



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 "<u>Embedded -</u> <u>Microcontrollers</u>"

Details

Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	64MHz
Connectivity	I ² C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, HLVD, POR, PWM, WDT
Number of I/O	24
Program Memory Size	64KB (32K x 16)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	3.8K x 8
Voltage - Supply (Vcc/Vdd)	2.3V ~ 5.5V
Data Converters	A/D 19x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°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/pic18f26k22t-i-ss

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong

Pin Number								
PDIP, SOIC	QFN, UQFN	Pin Name N RB4/IOC0/P1D/T5G/AN11		Buffer Type	Description			
25	22	RB4/IOC0/P1D/T5G/AN11						
		RB4	I/O	TTL	Digital I/O.			
		IOC0	Т	TTL	Interrupt-on-change pin.			
		P1D	0	CMOS	Enhanced CCP1 PWM output.			
		T5G	I	ST	Timer5 external clock gate input.			
		AN11	Ι	Analog	Analog input 11.			
26	23	RB5/IOC1/P2B/P3A/CCP3/T3CKI/T1	G/AN13	3				
		RB5	I/O	TTL	Digital I/O.			
		IOC1	I	TTL	Interrupt-on-change pin.			
		P2B ⁽¹⁾	0	CMOS	Enhanced CCP2 PWM output.			
		P3A ⁽¹⁾	0	CMOS	Enhanced CCP3 PWM output.			
		CCP3 ⁽¹⁾	I/O	ST	Capture 3 input/Compare 3 output/PWM 3 output.			
		T3CKI ⁽²⁾	Т	ST	Timer3 clock input.			
		T1G	Т	ST	Timer1 external clock gate input.			
		AN13	Ι	Analog	Analog input 13.			
27	24	RB6/IOC2/TX2/CK2/PGC						
		RB6	I/O	TTL	Digital I/O.			
		IOC2	Т	TTL	Interrupt-on-change pin.			
		TX2	0	—	EUSART asynchronous transmit.			
		CK2	I/O	ST	EUSART synchronous clock (see related RXx/DTx).			
		PGC	I/O ST In-Circuit Debugger and IC		In-Circuit Debugger and ICSP [™] programming clock pin.			
28	25	RB7/IOC3/RX2/DT2/PGD		-				
		RB7	I/O	TTL	Digital I/O.			
		IOC3	I	TTL	Interrupt-on-change pin.			
		RX2	I	ST	EUSART asynchronous receive.			
		DT2	I/O	ST	EUSART synchronous data (see related TXx/CKx).			
		PGD	I/O	ST	In-Circuit Debugger and ICSP™ programming data pin.			
11	8	RC0/P2B/T3CKI/T3G/T1CKI/SOSCO						
		RCO	I/O	ST	Digital I/O.			
		P2B ⁽²⁾	0	CMOS	Enhanced CCP1 PWM output.			
		ТЗСКІ ⁽¹⁾	I	ST	Timer3 clock input.			
		T3G	I	ST	Timer3 external clock gate input.			
		T1CKI	I	ST	Timer1 clock input.			
		SOSCO	0	—	Secondary oscillator output.			
12	9	RC1/P2A/CCP2/SOSCI	1	I	1			
		RC1	I/O	ST	Digital I/O.			
		P2A	0	CMOS	Enhanced CCP2 PWM output.			
		CCP2 ⁽¹⁾	I/O	ST	Capture 2 input/Compare 2 output/PWM 2 output.			
		SOSCI	Ι	Analog	Secondary oscillator input.			
Logond	Legend: TTL - TTL compatible input CMOS - CMOS compatible input or output: ST - Schmitt Trigger input with CMOS levels:							

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

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.

2.11.3 CLOCK SWITCH TIMING

When switching between one oscillator and another, the new oscillator may not be operating which saves power (see Figure 2-9). If this is the case, there is a delay after the SCS<1:0> bits of the OSCCON register are modified before the frequency change takes place. The OSTS and IOFS bits of the OSCCON register will reflect the current active status of the external and HFINTOSC oscillators. The timing of a frequency selection is as follows:

- 1. SCS<1:0> bits of the OSCCON register are modified.
- 2. The old clock continues to operate until the new clock is ready.
- 3. Clock switch circuitry waits for two consecutive rising edges of the old clock after the new clock ready signal goes true.
- 4. The system clock is held low starting at the next falling edge of the old clock.
- 5. Clock switch circuitry waits for an additional two rising edges of the new clock.
- 6. On the next falling edge of the new clock the low hold on the system clock is released and new clock is switched in as the system clock.
- 7. Clock switch is complete.

