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	Obsolete
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	24
Program Memory Size	8KB (2.75K x 24)
Program Memory Type	FLASH
EEPROM Size	512 x 8
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 19x10b/12b
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/pic24f08km102-e-ss

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.

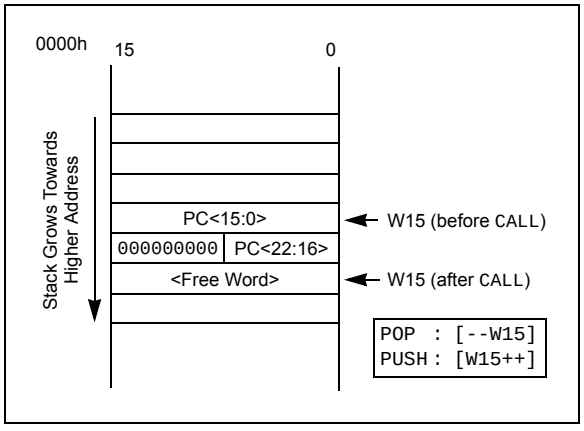
The Stack Pointer Limit Value (SPLIM) register, associated with the Stack Pointer, sets an upper address boundary for the stack. SPLIM is uninitialized at Reset. As is the case for the Stack Pointer, SPLIM<0> is forced to '0' as all stack operations must be word-aligned. Whenever an EA is generated using 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.

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'.

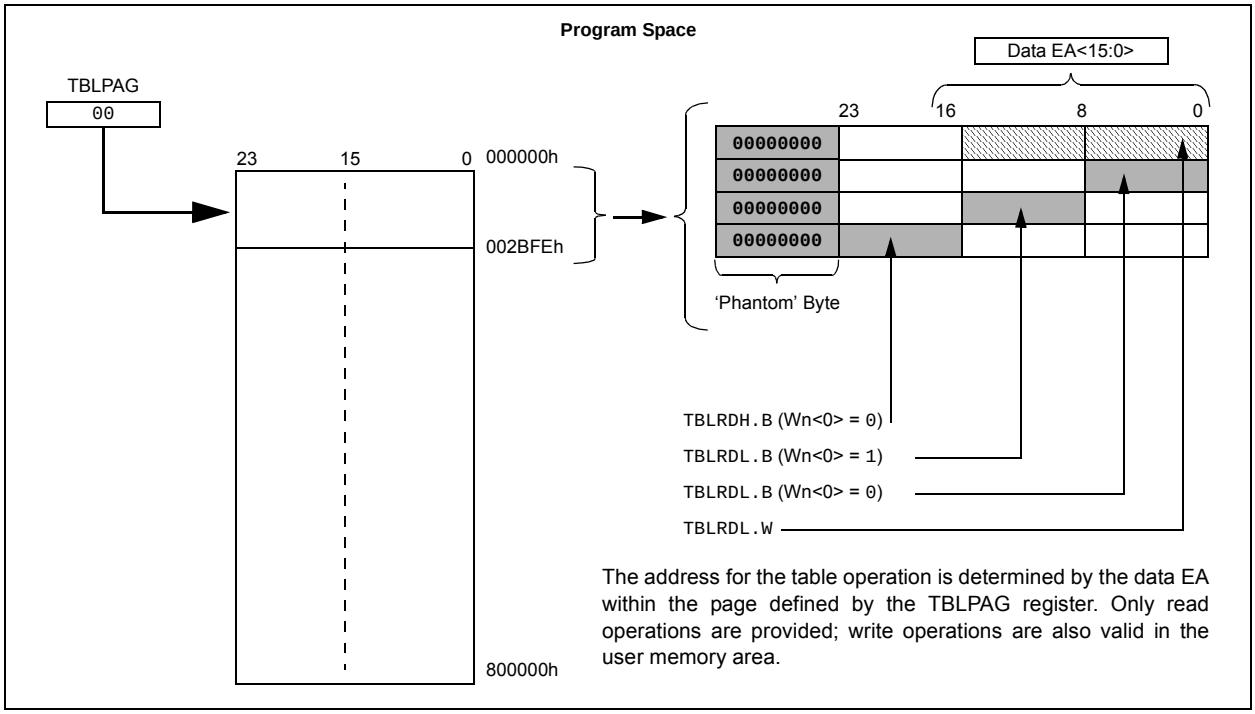
   In Byte mode, it maps the upper or lower byte of the program word to D<7:0> of the data address, as above. Note that the data will always be '0' when the upper 'phantom' byte is selected (Byte Select = 1).

