Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

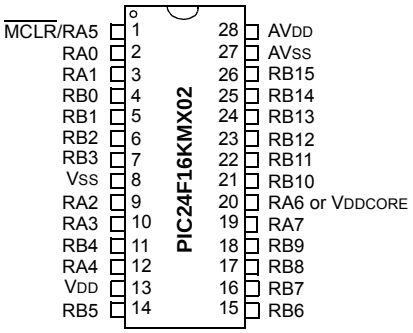
"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" 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	16-Bit
Speed	32MHz
Connectivity	I²C, IrDA, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LVD, POR, PWM, WDT
Number of I/O	23
Program Memory Size	8KB (2.75K x 24)
Program Memory Type	FLASH
EEPROM Size	512 x 8
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5V
Data Converters	A/D 19x10b/12b; D/A 2x8b
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/pic24fv08km202-i-ss

Pin Diagrams (Continued)

28-Pin SPDIP/SSOP/SOIC



Pin	Pin Features	
	PIC24FXXKMX02	PIC24FVXXKMX02
1	MCLR/VPP/RA5	
2	CVREF+/VREF+/ /AN0/ /CN2/RA0	
3	CVREF-/VREF-/AN1/CN3/RA1	CVREF-/VREF-/AN1/RA1
4	PGED1/AN2/CTCMP/ULPWU/C1IND/ / / /CN4/RB0	
5	PGEC1/ / /AN3/C1INC/ / /CTED12/CN5/RB1	
6	/ /AN4/C1INB/ / /U1RX/TCKIB/CTED13/CN6/RB2	
7	/AN5/C1INA/ / /CN7/RB3	
8	VSS	
9	OSCI/CLKI/AN13/CN30/RA2	
10	OSCO/CLKO/AN14/CN29/RA3	
11	SOSCI/AN15/ / /CN1/RB4	
12	SOSCO/SCLKI/AN16/PWRLCLK/ /CN0/RA4	
13	VDD	
14	PGED3/AN17/ASDA1/ / /OC1E/CLCINA/CN27/RB5	
15	PGEC3/AN18/ASCL1/ / /OC1F/CLCINB/CN24/RB6	
16	AN19/U1TX/INT0/CN23/RB7	AN19/U1TX/ / /INT0/CN23/RB7
17	AN20/SCL1/U1CTS/C3OUT/OC1B/CTED10/CN22/RB8	
18	AN21/SDA1/T1CK/U1RTS/U1BCLK/IC2/ /CLC1O/CTED4/CN21/RB9	
19	/IC1/ / /CTED3/CN9/RA7	
20	/OC1A/CTED1/INT2/CN8/RA6	VCAP OR VDDCORE
21	PGED2/SDI1/ /OC1C/CTED11/CN16/RB10	
22	PGEC2/SCK1/OC2A/CTED9/CN15/RB11	
23	/AN12/HLVDIN/ / / /CTED2/CN14/RB12	/AN12/HLVDIN/ / / /CTED2/INT2/CN14/RB12
24	/ /AN11/SDO1/OCFB/ /OC1D/CTPLS/CN13/RB13	
25	/CVREF/ / /AN10/ / /C1OUT/OCFA/CTED5/INT1/CN12/RB14	
26	/ /AN9/ /REFO/SS1/TCKIA/CTED6/CN11/RB15	
27	VSS/AVSS	
28	VDD/AVDD	

Legend: Values in indicate pin function differences between PIC24F(V)XXKM202 and PIC24F(V)XXKM102 devices.

TABLE 1-5: PIC24FV16KM204 FAMILY PINOUT DESCRIPTION (CONTINUED)

Function	F					FV					I/O	Buffer	Description
	Pin Number					Pin Number							
	20-Pin PDIP/ SSOP/ SOIC	28-Pin PDIP/ SSOP/ SOIC	28-Pin QFN	44-Pin QFN/ TQFP	48-Pin UQFN	20-Pin PDIP/ SSOP/ SOIC	28-Pin PDIP/ SSOP/ SOIC	28-Pin QFN	44-Pin QFN/ TQFP	48-Pin UQFN			
C1OUT	17	25	22	14	15	17	25	22	14	15	O	—	Comparator 1 Output
C2INA	—	5	2	22	24	—	5	2	22	24	I	ANA	Comparator 2 Input A (+)
C2INB	—	4	1	21	23	—	4	1	21	23	I	ANA	Comparator 2 Input B (-)
C2INC	—	7	4	24	26	—	7	4	24	26	I	ANA	Comparator 2 Input C (+)
C2IND	—	6	3	23	25	—	6	3	23	25	I	ANA	Comparator 2 Input D (-)
C2OUT	—	20	17	7	7	—	16	13	43	47	O	—	Comparator 2 Output
C3INA	—	26	23	15	16	—	26	23	15	16	I	ANA	Comparator 3 Input A (+)
C3INB	—	25	22	14	15	—	25	22	14	15	I	ANA	Comparator 3 Input B (-)
C3INC	—	2	27	19	21	—	2	27	19	21	I	ANA	Comparator 3 Input C (+)
C3IND	—	4	1	21	23	—	4	1	21	23	I	ANA	Comparator 3 Input D (-)
C3OUT	—	17	14	44	48	—	17	14	44	48	O	—	Comparator 3 Output
CLC1O	13	18	15	1	1	13	18	15	1	1	O	—	CLC 1 Output
CLC2O	—	19	16	6	6	—	19	16	6	6	O	—	CLC 2 Output
CLCINA	9	14	11	41	45	9	14	11	41	45	I	ST	CLC External Input A
CLCINB	10	15	12	42	46	10	15	12	42	46	I	ST	CLC External Input B
CLKI	7	9	6	30	33	7	9	6	30	33	I	ANA	Primary Clock Input
CLKO	8	10	7	31	34	8	10	7	31	34	O	—	System Clock Output
CN0	10	12	9	34	37	10	12	9	34	37	I	ST	Interrupt-on-Change Inputs
CN1	9	11	8	33	36	9	11	8	33	36	I	ST	Interrupt-on-Change Inputs
CN2	2	2	27	19	21	2	2	27	19	21	I	ST	Interrupt-on-Change Inputs
CN3	3	3	28	20	22	3	3	28	20	22	I	ST	Interrupt-on-Change Inputs
CN4	4	4	1	21	23	4	4	1	21	23	I	ST	Interrupt-on-Change Inputs
CN5	5	5	2	22	24	5	5	2	22	24	I	ST	Interrupt-on-Change Inputs
CN6	6	6	3	23	25	6	6	3	23	25	I	ST	Interrupt-on-Change Inputs
CN7	—	7	4	24	26	—	7	4	24	26	I	ST	Interrupt-on-Change Inputs
CN8	14	20	17	7	7	—	—	—	—	—	I	ST	Interrupt-on-Change Inputs
CN9	—	19	16	6	6	—	19	16	6	6	I	ST	Interrupt-on-Change Inputs
CN10	—	—	—	27	29	—	—	—	27	29	I	ST	Interrupt-on-Change Inputs
CN11	18	26	23	15	16	18	26	23	15	16	I	ST	Interrupt-on-Change Inputs
CN12	17	25	22	14	15	17	25	22	14	15	I	ST	Interrupt-on-Change Inputs

Legend: ANA = Analog level input/output, ST = Schmitt Trigger input buffer, I^2C™ = I^2C/SMBus input buffer

4.2.5 SOFTWARE STACK

In addition to its use as a working register, the W15 register in PIC24F devices is also used as a Software Stack Pointer. The pointer always points to the first available free word and grows from lower to higher addresses. It pre-decrements for stack pops and post-increments for stack pushes, as depicted in Figure 4-4.