See Figure 2-1 for more details.

If the HFINTOSC is the source of both the old and new frequency, there is no start-up delay before the new frequency is active. This is because the old and new frequencies are derived from the HFINTOSC via the postscaler and multiplexer.

Start-up delay specifications are located in **Section 27.0 "Electrical Specifications**", under AC Specifications (Oscillator Module).

2.12 Two-Speed Clock Start-up Mode

Two-Speed Start-up mode provides additional power savings by minimizing the latency between external oscillator start-up and code execution. In applications that make heavy use of the Sleep mode, Two-Speed Start-up will remove the external oscillator start-up time from the time spent awake and can reduce the overall power consumption of the device.

This mode allows the application to wake-up from Sleep, perform a few instructions using the HFINTOSC as the clock source and go back to Sleep without waiting for the primary oscillator to become stable.

Note: Executing a SLEEP instruction will abort the oscillator start-up time and will cause the OSTS bit of the OSCCON register to remain clear.

When the oscillator module is configured for LP, XT or HS modes, the Oscillator Start-up Timer (OST) is enabled (see **Section 2.5.1 "Oscillator Start-up Timer (OST)**"). The OST will suspend program execution until 1024 oscillations are counted. Two-Speed Start-up mode minimizes the delay in code execution by operating from the internal oscillator as the OST is counting. When the OST count reaches 1024 and the OSTS bit of the OSCCON register is set, program execution switches to the external oscillator.

2.12.1 TWO-SPEED START-UP MODE CONFIGURATION

Two-Speed Start-up mode is enabled when all of the following settings are configured as noted:

- Two-Speed Start-up mode is enabled when the IESO of the CONFIG1H Configuration register is set.
- SCS<1:0> (of the OSCCON register) = 00.
- FOSC<2:0> bits of the CONFIG1H Configuration register are configured for LP, XT or HS mode.

Two-Speed Start-up mode becomes active after:

- Power-on Reset (POR) and, if enabled, after Power-up Timer (PWRT) has expired, or
- · Wake-up from Sleep.



3.4.1 PRI_IDLE MODE

This mode is unique among the three low-power Idle modes, in that it does not disable the primary device clock. For timing sensitive applications, this allows for the fastest resumption of device operation with its more accurate primary clock source, since the clock source does not have to "warm-up" or transition from another oscillator.

PRI_IDLE mode is entered from PRI_RUN mode by setting the IDLEN bit and executing a SLEEP instruction. If the device is in another Run mode, set IDLEN first, then clear the SCS bits and execute SLEEP. Although the CPU is disabled, the peripherals continue to be clocked from the primary clock source specified by the FOSC<3:0> Configuration bits. The OSTS bit remains set (see Figure 3-6).

When a wake event occurs, the CPU is clocked from the primary clock source. A delay of interval TCSD is required between the wake event and when code execution starts. This is required to allow the CPU to become ready to execute instructions. After the wake-up, the OSTS bit remains set. The IDLEN and SCS bits are not affected by the wake-up (see Figure 3-7).

3.4.2 SEC_IDLE MODE

In SEC_IDLE mode, the CPU is disabled but the peripherals continue to be clocked from the SOSC oscillator. This mode is entered from SEC_RUN by setting the IDLEN bit and executing a SLEEP instruction. If the device is in another Run mode, set the IDLEN bit first, then set the SCS<1:0> bits to '01' and execute SLEEP. When the clock source is switched to the SOSC oscillator, the primary oscillator is shut down, the OSTS bit is cleared and the SOSCRUN bit is set.

When a wake event occurs, the peripherals continue to be clocked from the SOSC oscillator. After an interval of TCSD following the wake event, the CPU begins executing code being clocked by the SOSC oscillator. The IDLEN and SCS bits are not affected by the wake-up; the SOSC oscillator continues to run (see Figure 3-7).

Note: The SOSC oscillator should already be running prior to entering SEC_IDLE mode. At least one of the secondary oscillator enable bits (SOSCGO, T1SOSCEN, T3SOSCEN or T5SOSCEN) must be set when the SLEEP instruction is executed. Otherwise, the main system clock will continue to operate in the previously selected mode and the corresponding IDLE mode will be entered (i.e., PRI_IDLE or RC_IDLE).



