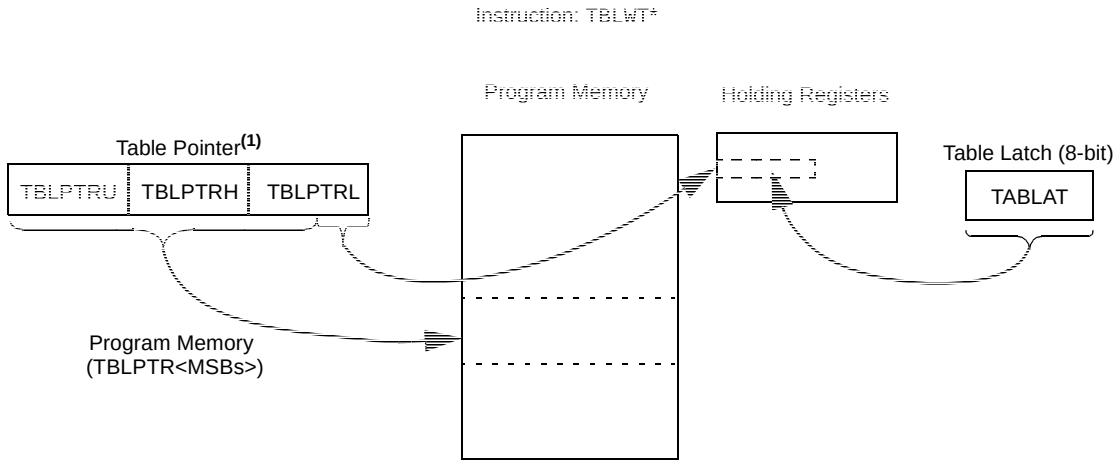
5.4.4 SPECIAL FUNCTION REGISTERS

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 5-1 and Table 5-2.

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.

FIGURE 6-2: TABLE WRITE OPERATION



Note 1: During table writes the Table Pointer does not point directly to Program Memory. The LSBs of TBLPRTL actually point to an address within the write block holding registers. The MSBs of the Table Pointer determine where the write block will eventually be written. The process for writing the holding registers to the program memory array is discussed in **Section 6.6 "Writing to Flash Program Memory"**.

6.2 Control Registers

Several control registers are used in conjunction with the TBLRD and TBLWT instructions. These include the:

- EECON1 register
- EECON2 register
- TABLAT register
- TBLPTR registers

6.2.1 EECON1 AND EECON2 REGISTERS

The EECON1 register (Register 6-1) is the control register for memory accesses. The EECON2 register is not a physical register; it is used exclusively in the memory write and erase sequences. Reading EECON2 will read all '0's.

The EEPGD control bit determines if the access will be a program or data EEPROM memory access. When EEPGD is clear, any subsequent operations will operate on the data EEPROM memory. When EEPGD is set, any subsequent operations will operate on the program memory.

The CFGS control bit determines if the access will be to the Configuration/Calibration registers or to program memory/data EEPROM memory. When CFGS is set, subsequent operations will operate on Configuration registers regardless of EEPGD (see **Section 24.0 "Special Features of the CPU"**). When CFGS is clear, memory selection access is determined by EEPGD.

The FREE bit allows the program memory erase operation. When FREE is set, an erase operation is initiated on the next WR command. When FREE is clear, only writes are enabled.

The WREN bit, when set, will allow a write operation. The WREN bit is clear on power-up.

The WRERR bit is set by hardware when the WR bit is set and cleared when the internal programming timer expires and the write operation is complete.

Note:	During normal operation, the WRERR is read as '1'. This can indicate that a write operation was prematurely terminated by a Reset, or a write operation was attempted improperly.

The WR control bit initiates write operations. The WR bit cannot be cleared, only set, by firmware. Then WR bit is cleared by hardware at the completion of the write operation.

Note:	The EEIF interrupt flag bit of the PIR2 register is set when the write is complete. The EEIF flag stays set until cleared by firmware.

7.0 DATA EEPROM MEMORY

The data EEPROM is a nonvolatile memory array, separate from the data RAM and program memory, which is used for long-term storage of program data. It is not directly mapped in either the register file or program memory space but is indirectly addressed through the Special Function Registers (SFRs). The EEPROM is readable and writable during normal operation over the entire VDD range.

Four SFRs are used to read and write to the data EEPROM as well as the program memory. They are:

- EECON1
- EECON2
- EEDATA
- EEADR
- EEADRH

The data EEPROM allows byte read and write. When interfacing to the data memory block, EEDATA holds the 8-bit data for read/write and the EEADR:EEADRH register pair hold the address of the EEPROM location being accessed.

The EEPROM data memory is rated for high erase/write cycle endurance. A byte write automatically erases the location and writes the new data (erase-before-write). The write time is controlled by an on-chip timer; it will vary with voltage and temperature as well as from chip-to-chip. Please refer to the Data EEPROM Memory parameters in **Section 27.0 "Electrical Specifications"** for limits.

7.1 EEADR and EEADRH Registers

The EEADR register is used to address the data EEPROM for read and write operations. The 8-bit range of the register can address a memory range of 256 bytes (00h to FFh). The EEADRH register expands the range to 1024 bytes by adding an additional two address bits.

7.2 EECON1 and EECON2 Registers

Access to the data EEPROM is controlled by two registers: EECON1 and EECON2. These are the same registers which control access to the program memory and are used in a similar manner for the data EEPROM.

The EECON1 register (Register 7-1) is the control register for data and program memory access. Control bit EEPGD determines if the access will be to program or data EEPROM memory. When the EEPGD bit is clear, operations will access the data EEPROM memory. When the EEPGD bit is set, program memory is accessed.

Control bit, CFGS, determines if the access will be to the Configuration registers or to program memory/data EEPROM memory. When the CFGS bit is set, subsequent operations access Configuration registers. When the CFGS bit is clear, the EEPGD bit selects either program Flash or data EEPROM memory.

The WREN bit, when set, will allow a write operation. On power-up, the WREN bit is clear.

The WRERR bit is set by hardware when the WR bit is set and cleared when the internal programming timer expires and the write operation is complete.

Note:	During normal operation, the WRERR may read as '1'. This can indicate that a write operation was prematurely terminated by a Reset, or a write operation was attempted improperly.

The WR control bit initiates write operations. The bit can be set but not cleared by software. It is cleared only by hardware at the completion of the write operation.

Note:	The EEIF interrupt flag bit of the PIR2 register is set when the write is complete. It must be cleared by software.

Control bits, RD and WR, start read and erase/write operations, respectively. These bits are set by firmware and cleared by hardware at the completion of the operation.

The RD bit cannot be set when accessing program memory (EEPGD = 1). Program memory is read using table read instructions. See **Section 6.1 "Table Reads and Table Writes"** regarding table reads.

The EECON2 register is not a physical register. It is used exclusively in the memory write and erase sequences. Reading EECON2 will read all '0's.

9.0 INTERRUPTS

The PIC18(L)F2X/4XK22 devices have multiple interrupt sources and an interrupt priority feature that allows most interrupt sources to be assigned a high or low priority level (INT0 does not have a priority bit, it is always a high priority). The high priority interrupt vector is at 0008h and the low priority interrupt vector is at 0018h. A high priority interrupt event will interrupt a low priority interrupt that may be in progress.

There are 19 registers used to control interrupt operation.

These registers are:

- INTCON, INTCON2, INTCON3
- PIR1, PIR2, PIR3, PIR4, PIR5
- PIE1, PIE2, PIE3, PIE4, PIE5
- IPR1, IPR2, IPR3, IPR4, IPR5
- RCON

It is recommended that the Microchip header files supplied with MPLAB® IDE be used for the symbolic bit names in these registers. This allows the assembler/ compiler to automatically take care of the placement of these bits within the specified register.

In general, interrupt sources have three bits to control their operation. They are:

- **Flag bit** to indicate that an interrupt event occurred
- **Enable bit** that allows program execution to branch to the interrupt vector address when the flag bit is set
- **Priority bit** to select high priority or low priority

9.1 Mid-Range Compatibility

When the IPEN bit is cleared (default state), the interrupt priority feature is disabled and interrupts are compatible with PIC® microcontroller mid-range devices. In Compatibility mode, the interrupt priority bits of the IPRx registers have no effect. The PEIE/GIEL bit of the INTCON register is the global interrupt enable for the peripherals. The PEIE/GIEL bit disables only the peripheral interrupt sources and enables the peripheral interrupt sources when the GIE/GIEH bit is also set. The GIE/GIEH bit of the INTCON register is the global interrupt enable which enables all non-peripheral interrupt sources and disables all interrupt sources, including the peripherals. All interrupts branch to address 0008h in Compatibility mode.