For a PC push during any CALL instruction, the MSB of the PC is zero-extended before the push, ensuring that the MSB is always clear.

Note:	A PC push during exception processing will concatenate the SRL register to the MSB of the PC prior to the push.

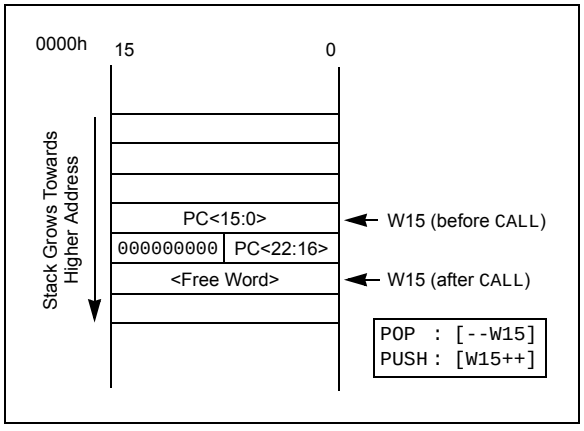
The Stack Pointer Limit Value (SPLIM) register, associated with the Stack Pointer, sets an upper address boundary for the stack. SPLIM is uninitialized at Reset. As is the case for the Stack Pointer, SPLIM<0> is forced to '0' as all stack operations must be word-aligned. Whenever an EA is generated using W15 as a source or destination pointer, the resulting address is compared with the value in SPLIM. If the contents of the Stack Pointer (W15) and the SPLIM register are equal, and a push operation is performed, a stack error trap will not occur. The stack error trap will occur on a subsequent push operation.

Thus, for example, if it is desirable to cause a stack error trap when the stack grows beyond address, 0DF6 in RAM, initialize the SPLIM with the value, 0DF4.

Similarly, a Stack Pointer underflow (stack error) trap is generated when the Stack Pointer address is found to be less than 0800h. This prevents the stack from interfering with the Special Function Register (SFR) space.

Note:	A write to the SPLIM register should not be immediately followed by an indirect read operation using W15.

FIGURE 4-4: CALL STACK FRAME



4.3 Interfacing Program and Data Memory Spaces

The PIC24F architecture uses a 24-bit-wide program space and 16-bit-wide Data Space (DS). The architecture is also a modified Harvard scheme, meaning that data can also be present in the program space. To use this data successfully, it must be accessed in a way that preserves the alignment of information in both spaces.

Apart from the normal execution, the PIC24F architecture provides two methods by which the program space can be accessed during operation:

- Using table instructions to access individual bytes or words anywhere in the program space
- Remapping a portion of the program space into the Data Space, PSV

Table instructions allow an application to read or write small areas of the program memory. This makes the method ideal for accessing data tables that need to be updated from time to time. It also allows access to all bytes of the program word. The remapping method allows an application to access a large block of data on a read-only basis, which is ideal for look ups from a large table of static data. It can only access the least significant word (lsw) of the program word.

4.3.1 ADDRESSING PROGRAM SPACE

Since the address ranges for the data and program spaces are 16 and 24 bits, respectively, a method is needed to create a 23-bit or 24-bit program address from 16-bit data registers. The solution depends on the interface method to be used.

For table operations, the 8-bit Table Memory Page Address register (TBLPAG) is used to define a 32K word region within the program space. This is concatenated with a 16-bit EA to arrive at a full 24-bit program space address. In this format, the Most Significant bit (MSb) of TBLPAG is used to determine if the operation occurs in the user memory (TBLPAG<7> = 0) or the configuration memory (TBLPAG<7> = 1).

For remapping operations, the 8-bit Program Space Visibility Page Address register (PSVPAG) is used to define a 16K word page in the program space. When the MSb of the EA is '1', PSVPAG is concatenated with the lower 15 bits of the EA to form a 23-bit program space address. Unlike the table operations, this limits remapping operations strictly to the user memory area.

See Table 4-35 and Figure 4-5 to know how the program EA is created for table operations and remapping accesses from the data EA. Here, P<23:0> refers to a program space word, whereas D<15:0> refers to a Data Space word.

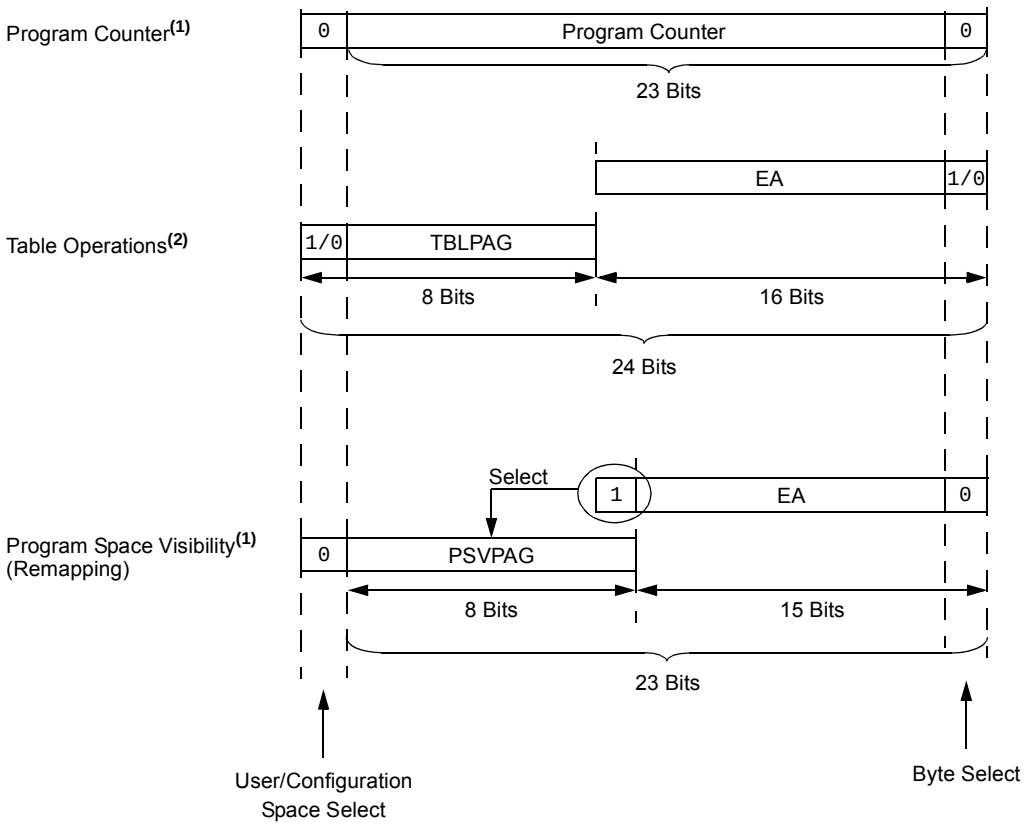
TABLE 4-35: PROGRAM SPACE ADDRESS CONSTRUCTION

Access Type	Access Space	Program Space Address				
		<23>	<22:16>	<15>	<14:1>	<0>
Instruction Access (Code Execution)	User	0	PC<22:1>			0
		0xx xxxx xxxx xxxx xxxx xxx0				
TBLRD/TBLWT (Byte/Word Read/Write)	User	TBLPAG<7:0>		Data EA<15:0>		
		0xxx xxxx		xxxx xxxx xxxx xxxx		
	Configuration	TBLPAG<7:0>		Data EA<15:0>		
		1xxx xxxx		xxxx xxxx xxxx xxxx		
Program Space Visibility (Block Remap/Read)	User	0	PSVPAG<7:0>(2)		Data EA<14:0>(1)	
		0	xxxx xxxx		xxx xxxx xxxx xxxx	

Note 1: Data EA<15> is always '1' in this case, but is not used in calculating the program space address. Bit 15 of the address is PSVPAG<0>.