6.6 Writing to Flash Program Memory

The programming block size is 64 bytes. Word or byte programming is not supported.

Table writes are used internally to load the holding registers needed to program the Flash memory. There are only as many holding registers as there are bytes in a write block (64 bytes).

Since the Table Latch (TABLAT) is only a single byte, the TBLWT instruction needs to be executed 64 times for each programming operation. All of the table write operations will essentially be short writes because only the holding registers are written. After all the holding registers have been written, the programming operation of that block of memory is started by configuring the EECON1 register for a program memory write and performing the long write sequence. The long write is necessary for programming the internal Flash. Instruction execution is halted during a long write cycle. The long write will be terminated by the internal programming timer.

The EEPROM on-chip timer controls the write time. The write/erase voltages are generated by an on-chip charge pump, rated to operate over the voltage range of the device.

Note: The default value of the holding registers on device Resets and after write operations is FFh. A write of FFh to a holding register does not modify that byte. This means that individual bytes of program memory may be modified, provided that the change does not attempt to change any bit from a '0' to a '1'. When modifying individual bytes, it is not necessary to load all holding registers before executing a long write operation.

FIGURE 6-5: TABLE WRITES TO FLASH PROGRAM MEMORY



6.6.1 FLASH PROGRAM MEMORY WRITE SEQUENCE

The sequence of events for programming an internal program memory location should be:

- 1. Read 64 bytes into RAM.
- 2. Update data values in RAM as necessary.
- 3. Load Table Pointer register with address being erased.
- 4. Execute the block erase procedure.
- 5. Load Table Pointer register with address of first byte being written.
- 6. Write the 64-byte block into the holding registers with auto-increment.
- 7. Set the EECON1 register for the write operation:
 - set EEPGD bit to point to program memory;
 - · clear the CFGS bit to access program memory;
 - set WREN to enable byte writes.

- 8. Disable interrupts.
- 9. Write 55h to EECON2.
- 10. Write 0AAh to EECON2.
- 11. Set the WR bit. This will begin the write cycle.
- 12. The CPU will stall for duration of the write (about 2 ms using internal timer).
- 13. Re-enable interrupts.
- 14. Verify the memory (table read).

This procedure will require about 6 ms to update each write block of memory. An example of the required code is given in Example 6-3.

Note: Before setting the WR bit, the Table Pointer address needs to be within the intended address range of the bytes in the holding registers.

7.6 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.

The microcontroller itself can both read and write to the internal data EEPROM, regardless of the state of the code-protect Configuration bit. Refer to Section 24.0 "Special Features of the CPU" for additional information.

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

7.8 Using the Data EEPROM

The data EEPROM is a high-endurance, byte addressable array that has been optimized for the storage of frequently changing information (e.g., program variables or other data that are updated often). When variables in one section change frequently, while variables in another section do not change, it is possible to exceed the total number of write cycles to the EEPROM without exceeding the total number of write cycles to a single byte. Refer to the Data EEPROM Memory parameters in **Section 27.0** "**Electrical Specifications**" for write cycle limits. If this is the case, then an array refresh must be performed. For this reason, variables that change infrequently (such as constants, IDs, calibration, etc.) should be stored in Flash program memory.

A simple data EEPROM refresh routine is shown in Example 7-3.

Note: If data EEPROM is only used to store constants and/or data that changes rarely, an array refresh is likely not required. See specification.

	CLRF	EEADR	;	Start at address 0
	CLRF	EEADRH	;	if > 256 bytes EEPROM
	BCF	EECON1, CFGS	;	Set for memory
	BCF	EECON1, EEPGD	;	Set for Data EEPROM
	BCF	INTCON, GIE	;	Disable interrupts
	BSF	EECON1, WREN	;	Enable writes
Loop			;	Loop to refresh array
	BSF	EECON1, RD	;	Read current address
	MOVLW	55h	;	
	MOVWF	EECON2	;	Write 55h
	MOVLW	0AAh	;	
	MOVWF	EECON2	;	Write OAAh
	BSF	EECON1, WR	;	Set WR bit to begin write
	BTFSC	EECON1, WR	;	Wait for write to complete
	BRA	\$-2		
	INCFSZ	EEADR, F	;	Increment address
	BRA	LOOP	;	Not zero, do it again
	INCFSZ	EEADRH, F	;	if > 256 bytes, Increment address
	BRA	LOOP	;	if > 256 bytes, Not zero, do it again
	BCF	EECON1, WREN	;	Disable writes
	BSF	INTCON, GIE	;	Enable interrupts

EXAMPLE 7-3: DATA EEPROM REFRESH ROUTINE

10.2 PORTB Registers

PORTB is an 8-bit wide, bidirectional port. The corresponding data direction register is TRISB. Setting a TRISB bit (= 1) will make the corresponding PORTB pin an input (i.e., disable the output driver). Clearing a TRISB bit (= 0) will make the corresponding PORTB pin an output (i.e., enable the output driver and put the contents of the output latch on the selected pin).