9.2 Interrupt Priority

The interrupt priority feature is enabled by setting the IPEN bit of the RCON register. When interrupt priority is enabled the GIE/GIEH and PEIE/GIEL global interrupt enable bits of Compatibility mode are replaced by the GIEH high priority, and GIEL low priority, global interrupt enables. When set, the GIEH bit of the INTCON register enables all interrupts that have their associated IPRx register or INTCONx register priority bit set (high priority). When clear, the GIEH bit disables all interrupt sources including those selected as low priority. When clear, the GIEL bit of the INTCON register disables only the interrupts that have their associated priority bit cleared (low priority). When set, the GIEL bit enables the low priority sources when the GIEH bit is also set.

When the interrupt flag, enable bit and appropriate Global Interrupt Enable (GIE) bit are all set, the interrupt will vector immediately to address 0008h for high priority, or 0018h for low priority, depending on level of the interrupting source's priority bit. Individual interrupts can be disabled through their corresponding interrupt enable bits.

9.3 Interrupt Response

When an interrupt is responded to, the Global Interrupt Enable bit is cleared to disable further interrupts. The GIE/GIEH bit is the Global Interrupt Enable when the IPEN bit is cleared. When the IPEN bit is set, enabling interrupt priority levels, the GIEH bit is the high priority global interrupt enable and the GIEL bit is the low priority Global Interrupt Enable. High priority interrupt sources can interrupt a low priority interrupt. Low priority interrupts are not processed while high priority interrupts are in progress.

The return address is pushed onto the stack and the PC is loaded with the interrupt vector address (0008h or 0018h). Once in the Interrupt Service Routine, the source(s) of the interrupt can be determined by polling the interrupt flag bits in the INTCONx and PIRx registers. The interrupt flag bits must be cleared by software before re-enabling interrupts to avoid repeating the same interrupt.

The "return from interrupt" instruction, `RETFIE`, exits the interrupt routine and sets the GIE/GIEH bit (GIEH or GIEL if priority levels are used), which re-enables interrupts.

For external interrupt events, such as the INT pins or the PORTB interrupt-on-change, the interrupt latency will be three to four instruction cycles. The exact latency is the same for one-cycle or two-cycle instructions. Individual interrupt flag bits are set, regardless of the status of their corresponding enable bits or the Global Interrupt Enable bit.

TABLE 14-10: REGISTERS ASSOCIATED WITH STANDARD PWM

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
CCP1CON	P1M<1:0>		DC1B<1:0>		CCP1M<3:0>				198
CCP2CON	P2M<1:0>		DC2B<1:0>		CCP2M<3:0>				198
CCP3CON	P3M<1:0>		DC3B<1:0>		CCP3M<3:0>				198
CCP4CON	—	—	DC4B<1:0>		CCP4M<3:0>				198
CCP5CON	—	—	DC5B<1:0>		CCP5M<3:0>				198
CCPTMRS0	C3TSEL<1:0>		—	C2TSEL<1:0>		—	C1TSEL<1:0>		201
CCPTMRS1	—	—	—	—	C5TSEL<1:0>		C4TSEL<1:0>		201
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	109
IPR1	—	ADIP	RC1IP	TX1IP	SSP1IP	CCP1IP	TMR2IP	TMR1IP	121
IPR2	OSCFIP	C1IP	C2IP	EEIP	BCL1IP	HLVDIP	TMR3IP	CCP2IP	122
IPR4	—	—	—	—	—	CCP5IP	CCP4IP	CCP3IP	124
PIE1	—	ADIE	RC1IE	TX1IE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	117
PIE2	OSCFIE	C1IE	C2IE	EEIE	BCL1IE	HLVDIE	TMR3IE	CCP2IE	118
PIE4	—	—	—	—	—	CCP5IE	CCP4IE	CCP3IE	120
PIR1	—	ADIF	RC1IF	TX1IF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	112
PIR2	OSCFIF	C1IF	C2IF	EEIF	BCL1IF	HLVDIF	TMR3IF	CCP2IF	113
PIR4	—	—	—	—	—	CCP5IF	CCP4IF	CCP3IF	115
PMD0	UART2MD	UART1MD	TMR6MD	TMR5MD	TMR4MD	TMR3MD	TMR2MD	TMR1MD	52
PMD1	MSSP2MD	MSSP1MD	—	CCP5MD	CCP4MD	CCP3MD	CCP2MD	CCP1MD	53
PR2	Timer2 Period Register								—
PR4	Timer4 Period Register								—
PR6	Timer6 Period Register								—
T2CON	—	T2OUTPS<3:0>				TMR2ON	T2CKPS<1:0>		166
T4CON	—	T4OUTPS<3:0>				TMR4ON	T4CKPS<1:0>		166
T6CON	—	T6OUTPS<3:0>				TMR6ON	T6CKPS<1:0>		166
TMR2	Timer2 Register								—
TMR4	Timer4 Register								—
TMR6	Timer6 Register								—
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	151
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	151
TRISD(1)	TRISD7	TRISD6	TRISD5	TRISD4	TRISD3	TRISD2	TRISD1	TRISD0	151
TRISE	WPUE3	—	—	—	—	TRISE2(1)	TRISE1(1)	TRISE0(1)	151

Legend: — = Unimplemented location, read as '0'. Shaded bits are not used by Standard PWM mode.

Note 1: These registers/bits are available on PIC18(L)F4XK22 devices.

TABLE 14-11: CONFIGURATION REGISTERS ASSOCIATED WITH STANDARD PWM

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
CONFIG3H	MCLRE	—	P2BMX	T3CMX	HFOFST	CCP3MX	PBADEN	CCP2MX	348

Legend: — = Unimplemented location, read as '0'. Shaded bits are not used by Standard PWM mode.

FIGURE 15-17: I²C SLAVE, 7-BIT ADDRESS, RECEPTION (SEN = 1, AHEN = 1, DHEN = 1)