2: PSVPAG can have only two values ('00' to access program memory and FF to access data EEPROM) on the PIC24F16KM family.

FIGURE 4-5: DATA ACCESS FROM PROGRAM SPACE ADDRESS GENERATION



Note 1: The LSb of program space addresses is always fixed as '0' in order to maintain word alignment of data in the program and Data Spaces.

2: Table operations are not required to be word-aligned. Table Read operations are permitted in the configuration memory space.

4.3.2 DATA ACCESS FROM PROGRAM MEMORY AND DATA EEPROM MEMORY USING TABLE INSTRUCTIONS

The TBLRDL and TBLWTL instructions offer a direct method of reading or writing the lower word of any address within the program memory without going through Data Space. It also offers a direct method of reading or writing a word of any address within data EEPROM memory. The TBLRDH and TBLWTH instructions are the only method to read or write the upper 8 bits of a program space word as data.

Note: The TBLRDH and TBLWTH instructions are not used while accessing data EEPROM memory.

The PC is incremented by 2 for each successive 24-bit program word. This allows program memory addresses to directly map to Data Space addresses. Program memory can thus be regarded as two 16-bit, word-wide address spaces, residing side by side, each with the same address range. TBLRDL and TBLWTL access the space which contains the least significant data word, and TBLRDH and TBLWTH access the space which contains the upper data byte.

Two table instructions are provided to move byte or word-sized (16-bit) data to and from program space. Both function as either byte or word operations.

1. TBLRDL (Table Read Low): In Word mode, it maps the lower word of the program space location (P<15:0>) to a data address (D<15:0>).

   In Byte mode, either the upper or lower byte of the lower program word is mapped to the lower byte of a data address. The upper byte is selected when byte select is '1'; the lower byte is selected when it is '0'.

2. TBLRDH (Table Read High): In Word mode, it maps the entire upper word of a program address (P<23:16>) to a data address. Note that D<15:8>, the 'phantom' byte, will always be '0'.

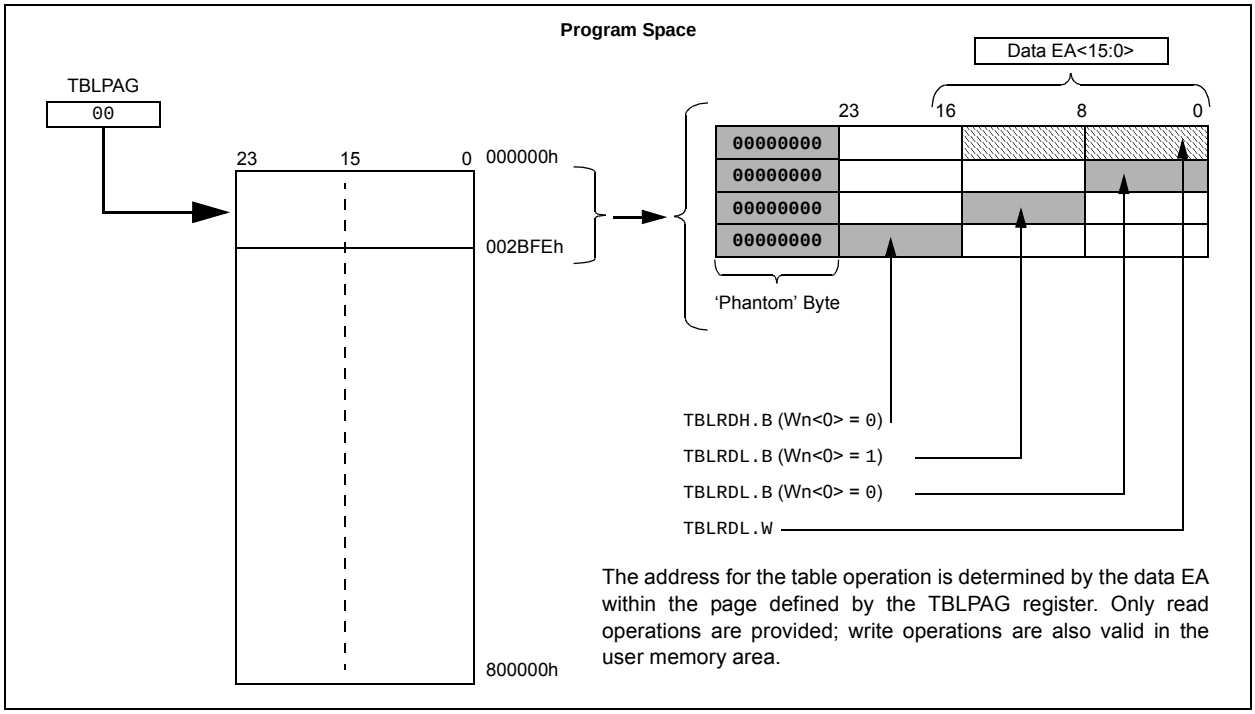
   In Byte mode, it maps the upper or lower byte of the program word to D<7:0> of the data address, as above. Note that the data will always be '0' when the upper 'phantom' byte is selected (Byte Select = 1).

In a similar fashion, two table instructions, TBLWTH and TBLWTL, are used to write individual bytes or words to a program space address. The details of their operation are explained in **Section 5.0 "Flash Program Memory"**.

For all table operations, the area of program memory space to be accessed is determined by the Table Memory Page Address register (TBLPAG). TBLPAG covers the entire program memory space of the device, including user and configuration spaces. When TBLPAG<7> = 0, the table page is located in the user memory space. When TBLPAG<7> = 1, the page is located in configuration space.

Note: Only Table Read operations will execute in the configuration memory space, and only then, in implemented areas, such as the Device ID. Table Write operations are not allowed.

FIGURE 4-6: ACCESSING PROGRAM MEMORY WITH TABLE INSTRUCTIONS



The address for the table operation is determined by the data EA within the page defined by the TBLPAG register. Only read operations are provided; write operations are also valid in the user memory area.

7.1 Clock Source Selection at Reset

If clock switching is enabled, the system clock source at device Reset is chosen, as shown in Table 7-2. If clock switching is disabled, the system clock source is always selected according to the Oscillator Configuration bits. For more information, see **Section 9.0 "Oscillator Configuration"**.

TABLE 7-2: OSCILLATOR SELECTION vs. TYPE OF RESET (CLOCK SWITCHING ENABLED)

Reset Type	Clock Source Determinant
POR	FNOSC<2:0> Configuration bits (FOSCSEL<2:0>)
BOR	
$\overline{\text{MCLR}}$	COSC<2:0> Control bits (OSCCON<14:12>)
WDTO	
SWR	

7.2 Device Reset Times

The Reset times for various types of device Reset are summarized in Table 7-3. Note that the system Reset signal, $\overline{\text{SYSRST}}$, is released after the POR and PWRT delay times expire.