The Data Latch register (LATB) is also memory mapped. Read-modify-write operations on the LATB register read and write the latched output value for PORTB.

		DODTR
EXAIVIPLE 10-2:	INTIALIZING	PURID

MOVLB	0xF	;	Set BSR for banked SFRs
CLRF	PORTB	;	Initialize PORTB by
		;	clearing output
		;	data latches
CLRF	LATB	;	Alternate method
		;	to clear output
		;	data latches
MOVLW	OFOh	;	Value for init
MOVWF	ANSELB	;	Enable RB<3:0> for
		;	digital input pins
		;	(not required if config bit
		;	PBADEN is clear)
MOVLW	0CFh	;	Value used to
		;	initialize data
		;	direction
MOVWF	TRISB	;	Set RB<3:0> as inputs
		;	RB<5:4> as outputs
		;	RB<7:6> as inputs

10.2.1 PORTB OUTPUT PRIORITY

Each PORTB pin is multiplexed with other functions. The pins, their combined functions and their output priorities are briefly described here. For additional information, refer to the appropriate section in this data sheet.

When multiple outputs are enabled, the actual pin control goes to the peripheral with the higher priority. Table 10-4 lists the PORTB pin functions from the highest to the lowest priority.

Analog input functions, such as ADC, comparator and SR latch inputs, are not shown in the priority lists.

These inputs are active when the I/O pin is set for Analog mode using the ANSELx registers. Digital output functions may control the pin when it is in Analog mode with the priority shown below.

10.3 Additional PORTB Pin Functions

PORTB pins RB<7:4> have an interrupt-on-change option. All PORTB pins have a weak pull-up option.

10.3.1 WEAK PULL-UPS

Each of the PORTB pins has an individually controlled weak internal pull-up. When set, each bit of the WPUB register enables the corresponding pin pull-up. When cleared, the RBPU bit of the INTCON2 register enables pull-ups on all pins which also have their corresponding WPUB bit set. When set, the RBPU bit disables all weak pull-ups. The weak pull-up is automatically turned off when the port pin is configured as an output. The pull-ups are disabled on a Power-on Reset.

Note:	On a Power-on Reset, RB<5:0> are configured as analog inputs by default and read as '0'; RB<7:6> are configured as digital inputs.
	When the PBADEN Configuration bit is set to '1', RB<5:0> will alternatively be configured as digital inputs on POR.

10.3.2 INTERRUPT-ON-CHANGE

Four of the PORTB pins (RB<7:4>) are individually configurable as interrupt-on-change pins. Control bits in the IOCB register enable (when set) or disable (when clear) the interrupt function for each pin.

When set, the RBIE bit of the INTCON register enables interrupts on all pins which also have their corresponding IOCB bit set. When clear, the RBIE bit disables all interrupt-on-changes.

Only pins configured as inputs can cause this interrupt to occur (i.e., any RB<7:4> pin configured as an output is excluded from the interrupt-on-change comparison).

For enabled interrupt-on-change pins, the values are compared with the old value latched on the last read of PORTB. The 'mismatch' outputs of the last read are OR'd together to set the PORTB Change Interrupt flag bit (RBIF) in the INTCON register.

This interrupt can wake the device from the Sleep mode, or any of the Idle modes. The user, in the Interrupt Service Routine, can clear the interrupt in the following manner:

- Any read or write of PORTB to clear the mismatch condition (except when PORTB is the source or destination of a MOVFF instruction).
- b) Execute at least one instruction after reading or writing PORTB, then clear the flag bit, RBIF.