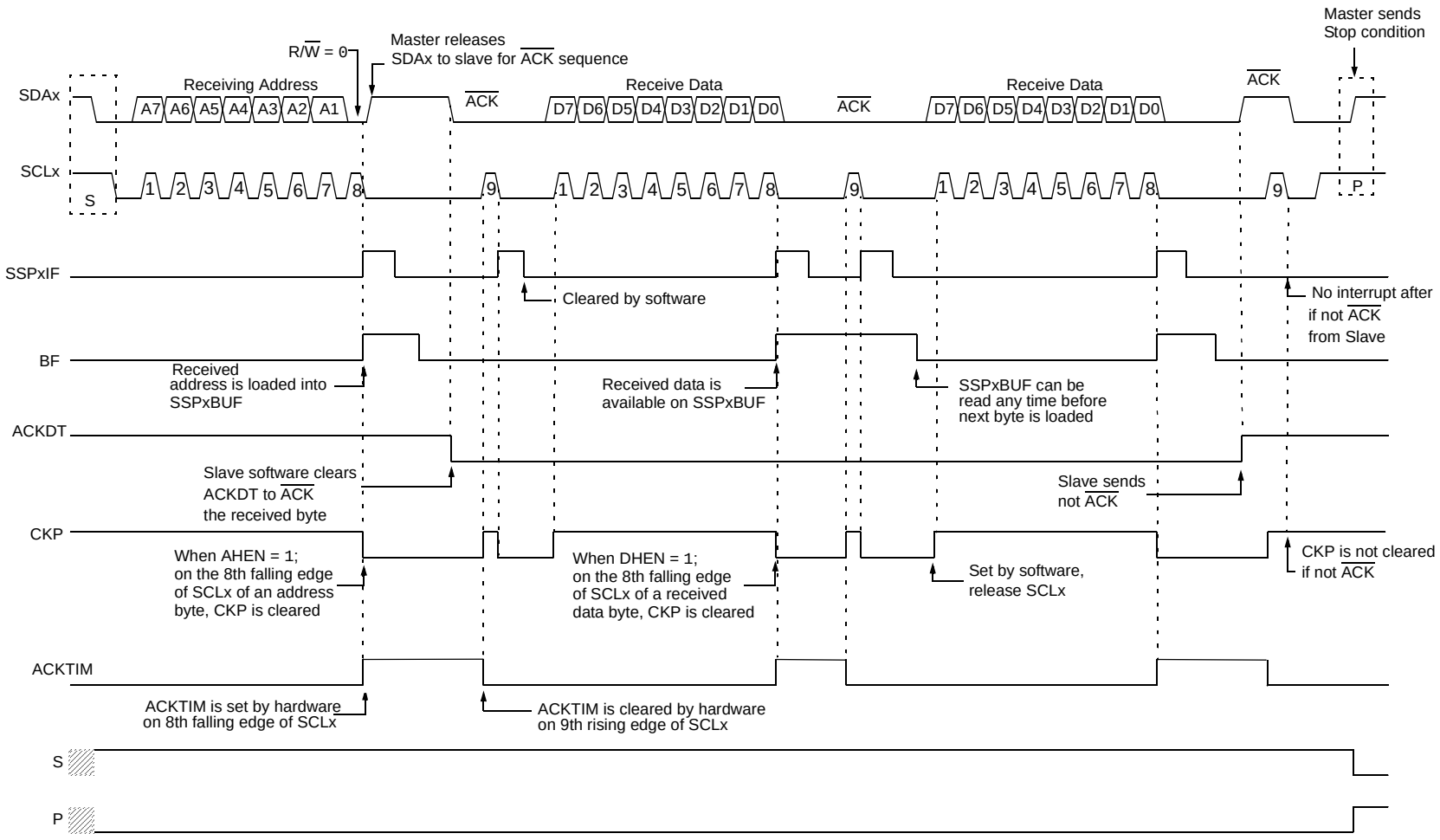


TABLE 15-2: REGISTERS ASSOCIATED WITH I^2C OPERATION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ANSELA	—	—	ANSA5	—	ANSA3	ANSA2	ANSA1	ANSA0	149
ANSELB	—	—	ANSB5	ANSB4	ANSB3	ANSB2	ANSB1(1)	ANSB0(1)	150
ANSELC	ANSC7	ANSC6	ANSC5	ANSC4	ANSC3	ANSC2	—	—	150
ANSELD	ANSD7	ANSD6	ANSD5	ANSD4	ANSD3	ANSD2	ANSD1(2)	ANSD0(2)	150
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	109
IPR1	—	ADIP	RC1IP	TX1IP	SSP1IP	CCP1IP	TMR2IP	TMR1IP	121
IPR2	OSCFIP	C1IP	C2IP	EEIP	BCL1IP	HLVDIP	TMR3IP	CCP2IP	122
IPR3	SSP2IP	BCL2IP	RC2IP	TX2IP	CTMUIP	TMR5GIP	TMR3GIP	TMR1GIP	123
PIE1	—	ADIE	RC1IE	TX1IE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	117
PIE2	OSCFIE	C1IE	C2IE	EEIE	BCL1IE	HLVDIE	TMR3IE	CCP2IE	118
PIE3	SSP2IE	BCL2IE	RC2IE	TX2IE	CTMUIE	TMR5GIE	TMR3GIE	TMR1GIE	119
PIR1	—	ADIF	RC1IF	TX1IF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	112
PIR2	OSCFIF	C1IF	C2IF	EEIF	BCL1IF	HLVDIF	TMR3IF	CCP2IF	113
PIR3	SSP2IF	BCL2IF	RC2IF	TX2IF	CTMUIF	TMR5GIF	TMR3GIF	TMR1GIF	114
PMD1	MSSP2MD	MSSP1MD	—	CCP5MD	CCP4MD	CCP3MD	CCP2MD	CCP1MD	53
SSP1ADD	SSP1 Address Register in I^2C Slave mode. SSP1 Baud Rate Reload Register in I^2C Master mode.								258
SSP1BUF	SSP1 Receive Buffer/Transmit Register								—
SSP1CON1	WCOL	SSPOV	SSPEN	CKP	SSPM<3:0>				253
SSP1CON2	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	255
SSP1CON3	ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	256
SSP1MSK	SSP1 MASK Register bits								257
SSP1STAT	SMP	CKE	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF	252
SSP2ADD	SSP2 Address Register in I^2C Slave mode. SSP2 Baud Rate Reload Register in I^2C Master mode.								258
SSP2BUF	SSP2 Receive Buffer/Transmit Register								—
SSP2CON1	WCOL	SSPOV	SSPEN	CKP	SSPM<3:0>				253
SSP2CON2	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	255
SSP2CON3	ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	256
SSP2MSK	SSP1 MASK Register bits								257
SSP2STAT	SMP	CKE	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF	252
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1(1)	TRISB0(1)	151
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	151
TRISD	TRISD7	TRISD6	TRISD5	TRISD4	TRISD3	TRISD2	TRISD1(2)	TRISD0(2)	151

Legend: Shaded bits are not used by the MSSPx in I^2C mode.

Note 1: PIC18(L)F2XK22 devices.

2: PIC18(L)F4XK22 devices.

REGISTER 15-4: SSPxCON2: SSPx CONTROL REGISTER 2

R/W-0	R-0	R/W-0	R/S/HC-0	R/S/HC-0	R/S/HC-0	R/S/HC-0	R/W/HC-0
GCEN	ACKSTAT	ACKDT	ACKEN(1)	RCEN(1)	PEN(1)	RSEN(1)	SEN(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	HC = Cleared by hardware S = User set

bit 7 **GCEN:** General Call Enable bit (in I^2C Slave mode only)
1 = Enable interrupt when a general call address (0x00 or 00h) is received in the SSPxSR
0 = General call address disabled

bit 6 **ACKSTAT:** Acknowledge Status bit (in I^2C mode only)
1 = Acknowledge was not received
0 = Acknowledge was received

bit 5 **ACKDT:** Acknowledge Data bit (in I^2C mode only)
In Receive mode:
Value transmitted when the user initiates an Acknowledge sequence at the end of a receive
1 = Not Acknowledge
0 = Acknowledge

bit 4 **ACKEN(1):** Acknowledge Sequence Enable bit (in I^2C Master mode only)
In Master Receive mode:
1 = Initiate Acknowledge sequence on SDAx and SCLx pins, and transmit ACKDT data bit. Automatically cleared by hardware.
0 = Acknowledge sequence idle

bit 3 **RCEN(1):** Receive Enable bit (in I^2C Master mode only)
1 = Enables Receive mode for I^2C
0 = Receive idle

bit 2 **PEN(1):** Stop Condition Enable bit (in I^2C Master mode only)
SCKx Release Control:
1 = Initiate Stop condition on SDAx and SCLx pins. Automatically cleared by hardware.
0 = Stop condition Idle

bit 1 **RSEN(1):** Repeated Start Condition Enabled bit (in I^2C Master mode only)
1 = Initiate Repeated Start condition on SDAx and SCLx pins. Automatically cleared by hardware.
0 = Repeated Start condition Idle

bit 0 **SEN(1):** Start Condition Enabled bit (in I^2C Master mode only)
In Master mode:
1 = Initiate Start condition on SDAx and SCLx pins. Automatically cleared by hardware.
0 = Start condition Idle
In Slave mode:
1 = Clock stretching is enabled for both slave transmit and slave receive (stretch enabled)
0 = Clock stretching is disabled

Note 1: For bits ACKEN, RCEN, PEN, RSEN, SEN: If the I^2C module is not in the Idle mode, this bit may not be set (no spooling) and the SSPxBUF may not be written (or writes to the SSPxBUF are disabled).

REGISTER 15-5: SSPxCON3: SSPx CONTROL REGISTER 3

R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN
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 **ACKTIM:** Acknowledge Time Status bit (I^2C mode only)**(3)**

1 = Indicates the I^2C bus is in an Acknowledge sequence, set on 8th falling edge of SCLx clock
0 = Not an Acknowledge sequence, cleared on 9th rising edge of SCLx clock

bit 6 **PCIE**: Stop Condition Interrupt Enable bit (I^2C mode only)

1 = Enable interrupt on detection of Stop condition
0 = Stop detection interrupts are disabled**(2)**

bit 5 **SCIE**: Start Condition Interrupt Enable bit (I^2C mode only)

1 = Enable interrupt on detection of Start or Restart conditions
0 = Start detection interrupts are disabled**(2)**

bit 4 **BOEN:** Buffer Overwrite Enable bit

In SPI Slave mode:**(1)**

1 = SSPxBUF updates every time that a new data byte is shifted in ignoring the BF bit
0 = If new byte is received with BF bit of the SSPxSTAT register already set, SSPxOV bit of the SSPxCON1 register is set, and the buffer is not updated

In I^2C Master mode:

This bit is ignored.

In I^2C Slave mode:

1 = SSPxBUF is updated and $\overline{ACK}$ is generated for a received address/data byte, ignoring the state of the SSPxOV bit only if the BF bit = 0.
0 = SSPxBUF is only updated when SSPxOV is clear

bit 3 **SDAHT:** SDAx Hold Time Selection bit (I^2C mode only)

1 = Minimum of 300 ns hold time on SDAx after the falling edge of SCLx
0 = Minimum of 100 ns hold time on SDAx after the falling edge of SCLx

bit 2 **SBCDE:** Slave Mode Bus Collision Detect Enable bit (I^2C Slave mode only)

If on the rising edge of SCLx, SDAx is sampled low when the module is outputting a high state, the BCLxIF bit of the PIR2 register is set, and bus goes idle

1 = Enable slave bus collision interrupts
0 = Slave bus collision interrupts are disabled

bit 1 **AHEN:** Address Hold Enable bit (I^2C Slave mode only)

1 = Following the 8th falling edge of SCLx for a matching received address byte; CKP bit of the SSPxCON1 register will be cleared and the SCLx will be held low.
0 = Address holding is disabled

Note 1: For daisy-chained SPI operation; allows the user to ignore all but the last received byte. SSPxOV is still set when a new byte is received and BF = 1, but hardware continues to write the most recent byte to SSPxBUF.