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.

REGISTER 8-10: IFS5: INTERRUPT FLAG STATUS REGISTER 5

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0, HS
—	—	—	—	—	—	—	ULPWUIF
bit 7							bit 0

Legend:	HS = Hardware Settable 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 15-1 **Unimplemented:** Read as '0'

bit 0 **ULPWUIF:** Ultra Low-Power Wake-up Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

REGISTER 8-11: IFS6: INTERRUPT FLAG STATUS REGISTER 6

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0, HS	R/W-0, HS
—	—	—	—	—	—	CLC2IF	CLC1IF
bit 7							bit 0

Legend:	HS = Hardware Settable 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 15-2 **Unimplemented:** Read as '0'

bit 1 **CLC2IF:** Configurable Logic Cell 2 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 0 **CLC1IF:** Configurable Logic Cell 1 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

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'

REGISTER 8-17: IEC5: INTERRUPT ENABLE CONTROL REGISTER 5

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	ULPWUIE
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 15-1 **Unimplemented:** Read as ‘0’

bit 0 **ULPWUIE:** Ultra Low-Power Wake-up Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

REGISTER 8-18: IEC6: INTERRUPT ENABLE CONTROL REGISTER 5

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	CLC2IE	CLC1IE
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 15-2 **Unimplemented:** Read as ‘0’

bit 1 **CLC2IE:** Configurable Logic Cell 2 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 0 **CLC1IE:** Configurable Logic Cell 1 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

REGISTER 8-21: IPC2: INTERRUPT PRIORITY CONTROL REGISTER 2

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	U1RXIP2	U1RXIP1	U1RXIP0	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	CCT2IP2	CCT2IP1	CCT2IP0
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 14-12 **U1RXIP<2:0>:** UART1 Receiver Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 11-3 **Unimplemented:** Read as ‘0’

bit 2-0 **CCT2IP<2:0>:** Capture/Compare 2 Timer Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

REGISTER 8-29: IPC15: INTERRUPT PRIORITY CONTROL REGISTER 15

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	RTCIP2	RTCIP1	RTCIP0
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 15-11 **Unimplemented:** Read as ‘0’

bit 10-8 **RTCIP<2:0>:** Real-Time Clock and Calendar Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

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

9.1 CPU Clocking Scheme

The system clock source can be provided by one of four sources:

- Primary Oscillator (POSC) on the OSCI and OSCO pins
- Secondary Oscillator (SOSC) on the SOSCI and SOSCO pins

  The PIC24FXXXXX family devices consist of two types of secondary oscillator:

  - High-Power Secondary Oscillator
  - Low-Power Secondary Oscillator

  These can be selected by using the SOSCSEL (FOSC<5>) bit.
- Fast Internal RC (FRC) Oscillator:
  - 8 MHz FRC Oscillator
  - 500 kHz Lower Power FRC Oscillator
- Low-Power Internal RC (LPRC) Oscillator with two modes:
  - High-Power/High-Accuracy mode
  - Low-Power/Low-Accuracy mode

The Primary Oscillator and 8 MHz FRC sources have the option of using the internal 4x PLL. The frequency of the FRC clock source can optionally be reduced by the programmable clock divider. The selected clock source generates the processor and peripheral clock sources.

The processor clock source is divided by two to produce the internal instruction cycle clock, FCY. In this document, the instruction cycle clock is also denoted by FOSC/2. The internal instruction cycle clock, FOSC/2, can be provided on the OSCO I/O pin for some operating modes of the Primary Oscillator.

9.2 Initial Configuration on POR

The oscillator source (and operating mode) that is used at a device Power-on Reset (POR) event is selected using Configuration bit settings. The Oscillator Configuration bit settings are located in the Configuration registers in the program memory (for more information, see **Section 25.1 "Configuration Bits"**). The Primary Oscillator Configuration bits, POSCMD<1:0> (FOSC<1:0>), and the Initial Oscillator Select Configuration bits, FNOSC<2:0> (FOSCSEL<2:0>), select the oscillator source that is used at a POR. The FRC Primary Oscillator with Postscaler (FRCDIV) is the default (unprogrammed) selection. The Secondary Oscillator, or one of the internal oscillators, may be chosen by programming these bit locations. The EC mode Frequency Range Configuration bits, POSCFREQ<1:0> (FOSC<4:3>), optimize power consumption when running in EC mode. The default configuration is "frequency range is greater than 8 MHz".