11.0 TIMER0 MODULE

The Timer0 module incorporates the following features:

- Software selectable operation as a timer or counter in both 8-bit or 16-bit modes
- · Readable and writable registers
- Dedicated 8-bit, software programmable prescaler
- · Selectable clock source (internal or external)
- Edge select for external clock
- Interrupt-on-overflow

The T0CON register (Register 11-1) controls all aspects of the module's operation, including the prescale selection. It is both readable and writable.

A simplified block diagram of the Timer0 module in 8-bit mode is shown in Figure 11-1. Figure 11-2 shows a simplified block diagram of the Timer0 module in 16-bit mode.

11.1 Register Definitions: Timer0 Control

REGISTER 11-1: TOCON: TIMERO CONTROL 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
TMR0ON	T08BIT	TOCS	TOSE	PSA		TOPS<2:0>	
bit 7							bit 0

Legend:									
R = Readab	ole bit	W = Writable bit	U = Unimplemented bit,	, read as '0'					
-n = Value a	t POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown					
bit 7	TMR0ON	I: Timer0 On/Off Control bit							
	1 = Enab	les Timer0							
	0 = Stops	s Timer0							
bit 6 T08BIT : Timer0 8-bit/16-bit Control bit									
	1 = Time	r0 is configured as an 8-bit ti	mer/counter						
	0 = Time	r0 is configured as a 16-bit ti	mer/counter						
bit 5	TOCS: Ti	mer0 Clock Source Select bi	t						
	1 = Trans	sition on T0CKI pin							
	0 = Interr	nal instruction cycle clock (C	LKOUT)						
bit 4	TOSE: Ti	mer0 Source Edge Select bit	t						
	1 = Incre	ment on high-to-low transitio	n on T0CKI pin						
	0 = Incre	ment on low-to-high transitio	n on T0CKI pin						
bit 3	PSA: Tim	PSA: Timer0 Prescaler Assignment bit							
	1 = TIme	r0 prescaler is NOT assigne	d. Timer0 clock input bypasse	es prescaler.					
	0 = Time	r0 prescaler is assigned. Tim	ner0 clock input comes from p	rescaler output.					
bit 2-0	T0PS<2:	0>: Timer0 Prescaler Select	bits						
	111 = 1 :2	256 prescale value							
	110 = 1 :	128 prescale value							
	101 = 1:6	64 prescale value							
	100 = 1:3	32 prescale value							
	011 = 1	rescale value							
	$0 \pm 0 = 1.0$	1 prescale value							
	000 = 12	prescale value							

PIC18(L)F2X/4XK22





15.6.13.2 Bus Collision During a Repeated Start Condition

During a Repeated Start condition, a bus collision occurs if:

- a) A low level is sampled on SDAx when SCLx goes from low level to high level (Case 1).
- SCLx goes low before SDAx is asserted low, indicating that another master is attempting to transmit a data '1' (Case 2).

When the user releases SDAx and the pin is allowed to float high, the BRG is loaded with SSPxADD and counts down to zero. The SCLx pin is then deasserted and when sampled high, the SDAx pin is sampled. If SDAx is low, a bus collision has occurred (i.e., another master is attempting to transmit a data '0', Figure 15-36). If SDAx is sampled high, the BRG is reloaded and begins counting. If SDAx goes from high-to-low before the BRG times out, no bus collision occurs because no two masters can assert SDAx at exactly the same time.

If SCLx goes from high-to-low before the BRG times out and SDAx has not already been asserted, a bus collision occurs. In this case, another master is attempting to transmit a data '1' during the Repeated Start condition, see Figure 15-37.

If, at the end of the BRG time-out, both SCLx and SDAx are still high, the SDAx pin is driven low and the BRG is reloaded and begins counting. At the end of the count, regardless of the status of the SCLx pin, the SCLx pin is driven low and the Repeated Start condition is complete.



FIGURE 15-36: BUS COLLISION DURING A REPEATED START CONDITION (CASE 1)





15.7 Baud Rate Generator

The MSSPx module has a Baud Rate Generator available for clock generation in both I²C and SPI Master modes. The Baud Rate Generator (BRG) reload value is placed in the SSPxADD register (Register 15-7). When a write occurs to SSPxBUF, the Baud Rate Generator will automatically begin counting down.

Once the given operation is complete, the internal clock will automatically stop counting and the clock pin will remain in its last state.

An internal signal "Reload" in Figure 15-40 triggers the value from SSPxADD to be loaded into the BRG counter.

This occurs twice for each oscillation of the module clock line. The logic dictating when the reload signal is asserted depends on the mode the MSSPx is being operated in.