2: This bit has no effect in Slave modes for which Start and Stop condition detection is explicitly listed as enabled.

3: The ACKTIM Status bit is active only when the AHEN bit or DHEN bit is set.

EXAMPLE 19-2: CURRENT CALIBRATION ROUTINE

```
#include "p18cxxx.h"

#define COUNT 500                           //@ 8MHz = 125uS.
#define DELAY for(i=0;i<COUNT;i++)
#define RCAL .027                           //R value is 4200000 (4.2M)
                                            //scaled so that result is in
                                            //1/100th of uA
#define ADSCALE 1023                        //for unsigned conversion 10 sig bits
#define ADREF 3.3                           //Vdd connected to A/D Vr+

int main(void)
{
    int i;
    int j = 0;                              //index for loop
    unsigned int Vread = 0;
    double VTot = 0;
    float Vavg=0, Vcal=0, CTMUISrc = 0;     //float values stored for calcs

//assume CTMU and A/D have been set up correctly
//see Example 25-1 for CTMU & A/D setup
setup();

CTMUCONHbits.CTMUEN = 1;                    //Enable the CTMU
CTMUCONLbits.EDG1STAT = 0;                  // Set Edge status bits to zero
CTMUCONLbits.EDG2STAT = 0;
    for(j=0;j<10;j++)
    {
        CTMUCONHbits.IDISSEN = 1;           //drain charge on the circuit
        DELAY;                              //wait 125us
        CTMUCONHbits.IDISSEN = 0;           //end drain of circuit

        CTMUCONLbits.EDG1STAT = 1;          //Begin charging the circuit
                                            //using CTMU current source
        DELAY;                              //wait for 125us
        CTMUCONLbits.EDG1STAT = 0;          //Stop charging circuit

        PIR1bits.ADIF = 0;                  //make sure A/D Int not set
        ADCON0bits.GO=1;                    //and begin A/D conv.
        while(!PIR1bits.ADIF);              //Wait for A/D convert complete

        Vread = ADRES;                      //Get the value from the A/D
        PIR1bits.ADIF = 0;                  //Clear A/D Interrupt Flag
        VTot += Vread;                      //Add the reading to the total
    }

    Vavg = (float)(VTot/10.000);            //Average of 10 readings
    Vcal = (float)(Vavg/ADSCALE*ADREF);
    CTMUISrc = Vcal/RCAL;                   //CTMUISrc is in 1/100ths of uA

}
```

19.6 Creating a Delay with the CTMU Module

A unique feature on board the CTMU module is its ability to generate system clock independent output pulses based on an external capacitor value. This is accomplished using the internal comparator voltage reference module, Comparator 2 input pin and an external capacitor. The pulse is output onto the CTPLS pin. To enable this mode, set the TGEN bit.

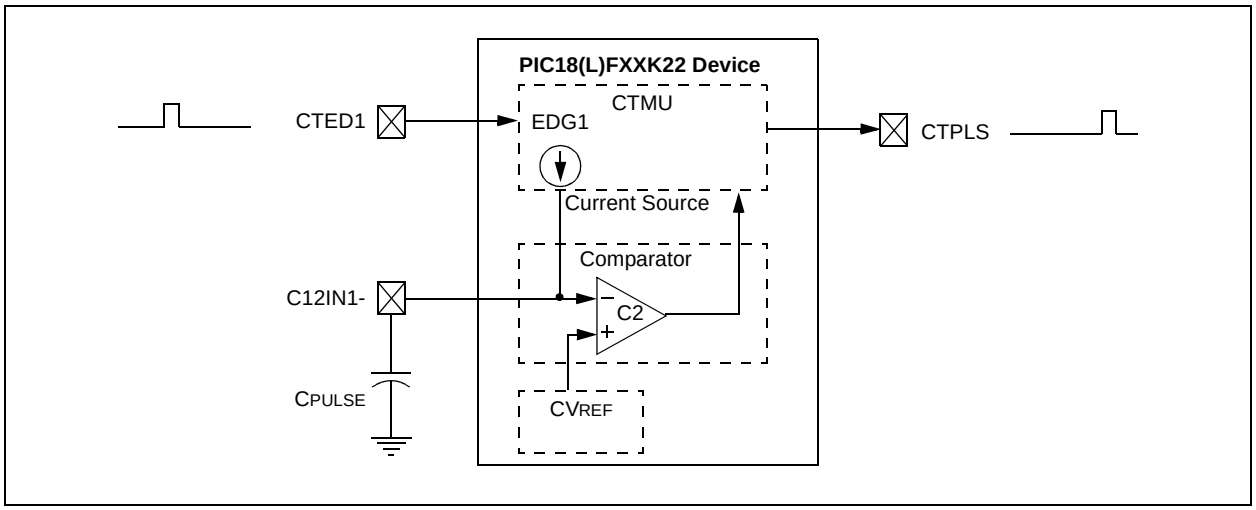
See Figure 19-4 for an example circuit. C_{PULSE} is chosen by the user to determine the output pulse width on CTPLS. The pulse width is calculated by $T = (C_{PULSE}/I) * V$, where I is known from the current source measurement step (**Section 19.3.1 "Current Source Calibration"**) and V is the internal reference voltage (CV_{REF}).

An example use of this feature is for interfacing with variable capacitive-based sensors, such as a humidity sensor. As the humidity varies, the pulse width output on CTPLS will vary. The CTPLS output pin can be connected to an input capture pin and the varying pulse width is measured to determine the humidity in the application.

Follow these steps to use this feature:

1. Initialize Comparator 2.
2. Initialize the comparator voltage reference.
3. Initialize the CTMU and enable time delay generation by setting the TGEN bit.
4. Set EDG1STAT.
5. When C_{PULSE} charges to the value of the voltage reference trip point, an output pulse is generated on CTPLS.

FIGURE 19-4: TYPICAL CONNECTIONS AND INTERNAL CONFIGURATION FOR PULSE DELAY GENERATION



19.7 Operation During Sleep/Idle Modes

19.7.1 SLEEP MODE AND DEEP SLEEP MODES

When the device enters any Sleep mode, the CTMU module current source is always disabled. If the CTMU is performing an operation that depends on the current source when Sleep mode is invoked, the operation may not terminate correctly. Capacitance and time measurements may return erroneous values.

19.7.2 IDLE MODE

The behavior of the CTMU in Idle mode is determined by the CTMUSIDL bit (CTMUCONH<5>). If CTMUSIDL is cleared, the module will continue to operate in Idle mode. If CTMUSIDL is set, the module's current source is disabled when the device enters Idle mode. If the module is performing an operation when Idle mode is invoked, in this case, the results will be similar to those with Sleep mode.

19.8 CTMU Peripheral Module Disable (PMD)

When this peripheral is not used, the Peripheral Module Disable bit can be set to disconnect all clock sources to the module, reducing power consumption to an absolute minimum. See **Section 3.6 "Selective Peripheral Module Control"**.

SUBWFB **Subtract W from f with Borrow**

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

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

Operation: (f) – (W) – $(\overline{C})$ → dest

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

Encoding:

0101	10da	ffff	ffff

Description: Subtract W and the CARRY flag (borrow) from register 'f' (2's complement method). If 'd' is '0', the result is stored in W. If 'd' is '1', the result is stored back in register 'f' (default).
If 'a' is '0', the Access Bank is selected. If 'a' is '1', the BSR is used to select the GPR bank.
If 'a' is '0' and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever f ≤ 95 (5Fh). See **Section 25.2.3 "Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode"** for details.

Words: 1

Cycles: 1

Q Cycle Activity:

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

Example 1:
```
SUBWFB  REG, 1, 0
```
Before Instruction
```
REG  =  19h   (0001 1001)
W    =  0Dh   (0000 1101)
C    =  1
```
After Instruction
```
REG  =  0Ch   (0000 1100)
W    =  0Dh   (0000 1101)
C    =  1
Z    =  0
N    =  0     ; result is positive
```
Example 2:
```
SUBWFB  REG, 0, 0
```
Before Instruction
```
REG  =  1Bh   (0001 1011)
W    =  1Ah   (0001 1010)
C    =  0
```
After Instruction
```
REG  =  1Bh   (0001 1011)
W    =  00h
C    =  1
Z    =  1     ; result is zero
N    =  0
```
Example 3:
```
SUBWFB  REG, 1, 0
```
Before Instruction
```
REG  =  03h   (0000 0011)
W    =  0Eh   (0000 1110)
C    =  1
```
After Instruction
```
REG  =  F5h   (1111 0101)
                ; [2's comp]
W    =  0Eh   (0000 1110)
C    =  0
Z    =  0
N    =  1     ; result is negative
```

SWAPF **Swap f**

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

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

Operation: (f<3:0>) → dest<7:4>,
(f<7:4>) → dest<3:0>

Status Affected: None

Encoding:

0011	10da	ffff	ffff

Description: The upper and lower nibbles of register 'f' are exchanged. If 'd' is '0', the result is placed in W. If 'd' is '1', the result is placed in register 'f' (default).
If 'a' is '0', the Access Bank is selected. If 'a' is '1', the BSR is used to select the GPR bank.
If 'a' is '0' and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever f ≤ 95 (5Fh). See **Section 25.2.3 "Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode"** for details.