The Configuration bits allow users to choose between the various clock modes, shown in Table 9-1.

9.2.1 CLOCK SWITCHING MODE CONFIGURATION BITS

The FCKSM<1:0> Configuration bits (FOSC<7:6>) are used jointly to configure device clock switching and the FSCM. Clock switching is enabled only when FCKSM1 is programmed ('0'). The FSCM is enabled only when FCKSM<1:0> are both programmed ('00').

TABLE 9-1: CONFIGURATION BIT VALUES FOR CLOCK SELECTION

Oscillator Mode	Oscillator Source	POSCMD<1:0>	FNOSC<2:0>	Notes
8 MHz FRC Oscillator with Postscaler (FRCDIV)	Internal	11	111	**1, 2**
500 kHz FRC Oscillator with Postscaler (LPFRCDIV)	Internal	11	110	**1**
Low-Power RC Oscillator (LPRC)	Internal	11	101	**1**
Secondary (Timer1) Oscillator (SOSC)	Secondary	00	100	**1**
Primary Oscillator (HS) with PLL Module (HSPLL)	Primary	10	011	
Primary Oscillator (EC) with PLL Module (ECPLL)	Primary	00	011	
Primary Oscillator (HS)	Primary	10	010	
Primary Oscillator (XT)	Primary	01	010	
Primary Oscillator (EC)	Primary	00	010	
8 MHz FRC Oscillator with PLL Module (FRCPLL)	Internal	11	001	**1**
8 MHz FRC Oscillator (FRC)	Internal	11	000	**1**

Note 1: The OSCO pin function is determined by the OSCIOFNC Configuration bit.

2: This is the default oscillator mode for an unprogrammed (erased) device.

REGISTER 12-1: T1CON: TIMER1 CONTROL REGISTER

R/W-0	U-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0
TON	—	TSIDL	—	—	—	TECS1(1)	TECS0(1)
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	U-0
—	TGATE	TCKPS1	TCKPS0	—	TSYNC	TCS	—
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 **TON:** Timer1 On bit
1 = Starts 16-bit Timer1
0 = Stops 16-bit Timer1

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

bit 13 **TSIDL:** Timer1 Stop in Idle Mode bit
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode

bit 12-10 **Unimplemented:** Read as '0'

bit 9-8 **TECS<1:0>:** Timer1 Extended Clock Select bits(1)
11 = Reserved; do not use
10 = Timer1 uses the LPRC as the clock source
01 = Timer1 uses the External Clock (EC) from T1CK
00 = Timer1 uses the Secondary Oscillator (SOSC) as the clock source

bit 7 **Unimplemented:** Read as '0'

bit 6 **TGATE:** Timer1 Gated Time Accumulation Enable bit
When TCS = 1:
This bit is ignored.
When TCS = 0:
1 = Gated time accumulation is enabled
0 = Gated time accumulation is disabled

bit 5-4 **TCKPS<1:0>:** Timer1 Input Clock Prescale Select bits
11 = 1:256
10 = 1:64
01 = 1:8
00 = 1:1

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

bit 2 **TSYNC:** Timer1 External Clock Input Synchronization Select bit
When TCS = 1:
1 = Synchronizes External Clock input
0 = Does not synchronize External Clock input
When TCS = 0:
This bit is ignored.

bit 1 **TCS:** Timer1 Clock Source Select bit
1 = Timer1 clock source is selected by TECS<1:0>
0 = Internal clock (FOSC/2)

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

Note 1: The TECSx bits are valid only when TCS = 1.

14.0 MASTER SYNCHRONOUS SERIAL PORT (MSSP)

Note: This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information on MSSP, refer to the *"PIC24F Family Reference Manual"*.