Table 15-3 demonstrates clock rates based on instruction cycles and the BRG value loaded into SSPxADD.



$$FCLOCK = \frac{Fosc}{(SSPxADD + 1)(4)}$$

FIGURE 15-40: BAUD RATE GENERATOR BLOCK DIAGRAM



Note: 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.

TABLE 15-3: MSSPx CLOCK RATE W/BRG

Fosc	Fcy	BRG Value	Fclock (2 Rollovers of BRG)
32 MHz	8 MHz	13h	400 kHz ⁽¹⁾
32 MHz	8 MHz	19h	308 kHz
32 MHz	8 MHz	4Fh	100 kHz
16 MHz	4 MHz	09h	400 kHz ⁽¹⁾
16 MHz	4 MHz	0Ch	308 kHz
16 MHz	4 MHz	27h	100 kHz
4 MHz	1 MHz	09h	100 kHz

Note 1: The I²C interface does not conform to the 400 kHz I²C specification (which applies to rates greater than 100 kHz) in all details, but may be used with care where higher rates are required by the application.

Mnemonic, Operands				16-Bit Instruction Word				Status	
		Description	Cycles	MSb			LSb	Affected	Notes
LITERAL	OPERAT	IONS							
ADDLW	k	Add literal and WREG	1	0000	1111	kkkk	kkkk	C, DC, Z, OV, N	
ANDLW	k	AND literal with WREG	1	0000	1011	kkkk	kkkk	Z, N	
IORLW	k	Inclusive OR literal with WREG	1	0000	1001	kkkk	kkkk	Z, N	
LFSR	f, k	Move literal (12-bit) 2nd word	2	1110	1110	00ff	kkkk	None	
		to FSR(f) 1st word		1111	0000	kkkk	kkkk		
MOVLB	k	Move literal to BSR<3:0>	1	0000	0001	0000	kkkk	None	
MOVLW	k	Move literal to WREG	1	0000	1110	kkkk	kkkk	None	
MULLW	k	Multiply literal with WREG	1	0000	1101	kkkk	kkkk	None	
RETLW	k	Return with literal in WREG	2	0000	1100	kkkk	kkkk	None	
SUBLW	k	Subtract WREG from literal	1	0000	1000	kkkk	kkkk	C, DC, Z, OV, N	
XORLW	k	Exclusive OR literal with WREG	1	0000	1010	kkkk	kkkk	Z, N	
DATA MEN	IORY ↔	PROGRAM MEMORY OPERATION	IS						
TBLRD*		Table Read	2	0000	0000	0000	1000	None	
TBLRD*+		Table Read with post-increment		0000	0000	0000	1001	None	
TBLRD*-		Table Read with post-decrement		0000	0000	0000	1010	None	
TBLRD+* Table Read with pre-increment			0000	0000	0000	1011	None		
TBLWT* Table Write 2		2	0000	0000	0000	1100	None		
TBLWT*+		Table Write with post-increment		0000	0000	0000	1101	None	
TBLWT*-		Table Write with post-decrement		0000	0000	0000	1110	None	
TBLWT+*		Table Write with pre-increment		0000	0000	0000	1111	None	

TABLE 25-2: PIC18(L)F2X/4XK22 INSTRUCTION SET (CONTINUED)

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.

26.11 Demonstration/Development Boards, Evaluation Kits, and Starter Kits

A wide variety of demonstration, development and evaluation boards for various PIC MCUs and dsPIC DSCs allows quick application development on fully functional systems. Most boards include prototyping areas for adding custom circuitry and provide application firmware and source code for examination and modification.

The boards support a variety of features, including LEDs, temperature sensors, switches, speakers, RS-232 interfaces, LCD displays, potentiometers and additional EEPROM memory.

The demonstration and development boards can be used in teaching environments, for prototyping custom circuits and for learning about various microcontroller applications.

In addition to the PICDEM[™] and dsPICDEM[™] demonstration/development board series of circuits, Microchip has a line of evaluation kits and demonstration software for analog filter design, KEELOQ[®] security ICs, CAN, IrDA[®], PowerSmart battery management, SEEVAL[®] evaluation system, Sigma-Delta ADC, flow rate sensing, plus many more.

Also available are starter kits that contain everything needed to experience the specified device. This usually includes a single application and debug capability, all on one board.

Check the Microchip web page (www.microchip.com) for the complete list of demonstration, development and evaluation kits.