Words: 1

Cycles: 1

Q Cycle Activity:

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

Example:
```
SWAPF   REG, 1, 0
```
Before Instruction
```
REG  =  53h
```
After Instruction
```
REG  =  35h
```

27.2 DC Characteristics: Power-Down Current, PIC18(L)F2X/4XK22 (Continued)

PIC18LF2X/4XK22		Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +125°C						
PIC18F2X/4XK22		**Standard Operating Conditions (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +125°C						
Param No.	**Device Characteristics**	**Typ +25°C**	**Typ +60°C**	**Max +85°C**	**Max +125°C**	**Units**	**Conditions**	
							VDD	**Notes**
D015	Comparators	7	7	18	18	µA	1.8V	LP mode
		7	7	18	18	µA	3.0V	
		7	7	18	18	µA	2.3V	
		7	7	18	18	µA	3.0V	
		8	8	20	20	µA	5.0V	
D016	Comparators	38	38	95	95	µA	1.8V	HP mode
		40	40	105	105	µA	3.0V	
		39	39	95	95	µA	2.3V	
		40	40	105	105	µA	3.0V	
		40	40	105	105	µA	5.0V	
D017	DAC	14	14	25	25	µA	2.0V	
		20	20	35	35	µA	3.0V	
		15	15	30	30	µA	2.3V	
		20	20	35	35	µA	3.0V	
		32	32	60	60	µA	5.0V	
D018	FVR(2)	15	16	25	25	µA	1.8V	
		15	16	25	25	µA	3.0V	
		28	28	45	45	µA	2.3V	
		31	31	55	55	µA	3.0V	
		66	66	100	100	µA	5.0V	
D013	A/D Converter(3)	185	185	370	370	µA	1.8V	A/D on, not converting
		210	210	400	400	µA	3.0V	
		200	200	380	380	µA	2.3V	
		210	210	400	400	µA	3.0V	
		250	250	450	450	µA	5.0V	

Note 1: The power-down current in Sleep mode does not depend on the oscillator type. Power-down current is measured with the part in Sleep mode, with all I/O pins in high-impedance state and tied to VDD or VSS and all features that add delta current disabled (such as WDT, Timer1 Oscillator, BOR, etc.).

2: On PIC18LF2X/4XK22 the BOR, HLVD and FVR enable internal band gap reference. With more than one of these modules enabled, the current consumption will be less than the sum of the specifications. On PIC18F2X/4XK22, the internal band gap reference is always enabled and its current consumption is included in the Power-down Base Current (IPD).

3: A/D converter differential currents apply only in Run mode. In Sleep or Idle mode both the ADC and the FRC turn off as soon as conversion (if any) is complete.

27.4 DC Characteristics: RC Idle Supply Current, PIC18(L)F2X/4XK22

PIC18LF2X/4XK22		Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +125°C					
PIC18F2X/4XK22		**Standard Operating Conditions (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +125°C					
Param No.	**Device Characteristics**	**Typ**	**Max**	**Units**	**Conditions**		
D045	Supply Current (IDD)(1),(2)	0.5	18	µA	-40°C	VDD = 1.8V	FOSC = 31 kHz (**RC_IDLE** mode, LFINTOSC source)
		0.6	18	µA	+25°C		
		0.7	—	µA	+60°C		
		0.75	20	µA	+85°C		
		2.3	22	µA	+125°C		
D046		1.1	20	µA	-40°C	VDD = 3.0V	
		1.2	20	µA	+25°C		
		1.3	—	µA	+60°C		
		1.4	22	µA	+85°C		
		3.2	25	µA	+125°C		
D047		17	30	µA	-40°C	VDD = 2.3V	FOSC = 31 kHz (**RC_IDLE** mode, LFINTOSC source)
		13	30	µA	+25°C		
		14	30	µA	+85°C		
		15	45	µA	+125°C		
D048		19	35	µA	-40°C	VDD = 3.0V	
		15	35	µA	+25°C		
		16	35	µA	+85°C		
		17	50	µA	+125°C		
D049		21	40	µA	-40°C	VDD = 5.0V	
		15	40	µA	+25°C		
		16	40	µA	+85°C		
		18	60	µA	+125°C		
D050		0.11	0.20	mA	-40°C to +125°C	VDD = 1.8V	FOSC = 500 kHz (**RC_IDLE** mode, MFINTOSC source)
D051		0.12	0.25	mA	-40°C to +125°C	VDD = 3.0V	
D052		0.14	0.21	mA	-40°C to +125°C	VDD = 2.3V	FOSC = 500 kHz (**RC_IDLE** mode, MFINTOSC source)
D053		0.15	0.25	mA	-40°C to +125°C	VDD = 3.0V	
D054		0.20	0.31	mA	-40°C to +125°C	VDD = 5.0V	

Note 1: The supply current is mainly a function of operating voltage, frequency and mode. Other factors, such as I/O pin loading and switching rate, oscillator type and circuit, internal code execution pattern and temperature, also have an impact on the current consumption.

Test condition: All Peripheral Module Control bits in PMD0, PMD1 and PMD2 set to '1'.

2: The test conditions for all IDD measurements in active operation mode are:

All I/O pins set as outputs driven to Vss;

$\overline{MCLR}$ = VDD;

OSC1 = external square wave, from rail-to-rail (PRI_RUN and PRI_IDLE only).