The Master Synchronous Serial Port (MSSP) module is an 8-bit serial interface, useful for communicating with other peripheral or microcontroller devices. These peripheral devices may be serial EEPROMs, Shift registers, display drivers, A/D Converters, etc. The MSSP module can operate in one of two modes:

- Serial Peripheral Interface (SPI)
- Inter-Integrated Circuit (I^2C™)
  - Full Master mode
  - Slave mode (with general address call)

The SPI interface supports these modes in hardware:

- Master mode
- Slave mode
- Daisy-Chaining Operation in Slave mode
- Synchronized Slave Operation

The I^2C interface supports the following modes in hardware:

- Master mode
- Multi-Master mode
- Slave mode with 10-Bit and 7-Bit Addressing and Address Masking
- Byte NACKing
- Selectable Address and Data Hold, and Interrupt Masking

14.1 I/O Pin Configuration for SPI

In SPI Master mode, the MSSP module will assert control over any pins associated with the SDOx and SCKx outputs. This does not automatically disable other digital functions associated with the pin and may result in the module driving the digital I/O port inputs. To prevent this, the MSSP module outputs must be disconnected from their output pins while the module is in SPI Master mode. While disabling the module temporarily may be an option, it may not be a practical solution in all applications.

The SDOx and SCKx outputs for the module can be selectively disabled by using the SDOxDIS and SCKxDIS bits in the PADCFG1 register (Register 14-10). Setting the bit disconnects the corresponding output for a particular module from its assigned pin.

REGISTER 14-8: SSPxADD: MSSPx SLAVE ADDRESS/BAUD RATE GENERATOR REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
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
ADD7	ADD6	ADD5	ADD4	ADD3	ADD2	ADD1	ADD0
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 15-8 **Unimplemented:** Read as '0'

bit 7-0 **ADD<7:0>:** Slave Address/Baud Rate Generator Value bits

SPI Master and I²C™ Master modes:
Reload value for the Baud Rate Generator. Clock period is (([SPxADD] + 1) * 2)/FOSC.

I²C Slave modes:
Represents 7 or 8 bits of the slave address, depending on the addressing mode used:

7-Bit mode: Address is ADD<7:1>; ADD<0> is ignored.

10-Bit LSb mode: ADD<7:0> are the Least Significant bits of the address.

10-Bit MSb mode: ADD<2:1> are the two Most Significant bits of the address; ADD<7:3> are always '11110' as a specification requirement; ADD<0> is ignored.

REGISTER 14-9: SSPxMSK: I²C™ SLAVE ADDRESS MASK REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
MSK7	MSK6	MSK5	MSK4	MSK3	MSK2	MSK1	MSK0(1)
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 15-8 **Unimplemented:** Read as '0'

bit 7-0 **MSK<7:0>:** Slave Address Mask Select bits(1)

1 = Masking of corresponding bit of SSPxADD is enabled
0 = Masking of corresponding bit of SSPxADD is disabled

Note 1: MSK0 is not used as a mask bit in 7-bit addressing.

15.0 UNIVERSAL ASYNCHRONOUS RECEIVER TRANSMITTER (UART)

Note: This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information on the Universal Asynchronous Receiver Transmitter, refer to the *"PIC24F Family Reference Manual"*, **"UART"** (DS39708).

The Universal Asynchronous Receiver Transmitter (UART) module is one of the serial I/O modules available in this PIC24F device family. The UART is a full-duplex, asynchronous system that can communicate with peripheral devices, such as personal computers, LIN/J2602, RS-232 and RS-485 interfaces. This module also supports a hardware flow control option with the $\overline{\text{UxCTS}}$ and $\overline{\text{UxRTS}}$ pins, and also includes an IrDA® encoder and decoder.

The primary features of the UART module are:

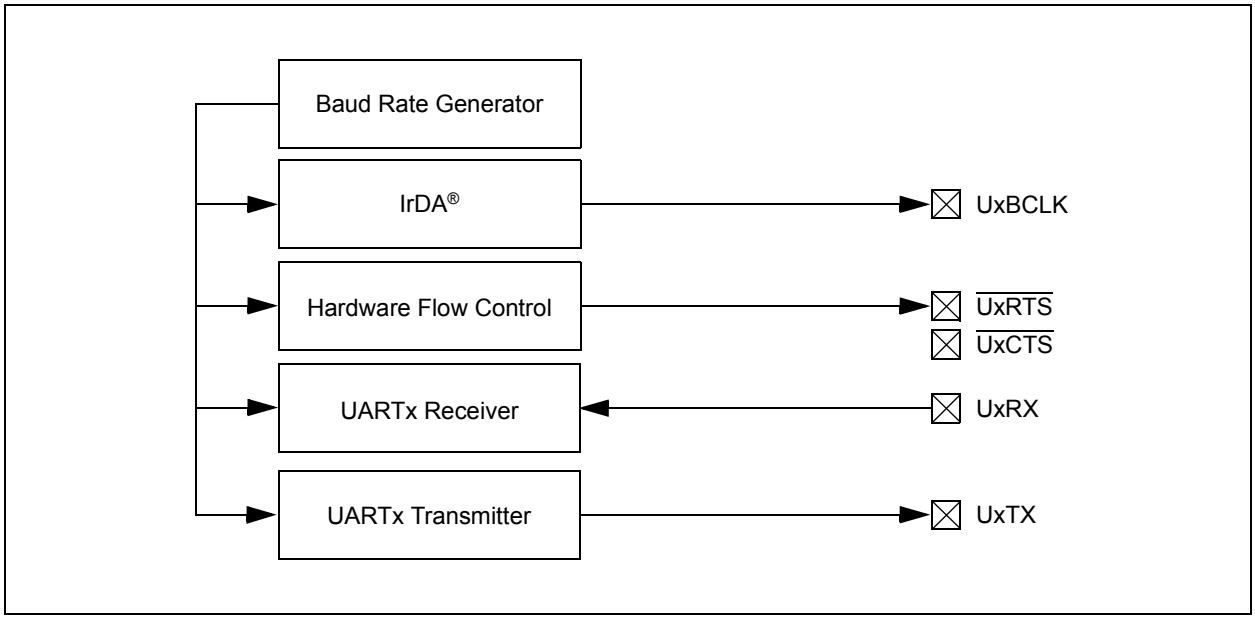
- Full-Duplex, 8-Bit or 9-Bit Data Transmission through the UxTX and UxRX Pins
- Even, Odd or No Parity Options (for 8-bit data)
- One or Two Stop bits
- Hardware Flow Control Option with $\overline{\text{UxCTS}}$ and $\overline{\text{UxRTS}}$ Pins
- Fully Integrated Baud Rate Generator (IBRG) with 16-Bit Prescaler
- Baud Rates Ranging from 1 Mbps to 15 bps at 16 MIPS
- 4-Deep, First-In-First-Out (FIFO) Transmit Data Buffer
- 4-Deep FIFO Receive Data Buffer
- Parity, Framing and Buffer Overrun Error Detection
- Support for 9-Bit mode with Address Detect (9^{th} bit = 1)
- Transmit and Receive Interrupts
- Loopback mode for Diagnostic Support
- Support for Sync and Break Characters
- Supports Automatic Baud Rate Detection
- IrDA® Encoder and Decoder Logic
- 16x Baud Clock Output for IrDA Support

A simplified block diagram of the UARTx module is shown in Figure 15-1. The UARTx module consists of these important hardware elements:

- Baud Rate Generator
- Asynchronous Transmitter
- Asynchronous Receiver

Note: Throughout this section, references to register and bit names that may be associated with a specific USART module are referred to generically by the use of 'x' in place of the specific module number. Thus, "UxSTA" might refer to the USART Status register for either USART1 or USART2.

FIGURE 15-1: UARTx MODULE SIMPLIFIED BLOCK DIAGRAM



NOTES:

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.