26.12 Third-Party Development Tools

Microchip also offers a great collection of tools from third-party vendors. These tools are carefully selected to offer good value and unique functionality.

- Device Programmers and Gang Programmers from companies, such as SoftLog and CCS
- Software Tools from companies, such as Gimpel and Trace Systems
- Protocol Analyzers from companies, such as Saleae and Total Phase
- Demonstration Boards from companies, such as MikroElektronika, Digilent[®] and Olimex
- Embedded Ethernet Solutions from companies, such as EZ Web Lynx, WIZnet and IPLogika[®]

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

PIC18LF2X/4XK22		Standard Operating Conditions (unless otherwise stated)Operating temperature $-40^{\circ}C \le TA \le +125^{\circ}C$								
PIC18F2X/4XK22		Standard Operating Conditions (unless otherwise stated)Operating temperature $-40^{\circ}C \le TA \le +125^{\circ}C$								
Param Device Characteristics		Тур	Тур	Max	Max	Unite		Conditions		
		+25°C	+60°C	+85°C	+125°C	Units	Vdd	Notes		
Power-	down Base Current (IPD) ⁽¹⁾									
D006	Sleep mode	0.01	0.04	2	10	μΑ	1.8V	WDT, BOR, FVR and		
		0.01	0.06	2	10	μA	3.0V	SOSC disabled, all Peripherals inactive		
		12	13	25	35	μA	2.3V			
		13	14	30	40	μA	3.0V			
		13	14	35	50	μA	5.0V			
Power-	down Module Differential Cur	rent (delt	a IPD)	1	r	n	1	1		
D007	Watchdog Timer	0.3	0.3	2.5	2.5	μA	1.8V			
		0.5	0.5	2.5	2.5	μA	3.0V			
		0.35	0.35	5.0	5.0	μA	2.3V			
		0.5	0.5	5.0	5.0	μA	3.0V			
		0.5	0.5	5.0	5.0	μΑ	5.0V			
D008	Brown-out Reset ⁽²⁾	8	8.5	15	16	μA	2.0V			
		9	9.5	15	16	μA	3.0V			
		3.4	3.4	15	16	μA	2.3V			
		3.8	3.8	15	16	μA	3.0V			
		5.2	5.2	15	16	μA	5.0V			
D010	High/Low Voltage Detect ⁽²⁾	6.5	6.7	15	15	μA	2.0V			
		7	7.5	15	15	μA	3.0V			
		2.1	2.1	15	15	μA	2.3V			
		2.4	2.4	15	15	μA	3.0V			
		3.2	3.2	15	15	μA	5.0V			
D011	Secondary Oscillator	0.5	1	3	10	μA	1.8V			
		0.6	1.1	4	10	μA	3.0V	32 kHz on SOSC		
		0.5	1	3	10	μA	2.3V			
		0.6	1.1	4	10	μA	3.0V			
		0.6	1.1	5	10	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.

27.11.2 TIMING CONDITIONS

The temperature and voltages specified in Table 27-6 apply to all timing specifications unless otherwise noted. Figure 27-6 specifies the load conditions for the timing specifications.

TABLE 27-6: TEMPERATURE AND VOLTAGE SPECIFICATIONS – AC

AC CHARACTERISTICS	Standard Operating Conditions (unless otherwise stated)					
	Operating temperature $-40^{\circ}C \le TA \le +125^{\circ}C$					
	Operating voltage VDD range as described in Section 27.1 "DC Characteristics:					
	Supply Voltage, PIC18(L)F2X/4XK22" and Section 27.9 "Memory Programming					
	Requirements".					

FIGURE 27-6: LOAD CONDITIONS FOR DEVICE TIMING SPECIFICATIONS



PIC18(L)F2X/4XK22



FIGURE 28-9: PIC18LF2X/4XK22 DELTA IPD SECONDARY OSCILLATOR





PIC18(L)F2X/4XK22





FIGURE 28-73: PIC18LF2X/4XK22 MAXIMUM IDD: SEC_RUN 32.768 kHz





FIGURE 28-83: PIC18(L)F2X/4XK22 SCHMITT TRIGGER BUFFER INPUT HIGH VOLTAGE









© 2010-2016 Microchip Technology Inc.



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





28-Lead Plastic Small Outline (SO) - Wide, 7.50 mm Body [SOIC]

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





Microchip Technology Drawing C04-052C Sheet 1 of 2