The time at which the device actually begins to execute code will also depend on the system oscillator delays, which include the Oscillator Start-up Timer (OST) and the PLL lock time. The OST and PLL lock times occur in parallel with the applicable $\overline{\text{SYSRST}}$ delay times.

The FSCM delay determines the time at which the FSCM begins to monitor the system clock source after the $\overline{\text{SYSRST}}$ signal is released.

TABLE 7-3: RESET DELAY TIMES FOR VARIOUS DEVICE RESETS

Reset Type	Clock Source	$\overline{\text{SYSRST}}$ Delay	System Clock Delay	Notes
POR(6)	EC	TPOR + TPWRT	—	1, 2
	FRC, FRCDIV	TPOR + TPWRT	TFRC	1, 2, 3
	LPRC	TPOR + TPWRT	TLPRC	1, 2, 3
	ECPLL	TPOR + TPWRT	TLOCK	1, 2, 4
	FRCPLL	TPOR + TPWRT	TFRC + TLOCK	1, 2, 3, 4
	XT, HS, SOSC	TPOR+ TPWRT	TOST	1, 2, 5
	XTPLL, HSPLL	TPOR + TPWRT	TOST + TLOCK	1, 2, 4, 5
BOR	EC	TPWRT	—	2
	FRC, FRCDIV	TPWRT	TFRC	2, 3
	LPRC	TPWRT	TLPRC	2, 3
	ECPLL	TPWRT	TLOCK	2, 4
	FRCPLL	TPWRT	TFRC + TLOCK	2, 3, 4
	XT, HS, SOSC	TPWRT	TOST	2, 5
	XTPLL, HSPLL	TPWRT	TFRC + TLOCK	2, 3, 4
All Others	Any Clock	—	—	None

Note 1: TPOR = Power-on Reset delay.

2: TPWRT = 64 ms nominal if the Power-up Timer is enabled; otherwise, it is zero.

3: TFRC and TLPRC = RC Oscillator start-up times.

4: TLOCK = PLL Lock time.

5: TOST = Oscillator Start-up Timer (OST). A 10-bit counter waits 1024 oscillator periods before releasing the oscillator clock to the system.

6: If Two-Speed Start-up is enabled, regardless of the Primary Oscillator selected, the device starts with FRC, and in such cases, FRC start-up time is valid.

Note: For detailed operating frequency and timing specifications, see **Section 27.0 "Electrical Characteristics"**.

REGISTER 8-14: IEC2: INTERRUPT ENABLE CONTROL REGISTER 2

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	U-0
—	—	—	—	—	—	CCT5IE	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-10 **Unimplemented:** Read as '0'

bit 9 **CCT5IE:** Capture/Compare 5 Timer Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 8-0 **Unimplemented:** Read as '0'

REGISTER 8-15: IEC3: INTERRUPT ENABLE CONTROL REGISTER 3

U-0	R/W-0	U-0	U-0	U-0	U-0	U-0	U-0
—	RTCIE	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	U-0
—	—	—	—	—	BCL2IE	SSP2IE	—
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 15 **Unimplemented:** Read as '0'

bit 14 **RTCIE:** Real-Time Clock and Calendar Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 13-3 **Unimplemented:** Read as '0'

bit 2 **BCL2IE:** MSSP2 I^2C™ Bus Collision Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 1 **SSP2IE:** MSSP2 SPI/I^2C Event Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

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

10.5 Doze Mode

Generally, changing clock speed and invoking one of the power-saving modes are the preferred strategies for reducing power consumption. There may be circumstances, however, where this is not practical. For example, it may be necessary for an application to maintain uninterrupted synchronous communication, even while it is doing nothing else. Reducing system clock speed may introduce communication errors, while using a power-saving mode may stop communications completely.

Doze mode is a simple and effective alternative method to reduce power consumption while the device is still executing code. In this mode, the system clock continues to operate from the same source and at the same speed. Peripheral modules continue to be clocked at the same speed, while the CPU clock speed is reduced. Synchronization between the two clock domains is maintained, allowing the peripherals to access the SFRs while the CPU executes code at a slower rate.

Doze mode is enabled by setting the DOZEN bit (CLKDIV<11>). The ratio between peripheral and core clock speed is determined by the DOZE<2:0> bits (CLKDIV<14:12>). There are eight possible configurations, from 1:1 to 1:128, with 1:1 being the default.

It is also possible to use Doze mode to selectively reduce power consumption in event driven applications. This allows clock-sensitive functions, such as synchronous communications, to continue without interruption. Meanwhile, the CPU Idles, waiting for something to invoke an interrupt routine. Enabling the automatic return to full-speed CPU operation on interrupts is enabled by setting the ROI bit (CLKDIV<15>). By default, interrupt events have no effect on Doze mode operation.

10.6 Selective Peripheral Module Control

Idle and Doze modes allow users to substantially reduce power consumption by slowing or stopping the CPU clock. Even so, peripheral modules still remain clocked, and thus, consume power. There may be cases where the application needs what these modes do not provide: the allocation of power resources to CPU processing with minimal power consumption from the peripherals.

PIC24F devices address this requirement by allowing peripheral modules to be selectively disabled, reducing or eliminating their power consumption. This can be done with two control bits:

- The Peripheral Enable bit, generically named, “XXXEN”, located in the module’s main control SFR.
- The Peripheral Module Disable (PMD) bit, generically named, “XXXMD”, located in one of the PMDx Control registers.

Both bits have similar functions in enabling or disabling its associated module. Setting the PMDx bits for a module, disables all clock sources to that module, reducing its power consumption to an absolute minimum. In this state, the control and status registers associated with the peripheral will also be disabled, so writes to those registers will have no effect, and read values will be invalid. Many peripheral modules have a corresponding PMDx bit.

In contrast, disabling a module by clearing its XXXEN bit, disables its functionality, but leaves its registers available to be read and written to. Power consumption is reduced, but not by as much as when the PMDx bits are used. Most peripheral modules have an enable bit; exceptions include capture, compare and RTCC.

To achieve more selective power savings, peripheral modules can also be selectively disabled when the device enters Idle mode. This is done through the control bit of the generic name format, “XXXIDL”. By default, all modules that can operate during Idle mode will do so. Using the disable on Idle feature disables the module while in Idle mode, allowing further reduction of power consumption during Idle mode, enhancing power savings for extremely critical power applications.

13.4 Input Capture Mode

Input Capture mode is used to capture a timer value from an independent timer base upon an event on an input pin or other internal Trigger source. The input capture features are useful in applications requiring frequency (time period) and pulse measurement. Figure 13-6 depicts a simplified block diagram of Input Capture mode.

Input Capture mode uses a dedicated 16/32-bit, synchronous, up counting timer for the capture function. The timer value is written to the FIFO when a capture event occurs. The internal value may be read (with a synchronization delay) using the CCPxTMRH/L register.

To use Input Capture mode, the CCSEL bit (CCPxCON1L<4>) must be set. The T32 and the MOD<3:0> bits are used to select the proper Capture mode, as shown in Table 13-4.