24.3 Pulse Generation and Delay

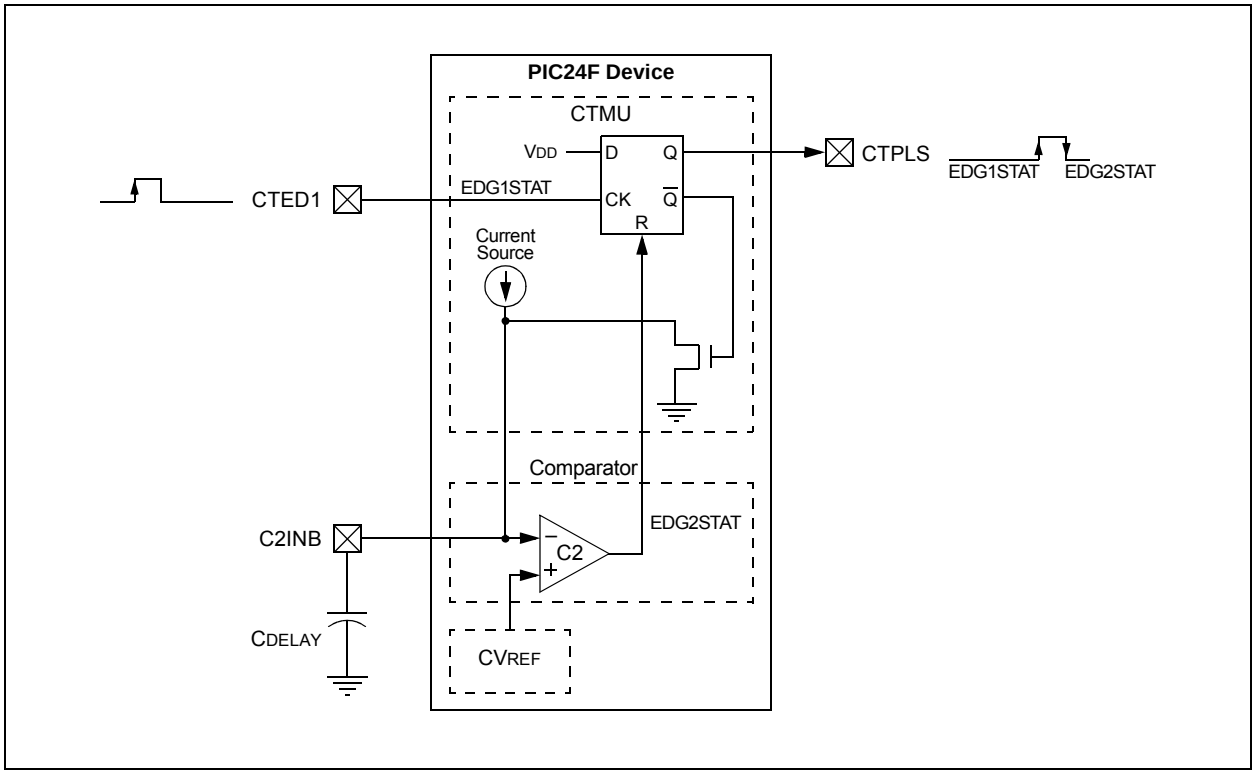
The CTMU module can also generate an output pulse with edges that are not synchronous with the device's system clock. More specifically, it can generate a pulse with a programmable delay from an edge event input to the module.

When the module is configured for pulse generation delay by setting the TGEN bit (CTMUCON1L<12>), the internal current source is connected to the B input of Comparator 2. A Capacitor (CDELAY) is connected to the Comparator 2 pin, C2INB, and the Comparator Voltage Reference, CVREF, is connected to C2INA. CVREF is then configured for a specific trip point. The module begins to charge CDELAY when an edge event is detected. While CVREF is greater than the voltage on CDELAY, the CTPLS pin is high.

When the voltage on CDELAY equals CVREF, CTPLS goes low. With Comparator 2 configured as the second edge, this stops the CTMU from charging. In this state event, the CTMU automatically connects to ground. The IDISSEN bit doesn't need to be set and cleared before the next CTPLS cycle.

Figure 24-3 illustrates the external connections for pulse generation, as well as the relationship of the different analog modules required. While CTED1 is shown as the input pulse source, other options are available. A detailed discussion on pulse generation with the CTMU module is provided in the "*PIC24F Family Reference Manual*".

FIGURE 24-3: TYPICAL CONNECTIONS AND INTERNAL CONFIGURATION FOR PULSE DELAY GENERATION



REGISTER 24-2: CTMUCON1H: CTMU CONTROL 1 HIGH REGISTER (CONTINUED)