FIGURE 28-50: PIC18F2X/4XK22 TYPICAL IDD: PRI_RUN EC MEDIUM POWER

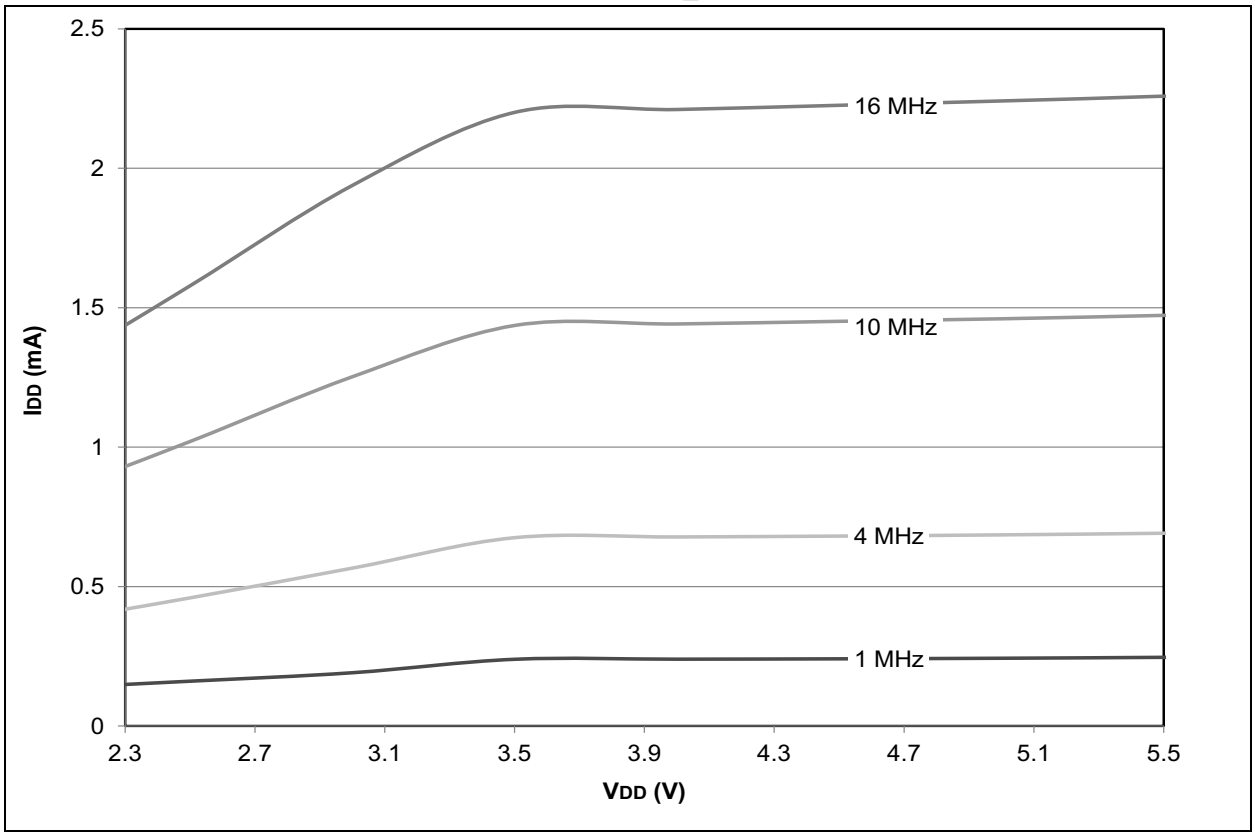


FIGURE 28-51: PIC18F2X/4XK22 MAXIMUM IDD: PRI_RUN EC MEDIUM POWER

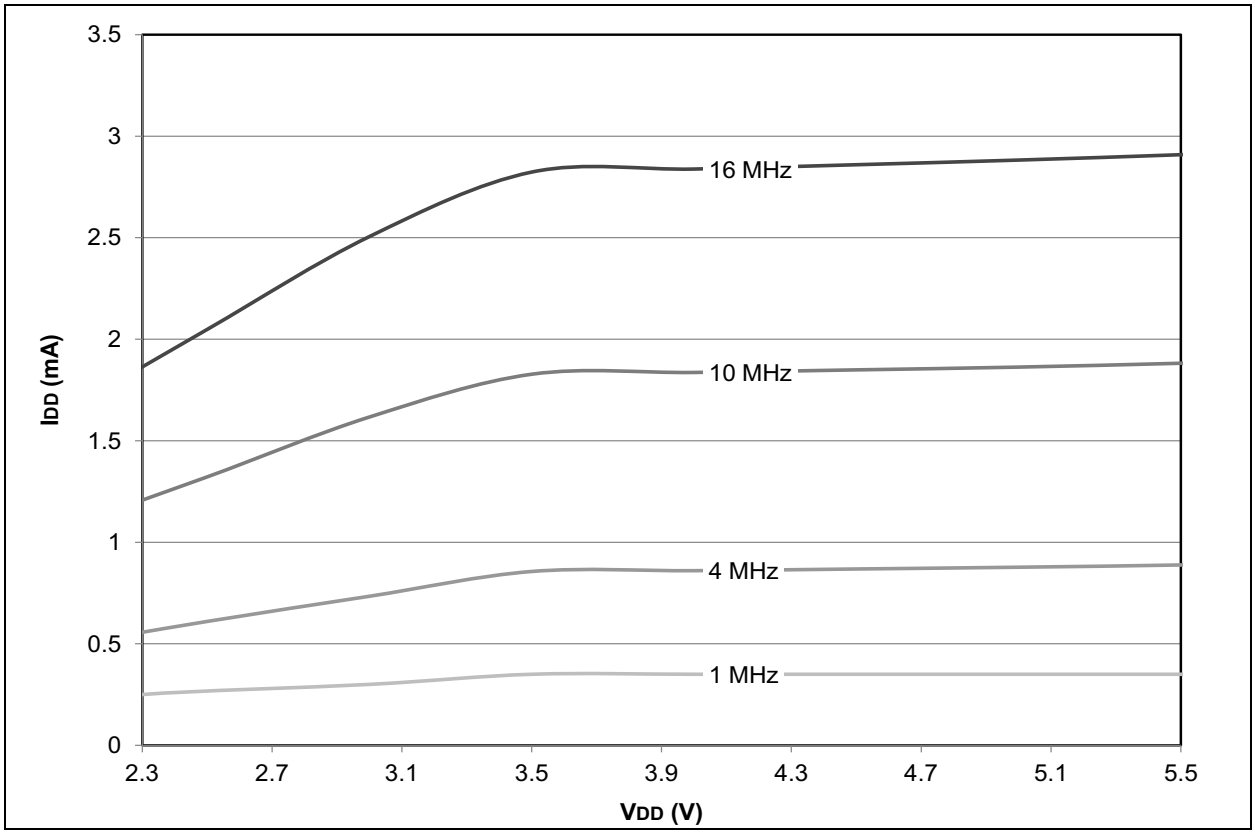


FIGURE 28-68: PIC18LF2X/4XK22 TYPICAL IDD: PRI_IDLE EC with PLL

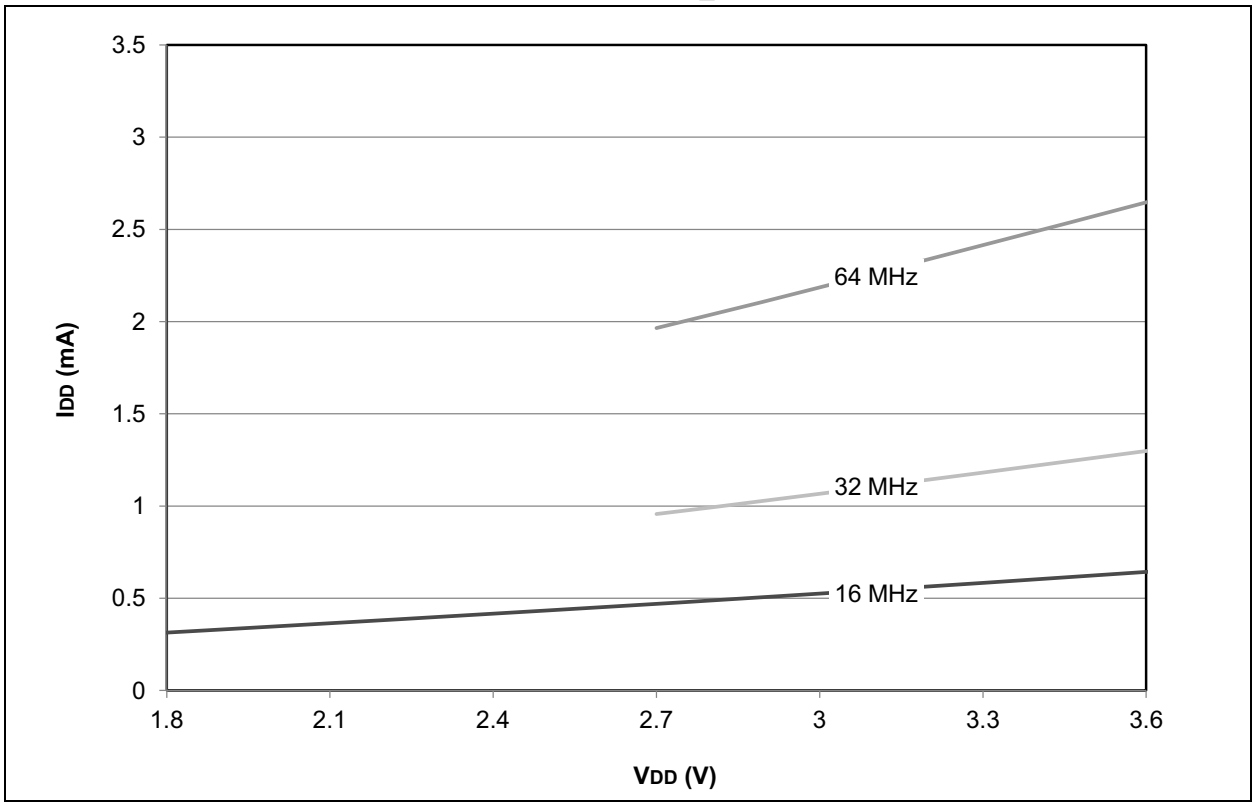


FIGURE 28-69: PIC18LF2X/4XK22 MAXIMUM IDD: PRI_IDLE EC with PLL

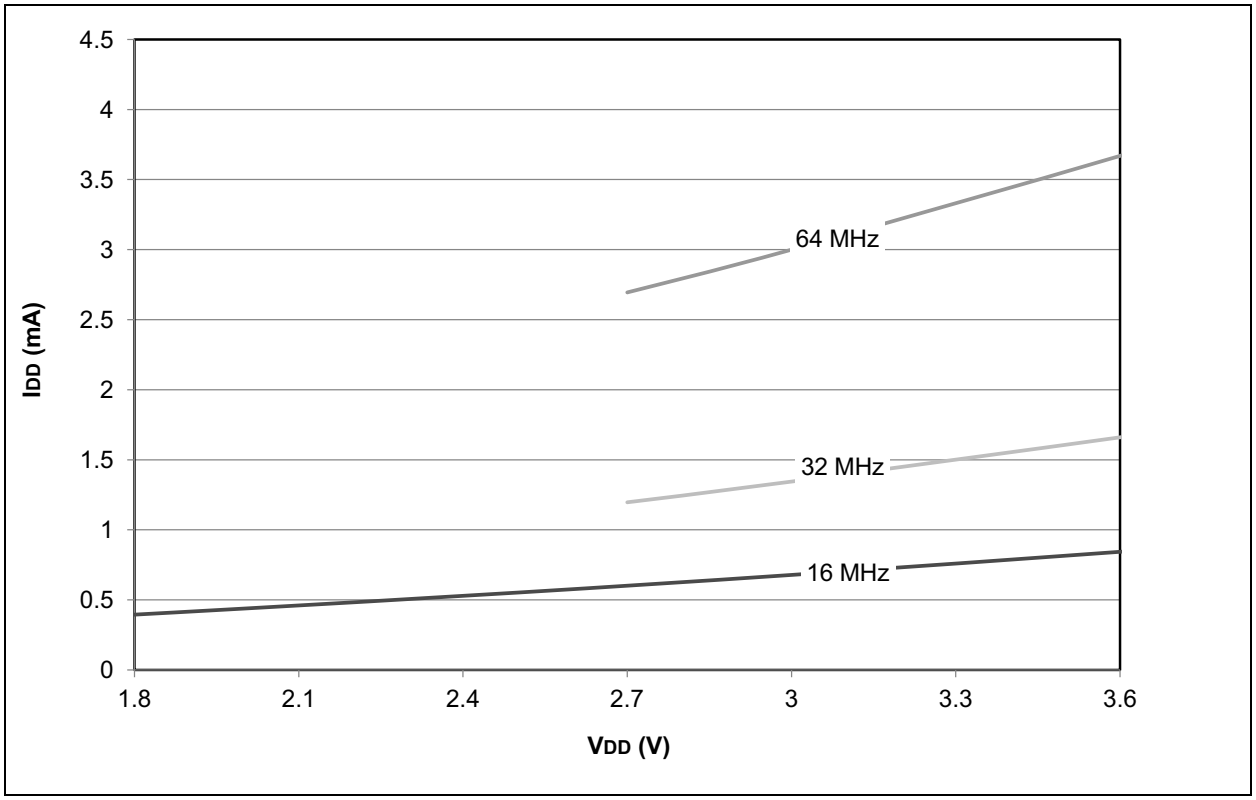


FIGURE 28-90: PIC18(L)F2X/4XK22 COMPARATOR OFFSET VOLTAGE, LOW-POWER MODE; VDD=5.5V

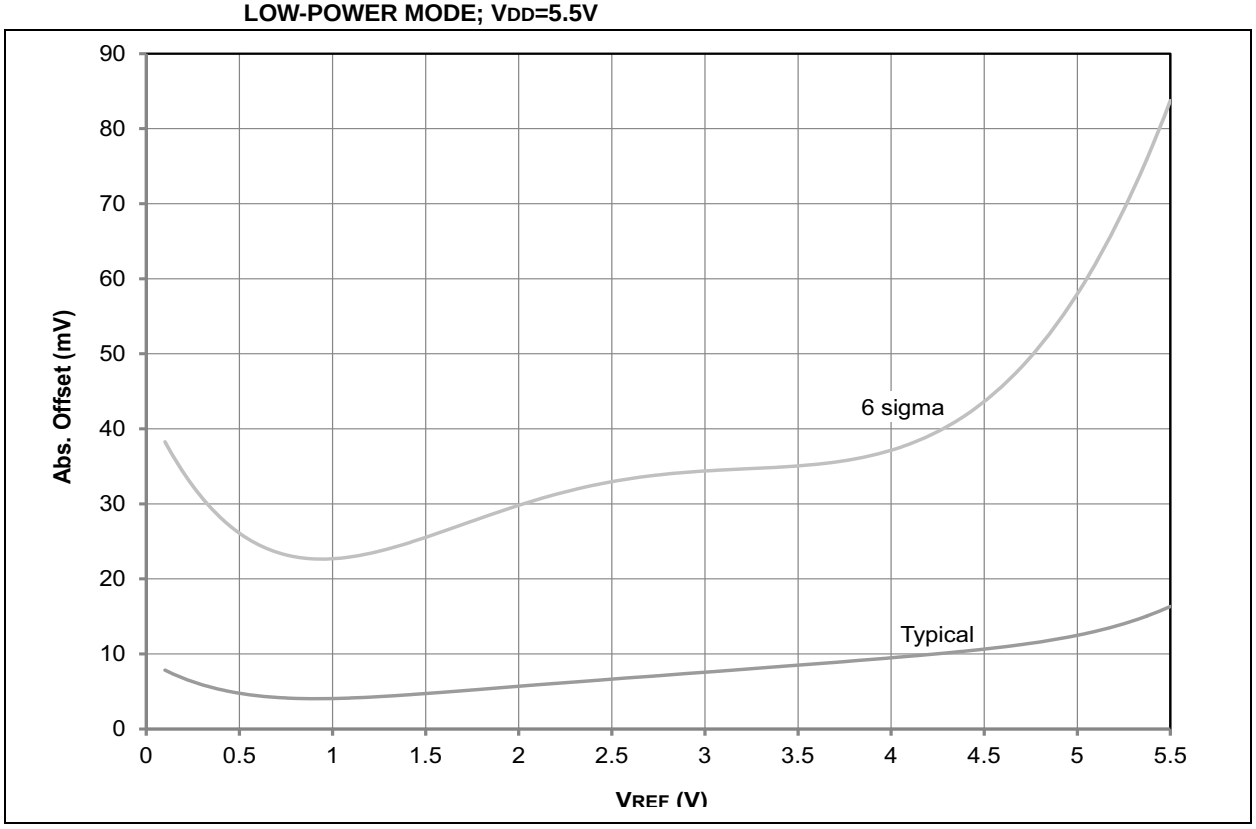


FIGURE 28-91: PIC18(L)F2X/4XK22 COMPARATOR OFFSET VOLTAGE, LOW-POWER MODE; VDD=3.0V