TABLE 13-4: INPUT CAPTURE MODES

MOD<3:0> (CCPxCON1L<3:0>)	T32 (CCPxCON1L<5>)	Operating Mode
0000	0	Edge Detect (16-bit capture)
0000	1	Edge Detect (32-bit capture)
0001	0	Every Rising (16-bit capture)
0001	1	Every Rising (32-bit capture)
0010	0	Every Falling (16-bit capture)
0010	1	Every Falling (32-bit capture)
0011	0	Every Rise/Fall (16-bit capture)
0011	1	Every Rise/Fall (32-bit capture)
0100	0	Every 4th Rising (16-bit capture)
0100	1	Every 4th Rising (32-bit capture)
0101	0	Every 16th Rising (16-bit capture)
0101	1	Every 16th Rising (32-bit capture)

FIGURE 13-6: INPUT CAPTURE x BLOCK DIAGRAM

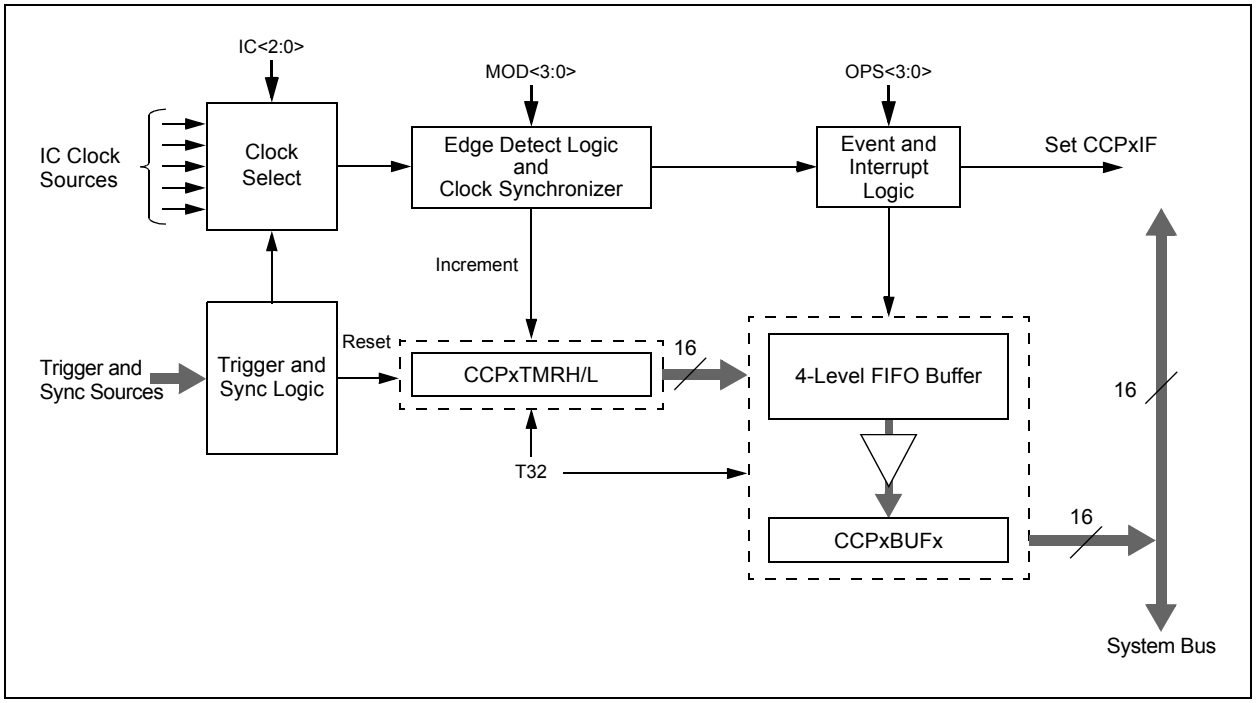
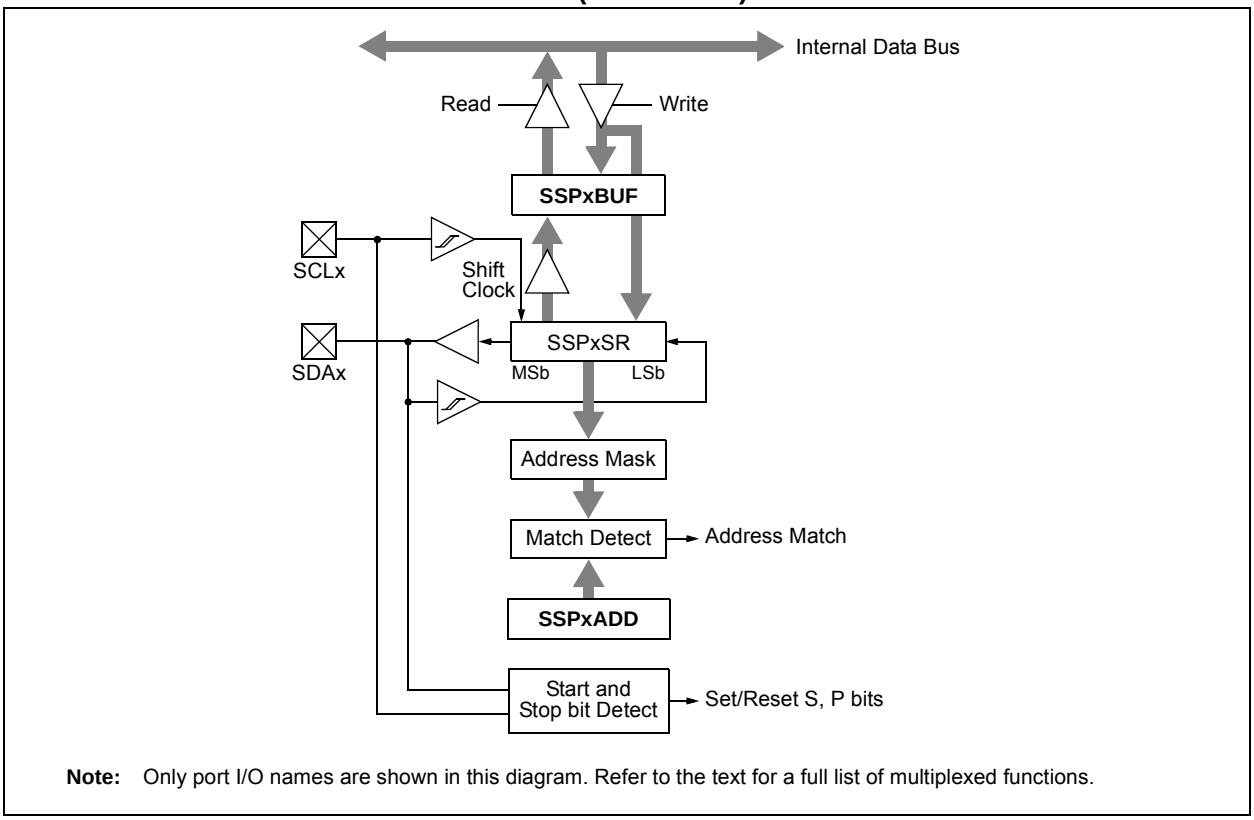


TABLE 13-6: SYNCHRONIZATION SOURCES

SYNC<4:0>	Synchronization Source
00000	None; Timer with Rollover on CCPxPR Match or FFFFh
00001	MCCP1 or SCCP1 Sync Output
00010	MCCP2 or SCCP2 Sync Output
00011	MCCP3 or SCCP3 Sync Output
00100	MCCP4 or SCCP4 Sync Output
00101	MCCP5 or SCCP5 Sync Output
00110 to 01010	Unused
01011	Timer1 Sync Output(1)
01100 to 10000	Unused
10001	CLC1 Output(1)
10010	CLC2 Output(1)
10011 to 11010	Unused
11011	A/D(1)
11110	Unused
11111	None; Timer with Auto-Rollover (FFFFh → 0000h)