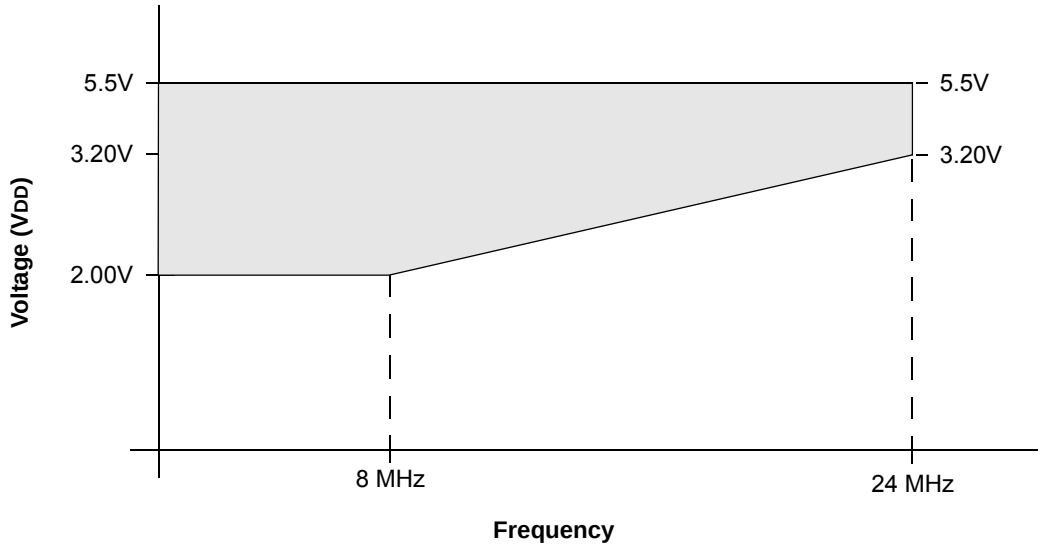
bit 6 **EDG2POL:** Edge 2 Polarity Select bit
1 = Edge 2 is programmed for a positive edge
0 = Edge 2 is programmed for a negative edge

bit 5-2 **EDG2SEL<3:0>:** Edge 2 Source Select bits
1111 = Edge 2 source is the Comparator 3 output
1110 = Edge 2 source is the Comparator 2 output
1101 = Edge 2 source is the Comparator 1 output
1100 = Unimplemented; do not use
1011 = Edge 2 source is CLC1
1010 = Edge 2 source is the MCCP2 Compare Event (CCP2IF)
1001 = Unimplemented; do not use
1000 = Edge 2 source is CTED13
0111 = Edge 2 source is CTED12
0110 = Edge 2 source is CTED11**(2)**
0101 = Edge 2 source is CTED10
0100 = Edge 2 source is CTED9**(2)**
0011 = Edge 2 source is CTED1
0010 = Edge 2 source is CTED2
0001 = Edge 2 source is the MCCP1 Compare Event (CCP1IF)
0000 = Edge 2 source is Timer1

bit 1-0 **Unimplemented:** Read as '0'

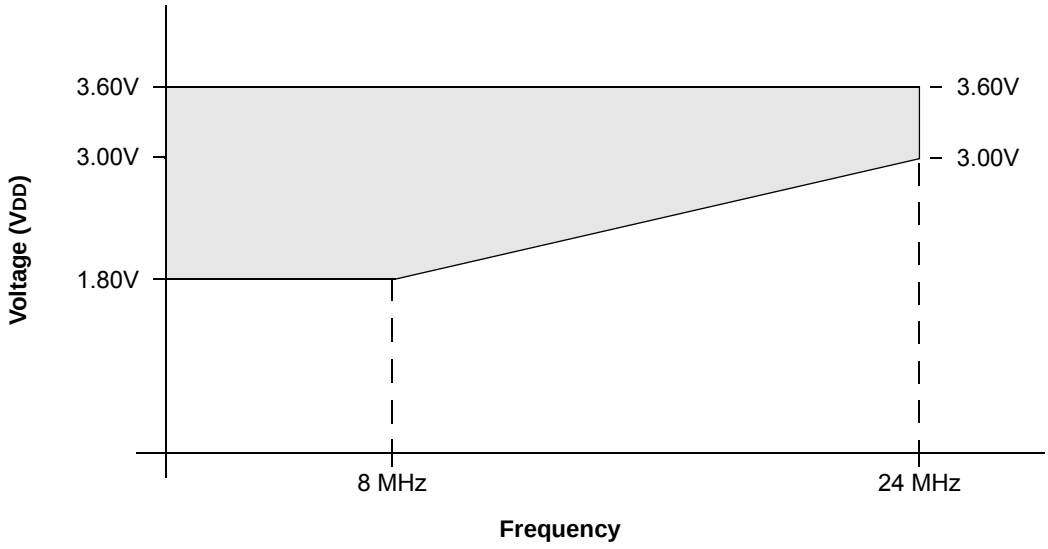
Note 1: Edge sources, CTED7 and CTED8, are not available on 28-pin and 20-pin devices.
2: Edge sources, CTED3, CTED9 and CTED11, are not available on 20-pin devices.

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.

TABLE 27-7: DC CHARACTERISTICS: IDLE CURRENT (IIDLE)

DC CHARACTERISTICS		Standard Operating Conditions: 1.8V to 3.6V (PIC24F16KMXXX) 2.0V to 5.5V (PIC24FV16KMXXX) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Parameter No.	**Device**	**Typical**	**Max**	**Units**	**Conditions**	
Idle Current (IIDLE)						
DC40	PIC24FV16KMXXX	120	200	µA	2.0V	0.5 MIPS, FOSC = 1 MHz(1)
		160	430	µA	5.0V	
	PIC24F16KMXXX	50	100	µA	1.8V	
		90	370	µA	3.3V	
DC42	PIC24FV16KMXXX	165	—	µA	2.0V	1 MIPS, FOSC = 2 MHz(1)
		260	—	µA	5.0V	
	PIC24F16KMXXX	95	—	µA	1.8V	
		180	—	µA	3.3V	
DC44	PIC24FV16KMXXX	3.1	6.5	mA	5.0V	16 MIPS, FOSC = 32 MHz(1)
	PIC24F16KMXXX	2.9	6.0	mA	3.3V	
DC46	PIC24FV16KMXXX	0.65	—	mA	2.0V	FRC (4 MIPS), FOSC = 8 MHz
		1.0	—	mA	5.0V	
	PIC24F16KMXXX	0.55	—	mA	1.8V	
		1.0	—	mA	3.3V	
DC50	PIC24FV16KMXXX	42	200	µA	2.0V	LPRC (15.5 KIPS), FOSC = 31 kHz
		65	225	µA	5.0V	
	PIC24F16KMXXX	2.2	18	µA	1.8V	
		4.0	40	µA	3.3V	

Legend: Unshaded rows represent PIC24F16KMXXX devices and shaded rows represent PIC24FV16KMXXX devices.

Note 1: The oscillator is in External Clock mode (FOSCSEL<2:0> = 010, FOSC<1:0> = 00).

T

Timer1 141
Timing Diagrams
A/D Conversion 295
Brown-out Reset Characteristics 284
Capture/Compare/PWM (MCCPx, SCCPx) 285
CLKO and I/O Timing 282
Example SPI Master Mode (CKE = 0) 286
Example SPI Master Mode (CKE = 1) 287
Example SPI Slave Mode (CKE = 0) 288
Example SPI Slave Mode (CKE = 1) 289
External Clock 280
I^2C Bus Data 290
I^2C Bus Start/Stop Bits 290
MSSPx I^2C Bus Data 293
MSSPx I^2C Bus Start/Stop Bits 292
Reset, Watchdog Timer. Oscillator Start-up Timer, Power-up Timer Characteristics 283
Timing Requirements
Capture/Compare/PWM (MCCPx, SCCPx) 285
Comparator 285
Comparator Voltage Reference Settling Time 285
I^2C Bus Data (Slave Mode) 291
I^2C Bus Data Requirements (Master Mode) 293
I^2C Bus Start/Stop Bits (Master Mode) 292
I^2C Bus Start/Stop Bits (Slave Mode) 290
SPI Mode (Master Mode, CKE = 0) 286
SPI Mode (Master Mode, CKE = 1) 287
SPI Mode (Slave Mode, CKE = 0) 288
SPI Slave Mode (CKE = 1) 289

U

UART
Baud Rate Generator (BRG) 174
Break and Sync Transmit Sequence 175
IrDA Support 175
Operation of $\overline{\text{UxCTS}}$ and $\overline{\text{UxRTS}}$ Control Pins 175
Receiving in 8-Bit or 9-Bit Data Mode 175
Transmitting in 8-Bit Data Mode 175
Transmitting in 9-Bit Data Mode 175
Universal Asynchronous Receiver Transmitter (UART) 173

V

Voltage Regulator (VREG) 134
Voltage-Frequency Graph (PIC24F16KM204 Extended) 267
Voltage-Frequency Graph (PIC24F16KM204 Industrial) 266
Voltage-Frequency Graph (PIC24FV16KM204 Extended) 267
Voltage-Frequency Graph (PIC24FV16KM204 Industrial) 266

W

Watchdog Timer (WDT) 257
Windowed Operation 258
WWW Address 332
WWW, On-Line Support 11