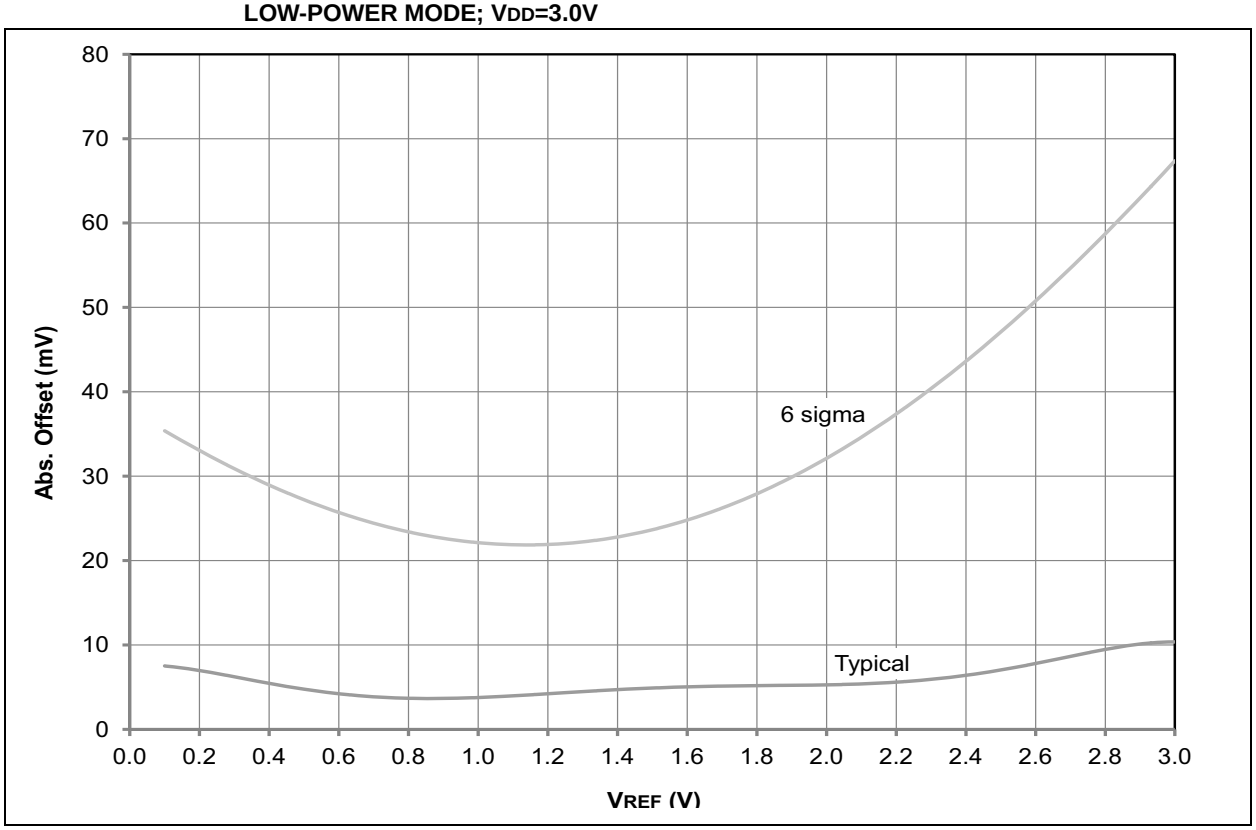


FIGURE 28-101: PIC18LF2X/4XK22 TYPICAL LF-INTOSC FREQUENCY vs. VDD
Min/Max = 31.25 kHz ± 15%, T = -40°C to +85°C

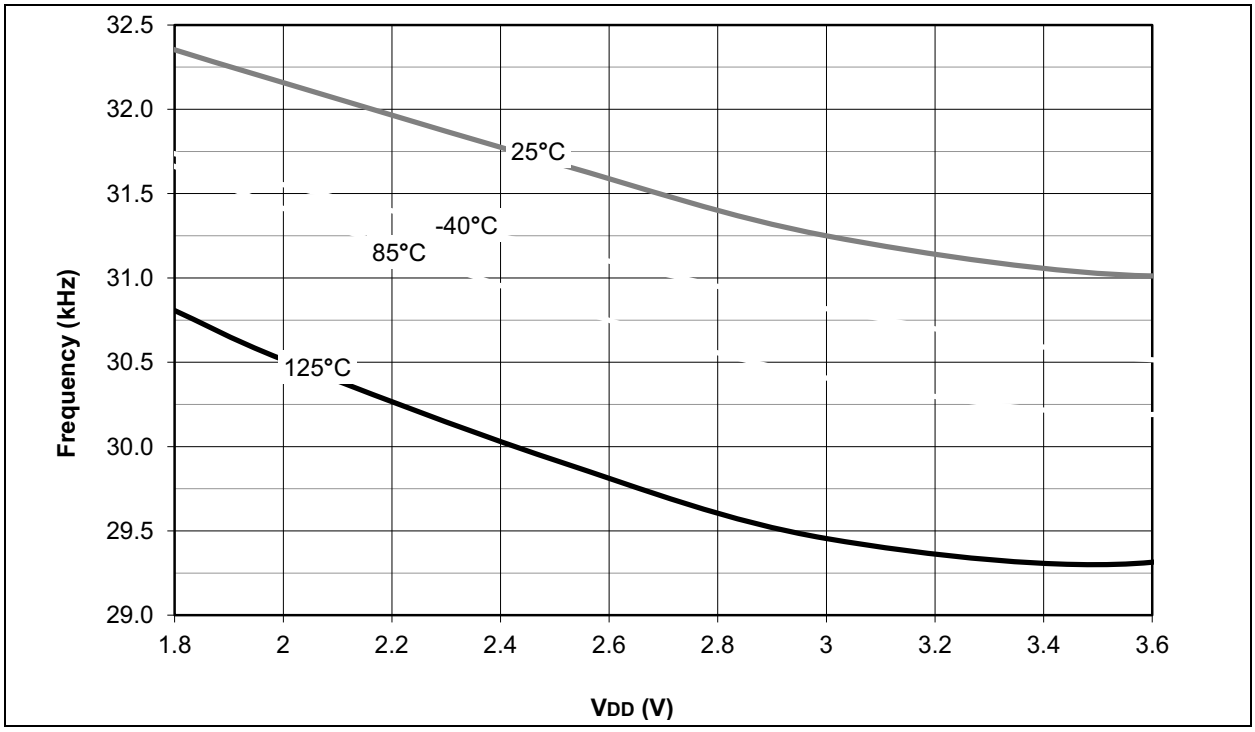
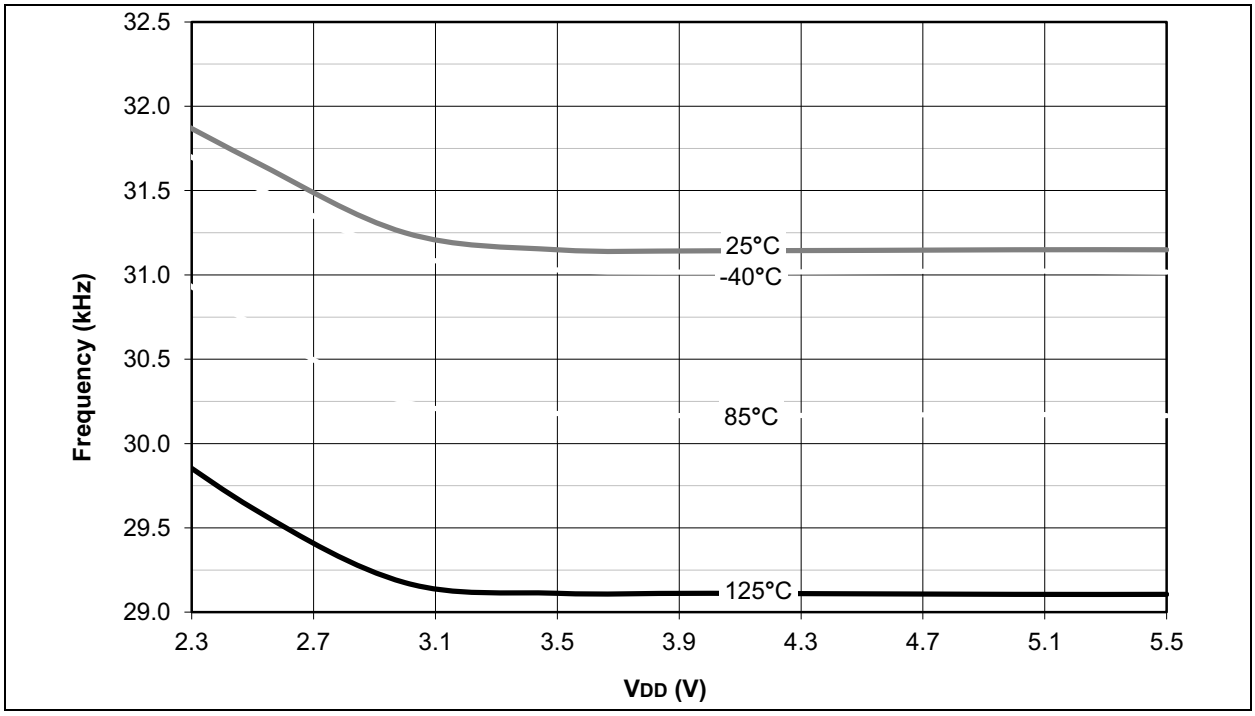
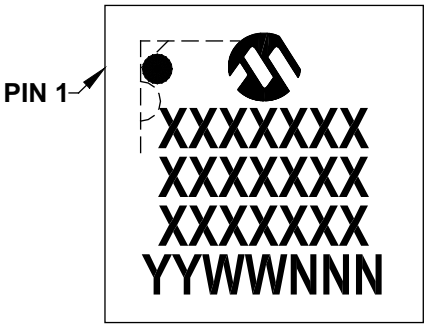


FIGURE 28-102: PIC18F2X/4XK22 TYPICAL LF-INTOSC FREQUENCY vs. VDD
Min/Max = 31.25 kHz ± 15%, T = -40°C to +85°C

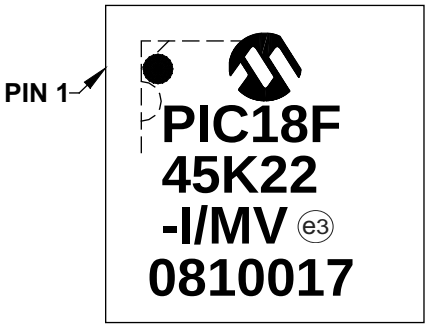


Package Marking Information (Continued)

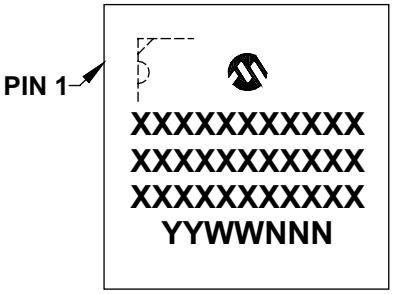
40-Lead UQFN (5x5x0.5 mm)



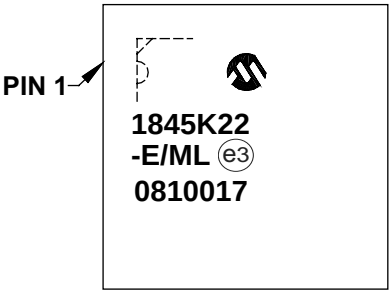
Example



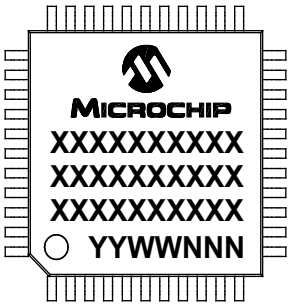
44-Lead QFN (8x8x0.9 mm)



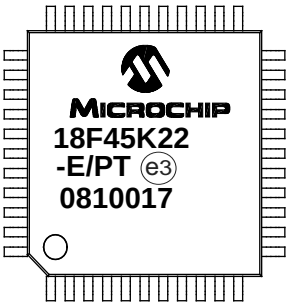
Example



44-Lead TQFP (10x10x1 mm)



Example



Legend:	XX...X	Customer-specific information or Microchip part number
	Y	Year code (last digit of calendar year)
	YY	Year code (last 2 digits of calendar year)
	WW	Week code (week of January 1 is week ‘01’)
	NNN	Alphanumeric traceability code
	(e3)	Pb-free JEDEC® designator for Matte Tin (Sn)
	*	This package is Pb-free. The Pb-free JEDEC designator ((e3)) can be found on the outer packaging for this package.

Note: In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line, thus limiting the number of available characters for customer-specific information.