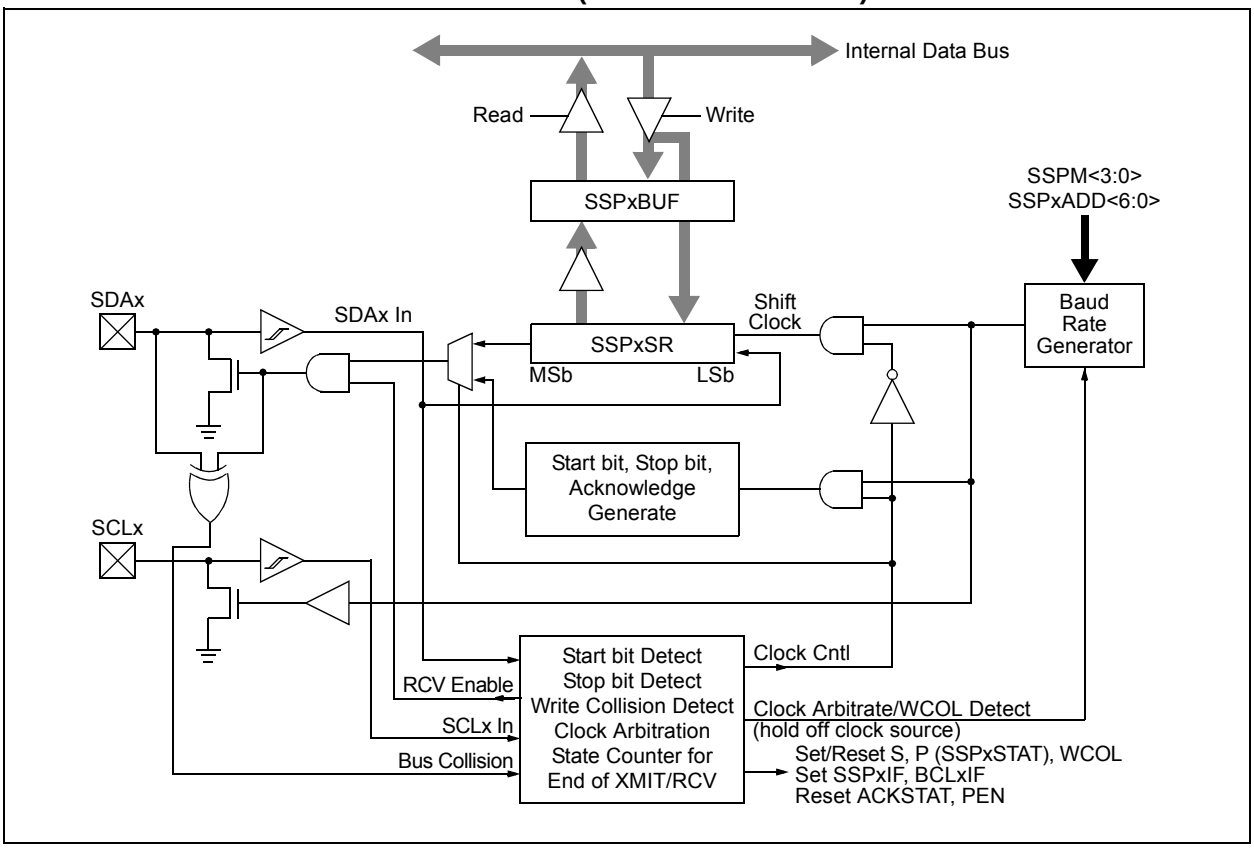
Note 1: These sources are only available when the source module is being used in a Synchronous mode.

FIGURE 14-3: MSSPx BLOCK DIAGRAM (I^2C™ MODE)



Note: Only port I/O names are shown in this diagram. Refer to the text for a full list of multiplexed functions.

FIGURE 14-4: MSSPx BLOCK DIAGRAM (I^2C™ MASTER MODE)



REGISTER 15-2: UxSTA: UARTx STATUS AND CONTROL REGISTER (CONTINUED)

bit 5 **ADDEN:** Address Character Detect bit (bit 8 of received data = 1)
1 = Address Detect mode is enabled; if 9-bit mode is not selected, this does not take effect
0 = Address Detect mode is disabled

bit 4 **RIDLE:** Receiver Idle bit (read-only)
1 = Receiver is Idle
0 = Receiver is active

bit 3 **PERR:** Parity Error Status bit (read-only)
1 = Parity error has been detected for the current character (character at the top of the receive FIFO)
0 = Parity error has not been detected

bit 2 **FERR:** Framing Error Status bit (read-only)
1 = Framing error has been detected for the current character (character at the top of the receive FIFO)
0 = Framing error has not been detected

bit 1 **OERR:** Receive Buffer Overrun Error Status bit (clear/read-only)
1 = Receive buffer has overflowed
0 = Receive buffer has not overflowed (clearing a previously set OERR bit (1 → 0 transition) will reset the receiver buffer and the RSR to the empty state)

bit 0 **URXDA:** UARTx Receive Buffer Data Available bit (read-only)
1 = Receive buffer has data; at least one more characters can be read
0 = Receive buffer is empty

REGISTER 16-1: RCFGCAL: RTCC CALIBRATION AND CONFIGURATION REGISTER(1) (CONTINUED)

bit 7-0 **CAL<7:0>:** RTC Drift Calibration bits

01111111 = Maximum positive adjustment; adds 508 RTC clock pulses every one minute
.
.
.
00000001 = Minimum positive adjustment; adds 4 RTC clock pulses every one minute
00000000 = No adjustment
11111111 = Minimum negative adjustment; subtracts 4 RTC clock pulses every one minute
.
.
.
10000000 = Maximum negative adjustment; subtracts 512 RTC clock pulses every one minute

Note 1: The RCFGCAL register is only affected by a POR.

2: A write to the RTCEN bit is only allowed when RTCWREN = 1.

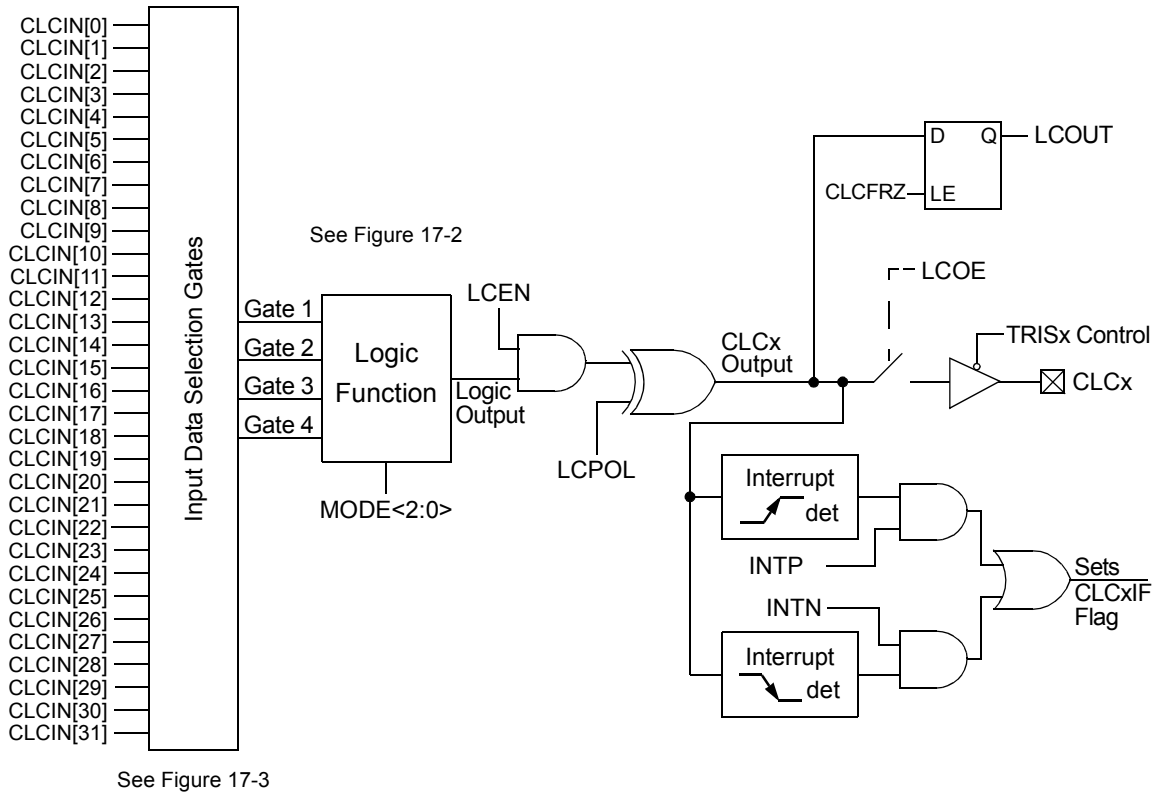
3: This bit is read-only; it is cleared to '0' on a write to the lower half of the MINSEC register.

17.0 CONFIGURABLE LOGIC CELL (CLC)

The Configurable Logic Cell (CLC) module allows the user to specify combinations of signals as inputs to a logic function and to use the logic output to control other peripherals or I/O pins. This provides greater flexibility and potential in embedded designs since the CLC module can operate outside the limitations of software execution and supports a vast amount of output designs.

There are four input gates to the selected logic function. These four input gates select from a pool of up to 32 signals that are selected using four data source selection multiplexers. Figure 17-1 shows an overview of the module. Figure 17-3 shows the details of the data source multiplexers and logic input gate connections.

FIGURE 17-1: CLCx MODULE



Note: All register bits shown in this figure can be found in the CLCxCONL register.

REGISTER 19-3: AD1CON3: A/D CONTROL REGISTER 3

R/W-0	R-0	r-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADRC	EXTSAM	r	SAMC4	SAMC3	SAMC2	SAMC1	SAMC0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADCS7	ADCS6	ADCS5	ADCS4	ADCS3	ADCS2	ADCS1	ADCS0
bit 7							bit 0

Legend:	r = Reserved bit		
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 15 **ADRC:** A/D Conversion Clock Source bit
1 = RC clock
0 = Clock is derived from the system clock

bit 14 **EXTSAM:** Extended Sampling Time bit
1 = A/D is still sampling after SAMP = 0
0 = A/D is finished sampling

bit 13 **Reserved:** Maintain as '0'

bit 12-8 **SAMC<4:0>:** Auto-Sample Time Select bits
11111 = 31 TAD
•
•
•
00001 = 1 TAD
00000 = 0 TAD

bit 7-0 **ADCS<7:0>:** A/D Conversion Clock Select bits
11111111-01000000 = Reserved
00111111 = 64 * TCY = TAD
•
•
•
00000001 = 2 * TCY = TAD
00000000 = TCY = TAD

REGISTER 21-1: AMPxCON: OP AMP x CONTROL REGISTER(1)

R/W-0	U-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
AMPEN	—	AMPSIDL	AMPSLP	—	—	—	—
bit 15							bit 8

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SPDSEL	—	NINSEL2	NINSEL1	NINSEL0	PINSEL2	PINSEL1	PINSEL0
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 15 **AMPEN:** Op Amp x Control Module Enable bit
1 = Module is enabled
0 = Module is disabled

bit 14 **Unimplemented:** Read as '0'

bit 13 **AMPSIDL:** Op Amp x Peripheral Stop in Idle Mode bit
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode

bit 12 **AMPSLP:** Op Amp x Peripheral Enabled in Sleep Mode bit
1 = Continues module operation when device enters Sleep mode
0 = Discontinues module operation in Sleep mode

bit 11-8 **Unimplemented:** Read as '0'

bit 7 **SPDSEL:** Op Amp x Power/Speed Select bit
1 = Higher power and bandwidth (faster response time)
0 = Lower power and bandwidth (slower response time)

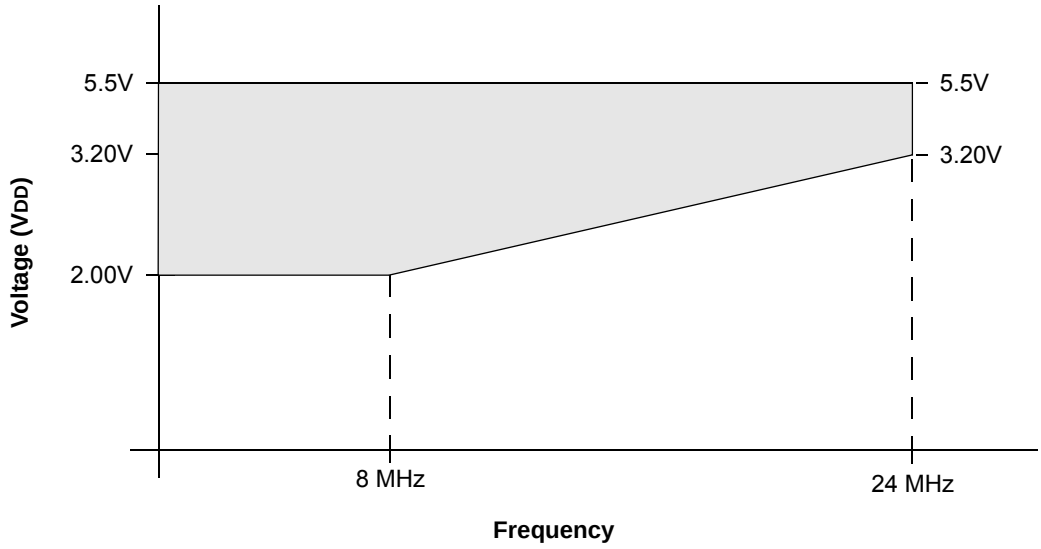
bit 6 **Unimplemented:** Read as '0'

bit 5-3 **NINSEL<2:0>:** Negative Op Amp Input Select bits
111 = Reserved; do not use
110 = Reserved; do not use
101 = Op amp negative input is connected to the op amp output (voltage follower)
100 = Reserved; do not use
011 = Reserved; do not use
010 = Op amp negative input is connected to the OAxIND pin
001 = Op amp negative input is connected to the OAxINB pin
000 = Op amp negative input is connected to AVSS

bit 2-0 **PINSEL<2:0>:** Positive Op Amp Input Select bits
111 = Op amp positive input is connected to the output of the A/D input multiplexer
110 = Reserved; do not use
101 = Op amp positive input is connected to the DAC1 output for OA1 (DAC2 output for OA2)
100 = Reserved; do not use
011 = Reserved; do not use
010 = Op amp positive input is connected to the OAxINC pin
001 = Op amp positive input is connected to the OAxINA pin
000 = Op amp positive input is connected to AVSS

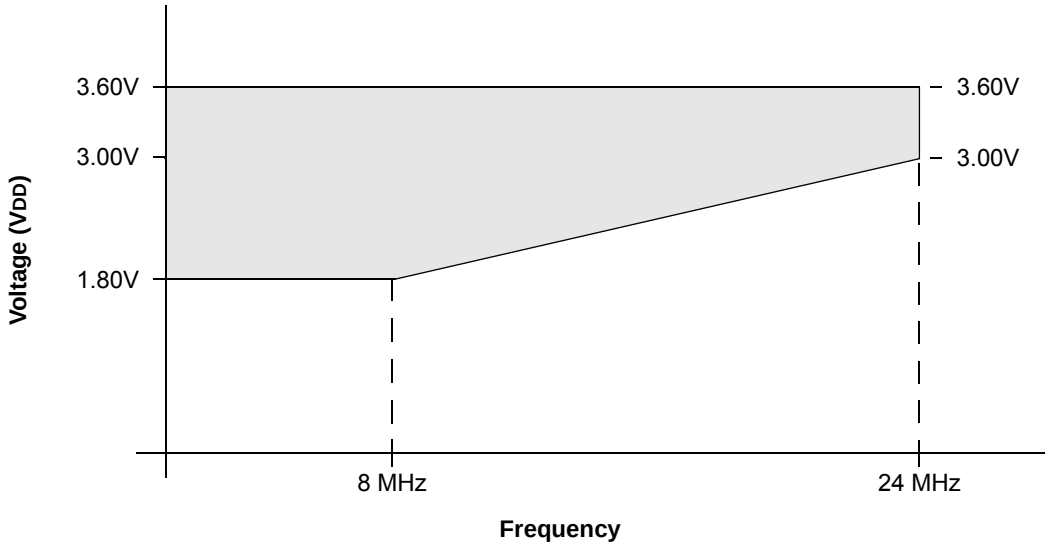
Note 1: This register is available only on PIC24F(V)16KM2XX devices.

FIGURE 27-3: PIC24FV16KM204 FAMILY VOLTAGE-FREQUENCY GRAPH (EXTENDED)



Note: For frequencies between 8 MHz and 24 MHz, FMAX = 13.33 MHz * (VDD – 2.0) + 8 MHz.

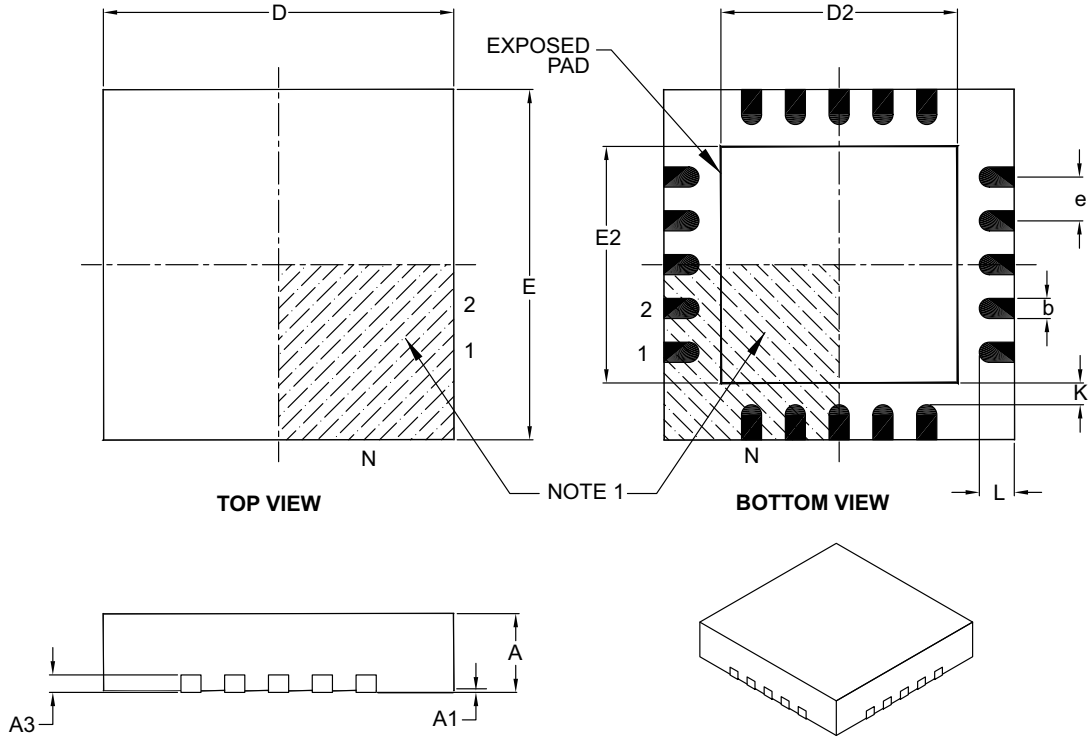
FIGURE 27-4: PIC24F16KM204 FAMILY VOLTAGE-FREQUENCY GRAPH (EXTENDED)



Note: For frequencies between 8 MHz and 24 MHz, FMAX = 13.33 MHz * (VDD – 1.8) + 8 MHz.

20-Lead Plastic Quad Flat, No Lead Package (ML) – 4x4x0.9 mm Body [QFN]

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



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		20	
Pitch	e		0.50 BSC	
Overall Height	A	0.80	0.90	1.00
Standoff	A1	0.00	0.02	0.05
Contact Thickness	A3		0.20 REF	
Overall Width	E		4.00 BSC	
Exposed Pad Width	E2	2.60	2.70	2.80
Overall Length	D		4.00 BSC	
Exposed Pad Length	D2	2.60	2.70	2.80
Contact Width	b	0.18	0.25	0.30
Contact Length	L	0.30	0.40	0.50
Contact-to-Exposed Pad	K	0.20	–	–

Notes:

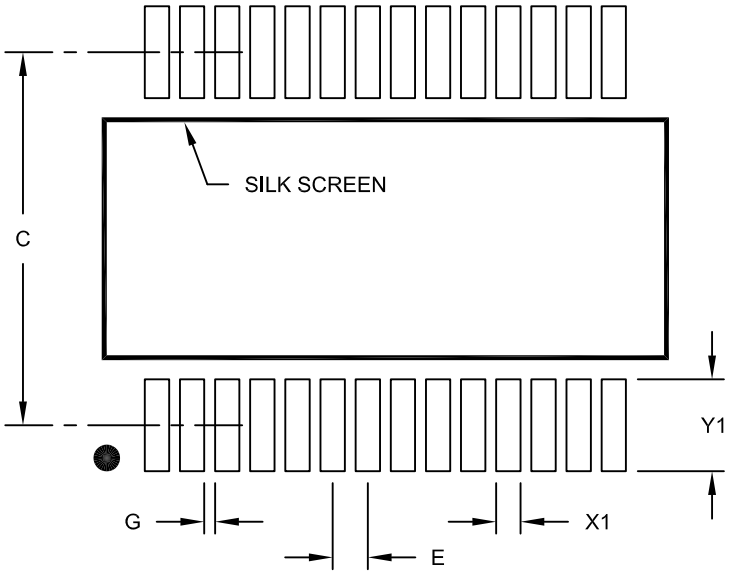
1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated.
3. Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-126B

28-Lead Plastic Shrink Small Outline (SS) - 5.30 mm Body [SSOP]

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	0.65 BSC		
Contact Pad Spacing	C		7.20	
Contact Pad Width (X28)	X1			0.45
Contact Pad Length (X28)	Y1			1.75
Distance Between Pads	G	0.20		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2073A

NOTES: