



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	I ² C, 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	2K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 19x10b/12b; D/A 2x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Through Hole
Package / Case	28-DIP (0.300", 7.62mm)
Supplier Device Package	28-SPDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic24f08km202-e-sp

PIC24FV16KM204 FAMILY

TABLE 1-4: DEVICE FEATURES FOR THE PIC24FV16KM104 FAMILY

Features	PIC24FV16KM104	PIC24FV16KM102	PIC24FV08KM102	PIC24FV08KM101
Operating Frequency	DC-32 MHz			
Program Memory (bytes)	16K	16K	8K	8K
Program Memory (instructions)	5632	5632	2816	2816
Data Memory (bytes)	1024			
Data EEPROM Memory (bytes)	512			
Interrupt Sources (soft vectors/NMI traps)	25 (21/4)			
Voltage Range	2.0-5.5V			
I/O Ports	PORTA<11:7,5:0> PORTB<15:0> PORTC<9:0>	PORTA<7,5:0> PORTB<15:0>		PORTA<5:0> PORTB<15:12,9:7, 4,2:0>
Total I/O Pins	37	23		17
Timers	5 (One 16-bit timer, two MCCPs/SCCPs with up to two 16/32 timers each)			
Capture/Compare/PWM modules	1 1			
MCCP				
SCCP				
Serial Communications	1 1			
MSSP				
UART				
Input Change Notification Interrupt	36	22		16
12-Bit Analog-to-Digital Module (input channels)	22	19		16
Analog Comparators	1			
8-Bit Digital-to-Analog Converters	—			
Operational Amplifiers	—			
Charge Time Measurement Unit (CTMU)	Yes			
Real-Time Clock and Calendar (RTCC)	—			
Configurable Logic Cell (CLC)	1			
Resets (and delays)	POR, BOR, RESET Instruction, MCLR, WDT, Illegal Opcode, REPEAT Instruction, Hardware Traps, Configuration Word Mismatch (PWRT, OST, PLL Lock)			
Instruction Set	76 Base Instructions, Multiple Addressing Mode Variations			
Packages	44-Pin QFN/TQFP, 48-Pin UQFN	28-Pin SPDIP/SSOP/SOIC/QFN		20-Pin SOIC/SSOP/PDIP

PIC24FV16KM204 FAMILY

4.2 Data Address Space

The PIC24F core has a separate, 16-bit-wide data memory space, addressable as a single linear range. The Data Space is accessed using two Address Generation Units (AGUs), one each for read and write operations. The Data Space memory map is displayed in Figure 4-3.

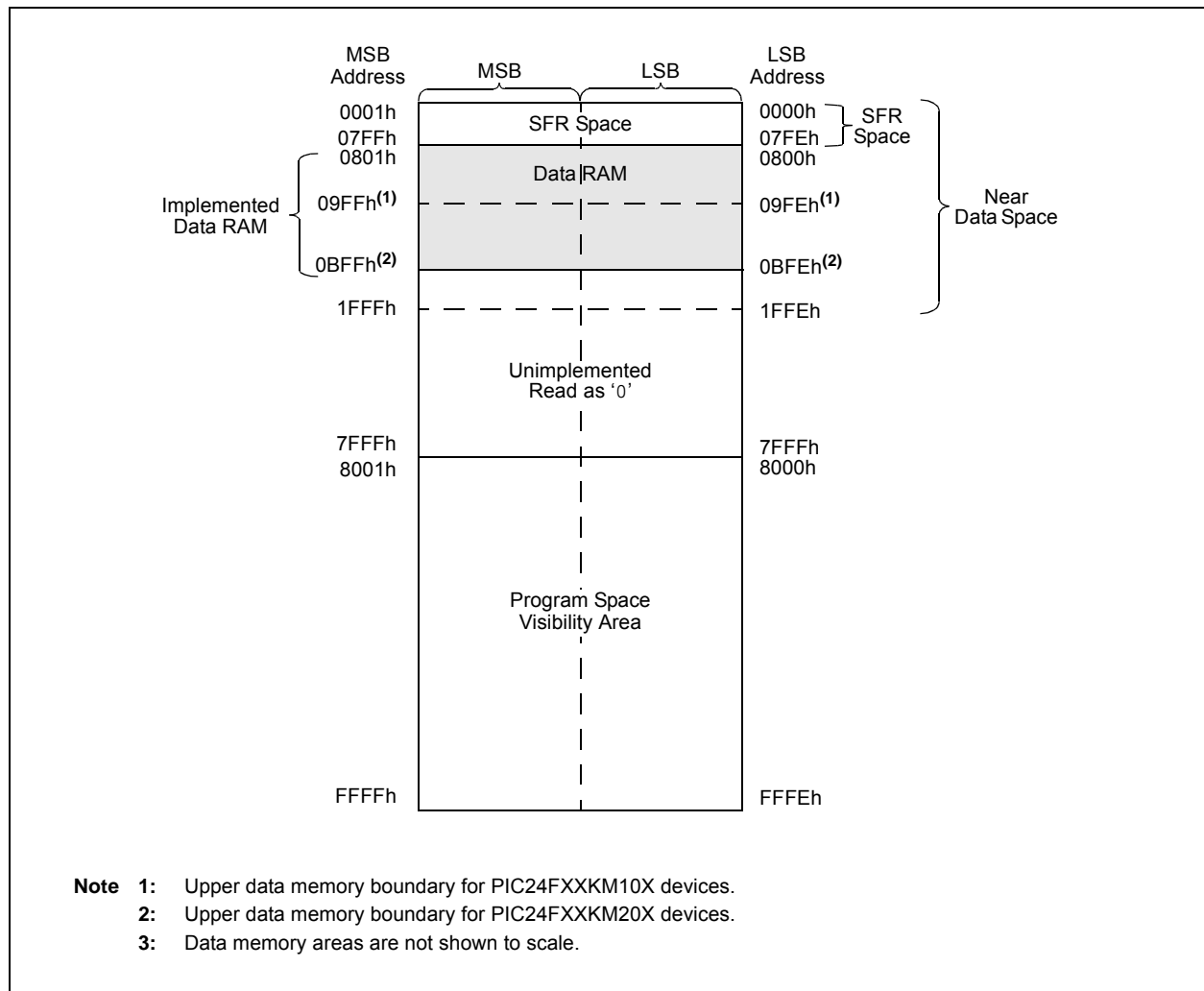
All Effective Addresses (EAs) in the data memory space are 16 bits wide and point to bytes within the Data Space. This gives a Data Space address range of 64 Kbytes or 32K words. The lower half of the data memory space (that is, when $EA_{<15>} = 0$) is used for implemented memory addresses, while the upper half ($EA_{<15>} = 1$) is reserved for the Program Space Visibility (PSV) area (see **Section 4.3.3 “Reading Data From Program Memory Using Program Space Visibility”**).

Depending on the particular device, PIC24FV16KM family devices implement either 512 or 1024 words of data memory. Should an EA point to a location outside of this area, an all zero word or byte will be returned.

4.2.1 DATA SPACE WIDTH

The data memory space is organized in byte-addressable, 16-bit-wide blocks. Data is aligned in data memory and registers as 16-bit words, but all the Data Space EAs resolve to bytes. The Least Significant Bytes (LSBs) of each word have even addresses, while the Most Significant Bytes (MSBs) have odd addresses.

FIGURE 4-3: DATA SPACE MEMORY MAP FOR PIC24FXXXXX FAMILY DEVICES⁽³⁾



PIC24FV16KM204 FAMILY

4.2.2 DATA MEMORY ORGANIZATION AND ALIGNMENT

To maintain backward compatibility with PIC® devices and improve Data Space memory usage efficiency, the PIC24F instruction set supports both word and byte operations. As a consequence of byte accessibility, all EA calculations are internally scaled to step through word-aligned memory. For example, the core recognizes that Post-Modified Register Indirect Addressing mode [Ws++] will result in a value of Ws + 1 for byte operations and Ws + 2 for word operations.

Data byte reads will read the complete word, which contains the byte, using the LSB of any EA to determine which byte to select. The selected byte is placed onto the LSB of the data path. That is, the data memory and the registers are organized as two parallel, byte-wide entities with shared (word) address decode, but separate write lines. Data byte writes only write to the corresponding side of the array or register, which matches the byte address.

All word accesses must be aligned to an even address. Misaligned word data fetches are not supported, so care must be taken when mixing byte and word operations, or translating from 8-bit MCU code. If a misaligned read or write is attempted, an address error trap will be generated. If the error occurred on a read, the instruction underway is completed; if it occurred on a write, the instruction will be executed, but the write will not occur. In either case, a trap is then executed, allowing the system and/or user to examine the machine state prior to execution of the address Fault.

All byte loads into any W register are loaded into the LSB; the MSB is not modified.

A Sign-Extend (SE) instruction is provided to allow the users to translate 8-bit signed data to 16-bit signed values. Alternatively, for 16-bit unsigned data, users can clear the MSB of any W register by executing a Zero-Extend (ZE) instruction on the appropriate address.

Although most instructions are capable of operating on word or byte data sizes, it should be noted that some instructions operate only on words.

4.2.3 NEAR DATA SPACE

The 8-Kbyte area between 0000h and 1FFFh is referred to as the Near Data Space. Locations in this space are directly addressable via a 13-bit absolute address field within all memory direct instructions. The remainder of the Data Space is addressable indirectly. Additionally, the whole Data Space is addressable using MOV instructions, which support Memory Direct Addressing (MDA) with a 16-bit address field. For PIC24FV16KM204 family devices, the entire implemented data memory lies in Near Data Space (NDS).

4.2.4 SFR SPACE

The first 2 Kbytes of the Near Data Space, from 0000h to 07FFh, are primarily occupied with Special Function Registers (SFRs). These are used by the PIC24F core and peripheral modules for controlling the operation of the device.

SFRs are distributed among the modules that they control and are generally grouped together by that module. Much of the SFR space contains unused addresses; these are read as '0'. The SFR space, where the SFRs are actually implemented, is provided in Table 4-2. Each implemented area indicates a 32-byte region where at least one address is implemented as an SFR. A complete listing of implemented SFRs, including their addresses, is provided in Table 4-3 through Table 4-26.

TABLE 4-2: IMPLEMENTED REGIONS OF SFR DATA SPACE

SFR Space Address								
	xx00	xx20	xx40	xx60	xx80	xxA0	xxC0	xxE0
000h	Core			ICN	Interrupts			—
100h	Timers	CLC	MCCP/SCCP					
200h	MSSP	UART	Op Amp	DAC	—	—	I/O	
300h	A/D/CMTU				—	—	—	—
400h	—	—	—	—	—	—	—	ANSEL
500h	—	—	—	—	—	—	—	—
600h	—	RTCC/Comp	—	Band Gap	—			
700h	—	—	System/ HLVD	NVM/PMD	—	—	—	—

Legend: — = No implemented SFRs in this block.

PIC24FV16KM204 FAMILY

4.3.3 READING DATA FROM PROGRAM MEMORY USING PROGRAM SPACE VISIBILITY

The upper 32 Kbytes of Data Space may optionally be mapped into a 16K word page of the program space. This provides transparent access of stored constant data from the Data Space without the need to use special instructions (i.e., `TBLRDH/H`).

Program space access through the Data Space occurs if the MSb of the Data Space, EA, is '1' and PSV is enabled by setting the PSV bit in the CPU Control (`CORCON<2>`) register. The location of the program memory space to be mapped into the Data Space is determined by the Program Space Visibility Page Address register (`PSVPAG`). This 8-bit register defines any one of 256 possible pages of 16K words in program space. In effect, `PSVPAG` functions as the upper 8 bits of the program memory address, with the 15 bits of the EA functioning as the lower bits.

By incrementing the PC by 2 for each program memory word, the lower 15 bits of Data Space addresses directly map to the lower 15 bits in the corresponding program space addresses.

Data reads from this area add an additional cycle to the instruction being executed, since two program memory fetches are required.

Although each Data Space address, 8000h and higher, maps directly into a corresponding program memory address (see Figure 4-7), only the lower 16 bits of the

24-bit program word are used to contain the data. The upper 8 bits of any program space locations used as data should be programmed with '1111 1111' or '0000 0000' to force a `NOP`. This prevents possible issues should the area of code ever be accidentally executed.

Note: PSV access is temporarily disabled during Table Reads/Writes.

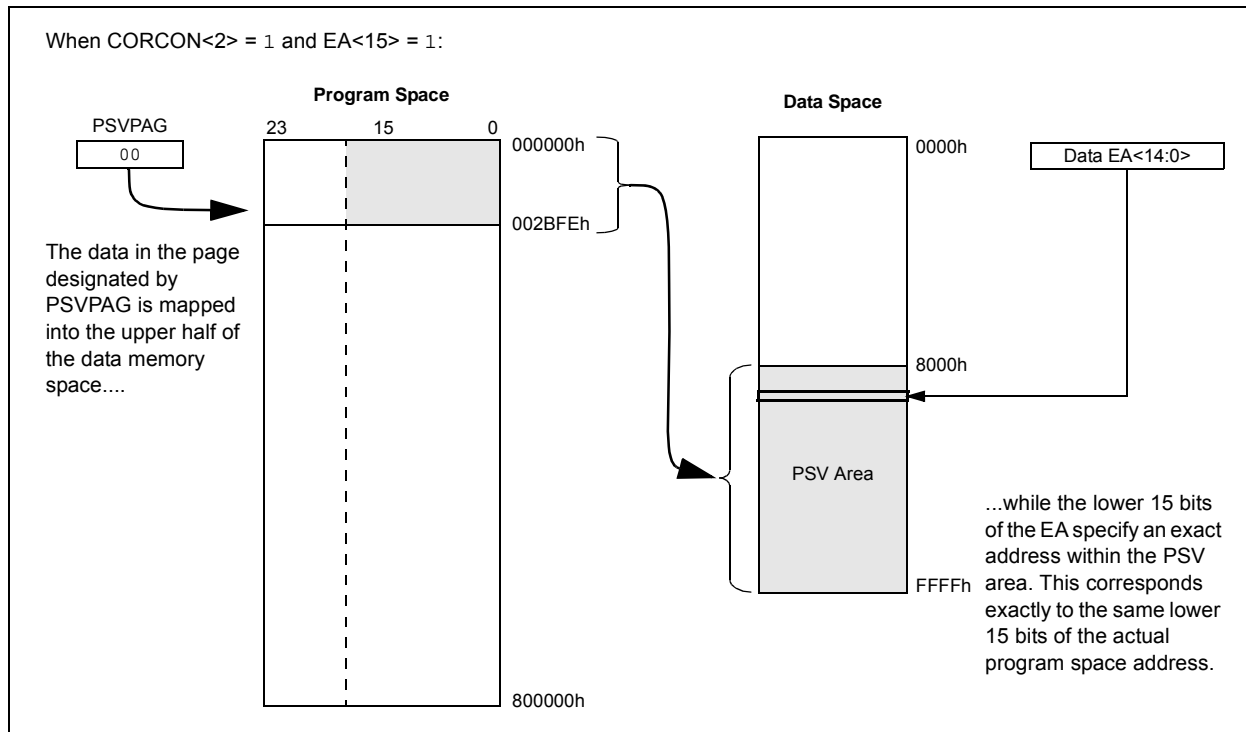
For operations that use PSV and are executed outside a `REPEAT` loop, the `MOV` and `MOV.D` instructions will require one instruction cycle in addition to the specified execution time. All other instructions will require two instruction cycles in addition to the specified execution time.

For operations that use PSV, which are executed inside a `REPEAT` loop, there will be some instances that require two instruction cycles in addition to the specified execution time of the instruction:

- Execution in the first iteration
- Execution in the last iteration
- Execution prior to exiting the loop due to an interrupt
- Execution upon re-entering the loop after an interrupt is serviced

Any other iteration of the `REPEAT` loop will allow the instruction accessing data, using PSV, to execute in a single cycle.

FIGURE 4-7: PROGRAM SPACE VISIBILITY OPERATION



PIC24FV16KM204 FAMILY

8.3 Interrupt Control and Status Registers

The PIC24FV16KM204 family of devices implements a total of 33 registers for the interrupt controller:

- INTCON1
- INTCON2
- IFS0 through IFS6
- IEC0 through IEC6
- IPC0 through IPC7, IPC10, IPC12, IPC15, IPC16, IPC18 through IPC20 and IPC24
- INTTREG

Global Interrupt Enable (GIE) control functions are controlled from INTCON1 and INTCON2. INTCON1 contains the Interrupt Nesting Disable (NSTDIS) bit, as well as the control and status flags for the processor trap sources. The INTCON2 register controls the external interrupt request signal behavior and the use of the AIVT.

The IFSx registers maintain all of the interrupt request flags. Each source of interrupt has a status bit, which is set by the respective peripherals, or external signal, and is cleared via software.

The IECx registers maintain all of the interrupt enable bits. These control bits are used to individually enable interrupts from the peripherals or external signals.

The IPCx registers are used to set the Interrupt Priority Level (IPL) for each source of interrupt. Each user interrupt source can be assigned to one of eight priority levels.

The INTTREG register contains the associated interrupt vector number and the new CPU Interrupt Priority Level, which are latched into the Vector Number (VECNUM<6:0>) and the Interrupt Level (ILR<3:0>) bit fields in the INTTREG register. The new Interrupt Priority Level is the priority of the pending interrupt.

The interrupt sources are assigned to the IFSx, IECx and IPCx registers in the same sequence. For example, the INT0 (External Interrupt 0) is depicted as having a vector number and a natural order priority of 0. The INT0IF status bit is found in IFS0<0>, the INT0IE enable bit in IEC0<0> and the INT0IP<2:0> priority bits are in the first position of IPC0 (IPC0<2:0>).

Although they are not specifically part of the interrupt control hardware, two of the CPU Control registers contain bits that control interrupt functionality. The ALU STATUS Register (SR) contains the IPL<2:0> bits (SR<7:5>). These indicate the current CPU Interrupt Priority Level. The user may change the current CPU Interrupt Priority Level by writing to the IPLx bits.

The CORCON register contains the IPL3 bit, which together with IPL<2:0>, also indicates the current CPU Interrupt Priority Level. IPL3 is a read-only bit so that the trap events cannot be masked by the user's software.

All Interrupt registers are described in Register 8-1 through Register 8-35, in the following sections.

13.0 CAPTURE/COMPARE/PWM/TIMER MODULES (MCCP AND SCCP)

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 MCCP/SCCP modules, refer to the “PIC24F Family Reference Manual”.

PIC24FV16KM204 family devices include several Capture/Compare/PWM/Timer base modules, which provide the functionality of three different peripherals of earlier PIC24F devices. The module can operate in one of three major modes:

- General Purpose Timer
- Input Capture
- Output Compare/PWM

The module is provided in two different forms, distinguished by the number of PWM outputs that the module can generate. Single output modules (SCCPs) provide only one PWM output. Multiple output modules (MCCPs) can provide up to six outputs and an extended range of power control features, depending on the pin count of the particular device. All other features of the modules are identical.

The SCCP and MCCP modules can be operated only in one of the three major modes at any time. The other modes are not available unless the module is reconfigured for the new mode.

A conceptual block diagram for the module is shown in Figure 13-1. All three modes share a time base generator and a common Timer register pair (CCPxTMRH/L); other shared hardware components are added as a particular mode requires.

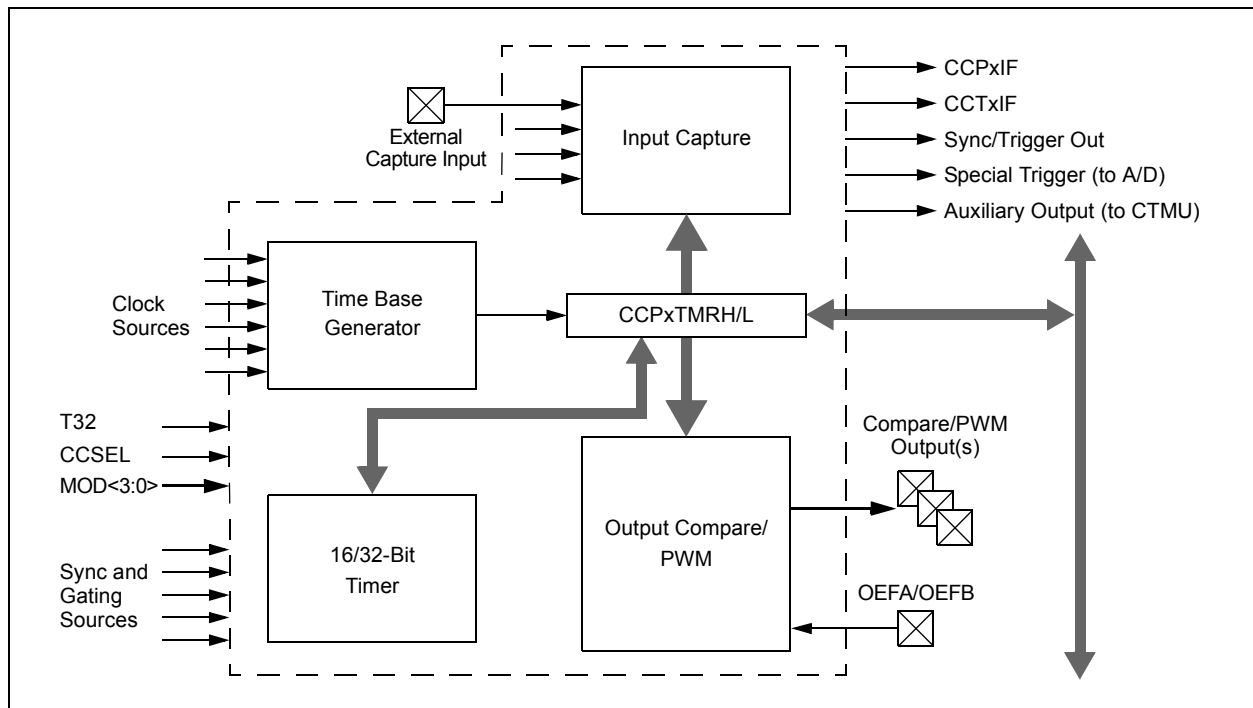
Each module has a total of seven control and status registers:

- CCPxCON1L (Register 13-1)
- CCPxCON1H (Register 13-2)
- CCPxCON2L (Register 13-3)
- CCPxCON2H (Register 13-4)
- CCPxCON3L (Register 13-5)
- CCPxCON3H (Register 13-6)
- CCPxSTATL (Register 13-7)

Each module also includes eight buffer/counter registers that serve as Timer Value registers or data holding buffers:

- CCPxTMRH/CCPxTMRL (Timer High/Low Counters)
- CCPxPRH/CCPxPRL (Timer Period High/Low)
- CCPxRA (Primary Output Compare Data Buffer)
- CCPxRB (Secondary Output Compare Data Buffer)
- CCPxBUFH/CCPxBUFL (Input Capture High/Low Buffers)

FIGURE 13-1: MCCPx/SCCPx CONCEPTUAL BLOCK DIAGRAM



PIC24FV16KM204 FAMILY

REGISTER 14-6: SSPxCON3: MSSPx CONTROL REGISTER 3 (SPI MODE)

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

R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ACKTIM	PCIE	SCIE	BOEN ⁽¹⁾	SDAHT	SBCDE	AHEN	DHEN
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 **ACKTIM:** Acknowledge Time Status bit (I²C™ mode only)
 Unused in SPI mode.
- bit 6 **PCIE:** Stop Condition Interrupt Enable bit (I²C mode only)
 Unused in SPI mode.
- bit 5 **SCIE:** Start Condition Interrupt Enable bit (I²C mode only)
 Unused in SPI mode.
- bit 4 **BOEN:** Buffer Overwrite Enable bit⁽¹⁾
In SPI Slave mode:
 1 = SSPxBUF updates every time that a new data byte is shifted in, ignoring the BF bit
 0 = If a new byte is received with the BF bit of the SSPxSTAT register already set, the SSPOV bit of the SSPxCON1 register is set and the buffer is not updated
- bit 3 **SDAHT:** SDAx Hold Time Selection bit (I²C mode only)
 Unused in SPI mode.
- bit 2 **SBCDE:** Slave Mode Bus Collision Detect Enable bit (I²C Slave mode only)
 Unused in SPI mode.
- bit 1 **AHEN:** Address Hold Enable bit (I²C Slave mode only)
 Unused in SPI mode.
- bit 0 **DHEN:** Data Hold Enable bit (Slave mode only)
 Unused in SPI mode.

Note 1: For Daisy-Chained SPI Operation: Allows the user to ignore all but the last received byte. SSPOV is still set when a new byte is received and BF = 1, but hardware continues to write the most recent byte to SSPxBUF.

PIC24FV16KM204 FAMILY

16.2 RTCC Module Registers

The RTCC module registers are organized into three categories:

- RTCC Control Registers
- RTCC Value Registers
- Alarm Value Registers

16.2.1 REGISTER MAPPING

To limit the register interface, the RTCC Timer and Alarm Time registers are accessed through corresponding register pointers. The RTCC Value register window (RTCVALH and RTCVALL) uses the RTCPTRx bits (RCFGCAL<9:8>) to select the desired Timer register pair (see Table 16-1).

By writing the RTCVALH byte, the RTCC Pointer value, the RTCPTR<1:0> bits decrement by one until they reach '00'. Once they reach '00', the MINUTES and SECONDS value will be accessible through RTCVALH and RTCVALL until the pointer value is manually changed.

TABLE 16-1: RTCVAL REGISTER MAPPING

RTCPTR<1:0>	RTCC Value Register Window	
	RTCVAL<15:8>	RTCVAL<7:0>
00	MINUTES	SECONDS
01	WEEKDAY	HOURS
10	MONTH	DAY
11	—	YEAR

The Alarm Value register window (ALRMVALH and ALRMVALL) uses the ALRMPTRx bits (ALCFGRPT<9:8>) to select the desired Alarm register pair (see Table 16-2).

By writing the ALRMVALH byte, the ALRMPTR<1:0> bits (Alarm Pointer value) decrement by one until they reach '00'. Once they reach '00', the ALRMMIN and ALRMSEC value will be accessible through ALRMVALH and ALRMVALL, until the pointer value is manually changed.

TABLE 16-2: ALRMVAL REGISTER MAPPING

ALRMPTR<1:0>	Alarm Value Register Window	
	ALRMVALH<15:8>	ALRMVALL<7:0>
00	ALRMMIN	ALRMSEC
01	ALRMWD	ALRMHR
10	ALRMMNTH	ALRMDAY
11	PWCSTAB	PWCSAMP

Considering that the 16-bit core does not distinguish between 8-bit and 16-bit read operations, the user must be aware that when reading either the ALRMVALH or ALRMVALL bytes, the ALRMPTR<1:0> value will be decremented. The same applies to the RTCVALH or RTCVALL bytes with the RTCPTR<1:0> being decremented.

Note: This only applies to read operations and not write operations.

16.2.2 WRITE LOCK

In order to perform a write to any of the RTCC Timer registers, the RTCWREN bit (RCFGCAL<13>) must be set (see Example 16-1 and Example 16-2).

Note: To avoid accidental writes to the timer, it is recommended that the RTCWREN bit (RCFGCAL<13>) is kept clear at any other time. For the RTCWREN bit to be set, there is only one instruction cycle time window allowed between the 55h/AA sequence and the setting of RTCWREN. Therefore, it is recommended that code follow the procedure in Example 16-2.

16.2.3 SELECTING RTCC CLOCK SOURCE

There are four reference source clock options that can be selected for the RTCC using the RTCCLK<1:0> bits (RTCPWC<11:10>): 00 = Secondary Oscillator, 01 = LPRC, 10 = 50 Hz External Clock and 11 = 60 Hz External Clock.

EXAMPLE 16-1: SETTING THE RTCWREN BIT IN ASSEMBLY

```
push    w7           ; Store W7 and W8 values on the stack.
push    w8
disi    #5           ; Disable interrupts until sequence is complete.
mov     #0x55, w7     ; Write 0x55 unlock value to NVMKEY.
mov     w7, NVMKEY
mov     #0xAA, w8     ; Write 0xAA unlock value to NVMKEY.
mov     w8, NVMKEY
bset    RCFGCAL, #13  ; Set the RTCWREN bit.
pop     w8           ; Restore the original W register values from the stack.
pop     w7
```

EXAMPLE 16-2: SETTING THE RTCWREN BIT IN 'C'

```
//This builtin function executes implements the unlock sequence and sets
//the RTCWREN bit.
__builtin_write_RTCWREN();
```

PIC24FV16KM204 FAMILY

NOTES:

20.0 8-BIT DIGITAL-TO-ANALOG CONVERTER (DAC)

PIC24FV16KM204 family devices include two 8-bit Digital-to-Analog Converters (DACs) for generating analog outputs from digital data. A simplified block diagram for a single DAC is shown in Figure 20-1. Both of the DACs are identical.

$$V_{DAC} = \frac{V_{DACREF} \times DAC_{xDAT}}{256}$$

Each DAC includes these features:

- Precision 8-bit resistor ladder for high accuracy
- Fast settling time, supporting 1 Msps effective sampling rates
- Buffered output voltage
- Three user-selectable voltage reference options
- Multiple conversion Trigger options, plus a manual convert-on-write option
- Left and right justified input data options
- User-selectable Sleep and Idle mode operation

When using the DAC, it is recommended to set the ANSx and TRISx bits for the DACx output pin to configure it as an analog output. See **Section 11.2 “Configuring Analog Port Pins”** for more information.

The diagram illustrates the internal architecture of the DACx peripheral. It features a multiplexer at the top left that selects between DACxREF, AVDD, and BGBUF1 to produce the DACREF<1:0> signal. This signal, along with DACEN (the AND of DACSIDL and DACSLP), is fed into an 8-Bit Resistor Ladder. The DACxCON and DACxDAT registers provide an 8-bit digital input to the ladder. The ladder's output is connected to the DACxOUT Pin through a switch controlled by DACOE. A Trigger and Interrupt Logic block receives a 32-bit DACx Trigger Sources input and outputs DACTRIG to the DACxCON register and DACxIF to the CPU. The DACxCON register also provides the DACTSEL<4:0> signal to a multiplexer that selects the trigger source for the DACx Trigger Sources input. The entire peripheral is powered by AVss.

PIC24FV16KM204 FAMILY

REGISTER 20-1: DACxCON: DACx CONTROL REGISTER

R/W-0	U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
DACEN	—	DACSIDL	DACSLP	DACFM	—	SRDIS	DACTRIG
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
DACOE	DACTSEL4	DACTSEL3	DACTSEL2	DACTSEL1	DACTSEL0	DACREF1	DACREF0
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 **DACEN:** DACx Enable bit
1 = Module is enabled
0 = Module is disabled
- bit 14 **Unimplemented:** Read as '0'
- bit 13 **DACSIDL:** DACx Stop in Idle Mode bit
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode
- bit 12 **DACSLP:** DACx Enable Peripheral During Sleep bit
1 = DACx continues to output the most recent value of DACxDAT during Sleep mode
0 = DACx is powered down in Sleep mode; DACxOUT pin is controlled by the TRISx and LATx bits
- bit 11 **DACFM:** DACx Data Format Select bit
1 = Data is left justified (data stored in DACxDAT<15:8>)
0 = Data is right justified (data stored in DACxDAT<7:0>)
- bit 10 **Unimplemented:** Read as '0'
- bit 9 **SRDIS:** Soft Reset Disable bit
1 = DACxCON and DACxDAT SFRs reset only on a POR or BOR Reset
0 = DACxCON and DACxDAT SFRs reset on any type of device Reset
- bit 8 **DACTRIG:** DACx Trigger Input Enable bit
1 = Analog output value updates when the selected (by DACTSEL<4:0>) event occurs
0 = Analog output value updates as soon as DACxDAT is written (DAC Trigger is ignored)
- bit 7 **DACOE:** DACx Output Enable bit
1 = DACx output pin is enabled and driven on the DACxOUT pin
0 = DACx output pin is disabled, DACx output is available internally to other peripherals only

Note 1: BGBUF1 voltage is configured by BUFREF<1:0> (BUFCON0<1:0>).

PIC24FV16KM204 FAMILY

REGISTER 25-4: FOSC: OSCILLATOR CONFIGURATION REGISTER

R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
FCKSM1	FCKSM0	SOSCSEL	POSCFREQ1	POSCFREQ0	OSCIOFNC	POSCMD1	POSCMD0
bit 7							bit 0

Legend:

R = Readable bit

P = Programmable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

- bit 7-6 **FCKSM<1:0>**: Clock Switching and Fail-Safe Clock Monitor Selection Configuration bits
1x = Clock switching is disabled, Fail-Safe Clock Monitor is disabled
01 = Clock switching is enabled, Fail-Safe Clock Monitor is disabled
00 = Clock switching is enabled, Fail-Safe Clock Monitor is enabled
- bit 5 **SOSCSEL**: Secondary Oscillator Power Selection Configuration bit
1 = Secondary Oscillator is configured for high-power operation
0 = Secondary Oscillator is configured for low-power operation
- bit 4-3 **POSCFREQ<1:0>**: Primary Oscillator Frequency Range Configuration bits
11 = Primary Oscillator/External Clock input frequency is greater than 8 MHz
10 = Primary Oscillator/External Clock input frequency is between 100 kHz and 8 MHz
01 = Primary Oscillator/External Clock input frequency is less than 100 kHz
00 = Reserved; do not use
- bit 2 **OSCIOFNC**: CLKO Enable Configuration bit
1 = CLKO output signal is active on the OSCO pin; Primary Oscillator must be disabled or configured for the External Clock (EC) mode for the CLKO to be active (POSCMD<1:0> = 11 or 00)
0 = CLKO output is disabled
- bit 1-0 **POSCMD<1:0>**: Primary Oscillator Configuration bits
11 = Primary Oscillator mode is disabled
10 = HS Oscillator mode is selected
01 = XT Oscillator mode is selected
00 = External Clock mode is selected

26.0 DEVELOPMENT SUPPORT

The PIC® microcontrollers (MCU) and dsPIC® digital signal controllers (DSC) are supported with a full range of software and hardware development tools:

- Integrated Development Environment
 - MPLAB® X IDE Software
- Compilers/Assemblers/Linkers
 - MPLAB XC Compiler
 - MPASM™ Assembler
 - MPLINK™ Object Linker/
MPLIB™ Object Librarian
 - MPLAB Assembler/Linker/Librarian for
Various Device Families
- Simulators
 - MPLAB X SIM Software Simulator
- Emulators
 - MPLAB REAL ICE™ In-Circuit Emulator
- In-Circuit Debuggers/Programmers
 - MPLAB ICD 3
 - PICKit™ 3
- Device Programmers
 - MPLAB PM3 Device Programmer
- Low-Cost Demonstration/Development Boards,
Evaluation Kits and Starter Kits
- Third-party development tools

26.1 MPLAB X Integrated Development Environment Software

The MPLAB X IDE is a single, unified graphical user interface for Microchip and third-party software, and hardware development tool that runs on Windows®, Linux and Mac OS® X. Based on the NetBeans IDE, MPLAB X IDE is an entirely new IDE with a host of free software components and plug-ins for high-performance application development and debugging. Moving between tools and upgrading from software simulators to hardware debugging and programming tools is simple with the seamless user interface.

With complete project management, visual call graphs, a configurable watch window and a feature-rich editor that includes code completion and context menus, MPLAB X IDE is flexible and friendly enough for new users. With the ability to support multiple tools on multiple projects with simultaneous debugging, MPLAB X IDE is also suitable for the needs of experienced users.

Feature-Rich Editor:

- Color syntax highlighting
- Smart code completion makes suggestions and provides hints as you type
- Automatic code formatting based on user-defined rules
- Live parsing

User-Friendly, Customizable Interface:

- Fully customizable interface: toolbars, toolbar buttons, windows, window placement, etc.
- Call graph window

Project-Based Workspaces:

- Multiple projects
- Multiple tools
- Multiple configurations
- Simultaneous debugging sessions

File History and Bug Tracking:

- Local file history feature
- Built-in support for Bugzilla issue tracker

PIC24FV16KM204 FAMILY

26.2 MPLAB XC Compilers

The MPLAB XC Compilers are complete ANSI C compilers for all of Microchip's 8, 16 and 32-bit MCU and DSC devices. These compilers provide powerful integration capabilities, superior code optimization and ease of use. MPLAB XC Compilers run on Windows, Linux or MAC OS X.

For easy source level debugging, the compilers provide debug information that is optimized to the MPLAB X IDE.

The free MPLAB XC Compiler editions support all devices and commands, with no time or memory restrictions, and offer sufficient code optimization for most applications.

MPLAB XC Compilers include an assembler, linker and utilities. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. MPLAB XC Compiler uses the assembler to produce its object file. Notable features of the assembler include:

- Support for the entire device instruction set
- Support for fixed-point and floating-point data
- Command-line interface
- Rich directive set
- Flexible macro language
- MPLAB X IDE compatibility

26.3 MPASM Assembler

The MPASM Assembler is a full-featured, universal macro assembler for PIC10/12/16/18 MCUs.

The MPASM Assembler generates relocatable object files for the MPLINK Object Linker, Intel® standard HEX files, MAP files to detail memory usage and symbol reference, absolute LST files that contain source lines and generated machine code, and COFF files for debugging.

The MPASM Assembler features include:

- Integration into MPLAB X IDE projects
- User-defined macros to streamline assembly code
- Conditional assembly for multipurpose source files
- Directives that allow complete control over the assembly process

26.4 MPLINK Object Linker/ MPLIB Object Librarian

The MPLINK Object Linker combines relocatable objects created by the MPASM Assembler. It can link relocatable objects from precompiled libraries, using directives from a linker script.

The MPLIB Object Librarian manages the creation and modification of library files of precompiled code. When a routine from a library is called from a source file, only the modules that contain that routine will be linked in with the application. This allows large libraries to be used efficiently in many different applications.

The object linker/library features include:

- Efficient linking of single libraries instead of many smaller files
- Enhanced code maintainability by grouping related modules together
- Flexible creation of libraries with easy module listing, replacement, deletion and extraction

26.5 MPLAB Assembler, Linker and Librarian for Various Device Families

MPLAB Assembler produces relocatable machine code from symbolic assembly language for PIC24, PIC32 and dsPIC DSC devices. MPLAB XC Compiler uses the assembler to produce its object file. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. Notable features of the assembler include:

- Support for the entire device instruction set
- Support for fixed-point and floating-point data
- Command-line interface
- Rich directive set
- Flexible macro language
- MPLAB X IDE compatibility

PIC24FV16KM204 FAMILY

TABLE 27-8: DC CHARACTERISTICS: POWER-DOWN CURRENT (IPD)

DC CHARACTERISTICS		Standard Operating Conditions: 1.8V to 3.6V (PIC24F16KM204) 2.0V to 5.5V (PIC24FV16KM204) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended					
Parameter No.	Device	Typical ⁽¹⁾	Max	Units	Conditions		
Power-Down Current (IPD)							
DC60	PIC24FV16KMXXX	6.0	—	μA	-40°C	2.0V	Sleep Mode ⁽²⁾
			8.0		+25°C		
			8.5		+60°C		
			9.0		+85°C		
			15.0		+125°C		
		6.0	—	μA	-40°C	5.0V	
			8.0		+25°C		
			9.0		+60°C		
			10.0		+85°C		
			15.0		+125°C		
	PIC24F16KMXXX	0.025	—	μA	-40°C	1.8V	
			0.80		+25°C		
			1.5		+60°C		
			2.0		+85°C		
			7.5		+125°C		
		0.040	—	μA	-40°C	3.3V	
			1.0		+25°C		
			2.0		+60°C		
			3.0		+85°C		
			7.5		+125°C		
DC61	PIC24FV16KMXXX	0.25	—	μA	+85°C	2.0V	Low-Voltage Sleep Mode ⁽²⁾
			7.5		+125°C		
		0.35	3.0	μA	+85°C	5.0V	
			7.5		+125°C		

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

Note 1: Data in the Typical column is at 3.3V, +25°C (PIC24F16KMXXX) or 5.0V, +25°C (PIC24FV16KMXXX) unless otherwise stated. Parameters are for design guidance only and are not tested.

2: Base IPD is measured with all peripherals and clocks shut down. All I/Os are configured as outputs and set low. PMSLP is set to '0' and WDT, etc., are all switched off.

3: The Δ current is the additional current consumed when the module is enabled. This current should be added to the base IPD current.

PIC24FV16KM204 FAMILY

TABLE 27-9: DC CHARACTERISTICS: I/O PIN INPUT SPECIFICATIONS

DC CHARACTERISTICS			Standard Operating Conditions: 1.8V to 3.6V (PIC24F16KM204) 2.0V to 5.5V (PIC24FV16KM204) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	Sym	Characteristic	Min	Typ ⁽¹⁾	Max	Units	Conditions
DI10 DI15 DI16 DI17 DI18 DI19	V _{IL}	Input Low Voltage⁽⁴⁾					
		I/O Pins	V _{SS}	—	0.2 V _{DD}	V	
		MCLR	V _{SS}	—	0.2 V _{DD}	V	
		OSCI (XT mode)	V _{SS}	—	0.2 V _{DD}	V	
		OSCI (HS mode)	V _{SS}	—	0.2 V _{DD}	V	
		I/O Pins with I ² C™ Buffer	V _{SS}	—	0.3 V _{DD}	V	SMBus disabled
		I/O Pins with SMBus Buffer	V _{SS}	—	0.8	V	SMBus enabled
DI20 DI25 DI26 DI27 DI28 DI29	V _{IH}	Input High Voltage^(4,5)					
		I/O Pins:					
		with Analog Functions	0.8 V _{DD}	—	V _{DD}	V	
		Digital Only	0.8 V _{DD}	—	V _{DD}	V	
		MCLR	0.8 V _{DD}	—	V _{DD}	V	
		OSCI (XT mode)	0.7 V _{DD}	—	V _{DD}	V	
		OSCI (HS mode)	0.7 V _{DD}	—	V _{DD}	V	
		I/O Pins with I ² C Buffer:					
		with Analog Functions	0.7 V _{DD}	—	V _{DD}	V	
		Digital Only	0.7 V _{DD}	—	V _{DD}	V	
		I/O Pins with SMBus	2.1	—	V _{DD}	V	2.5V ≤ V _{PIN} ≤ V _{DD}
DI30	ICNPU	CNx Pull-up Current	50	250	500	μA	V _{DD} = 3.3V, V _{PIN} = V _{SS}
DI31	IPU	Maximum Load Current for Digital High Detection w/Internal Pull-up	—	—	30	μA	V _{DD} = 2.0V
			—	—	1000	μA	V _{DD} = 3.3V
DI50 DI51	I _{IL}	Input Leakage Current^(2,3)					
		I/O Ports	—	0.050	±0.100	μA	V _{SS} ≤ V _{PIN} ≤ V _{DD} , Pin at high-impedance
		Pins with OAxOUT Functions (RB15 and RB3)	—	0.100	±0.200	μA	V _{SS} ≤ V _{PIN} ≤ V _{DD} , Pin at high-impedance

Note 1: Data in “Typ” column is at 3.3V, +25°C unless otherwise stated.

2: The leakage current on the MCLR pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.

3: Negative current is defined as current sourced by the pin.

4: Refer to Table 1-4 and Table 1-5 for I/O pin buffer types.

5: V_{IH} requirements are met when the internal pull-ups are enabled.

PIC24FV16KM204 FAMILY

FIGURE 27-11: EXAMPLE SPI MASTER MODE TIMING (CKE = 0)

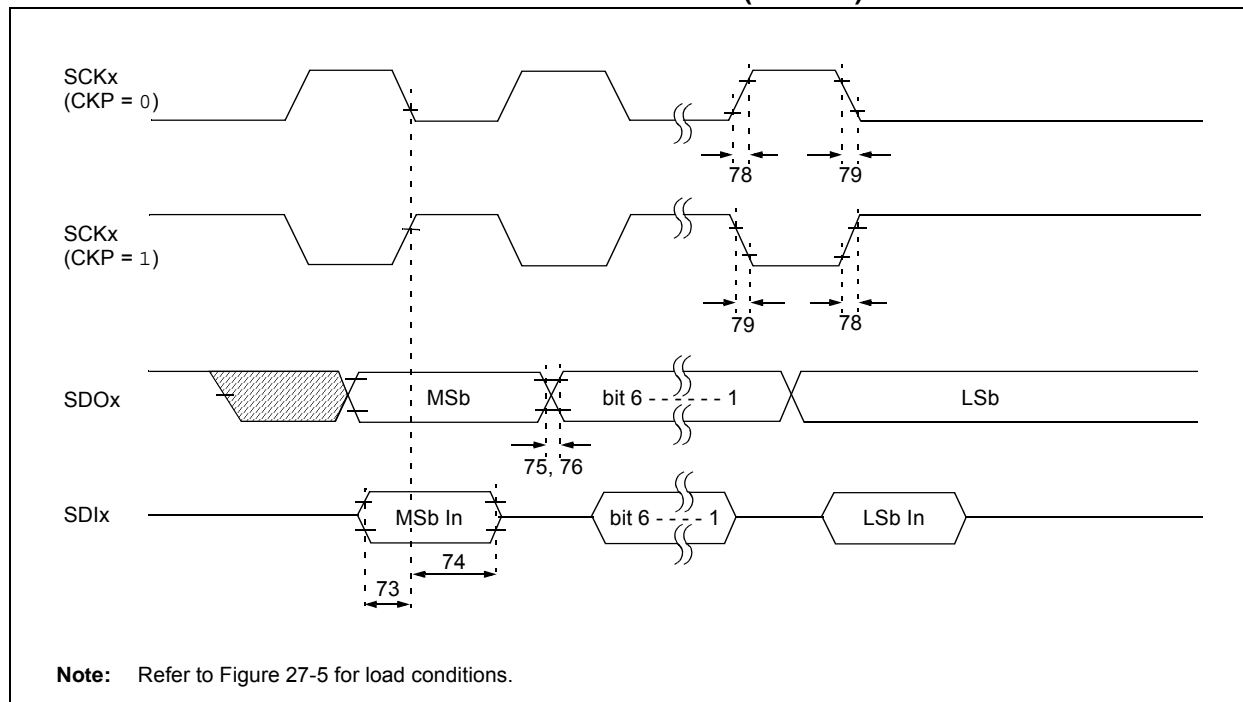
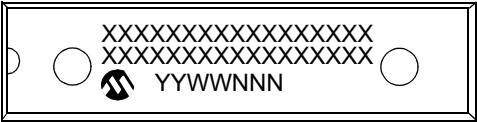


TABLE 27-29: EXAMPLE SPI MODE REQUIREMENTS (MASTER MODE, CKE = 0)

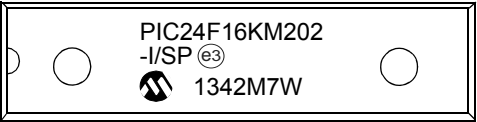
Param No.	Symbol	Characteristic	Min	Max	Units	Conditions
73	TdIV2sCH, TdIV2sCL	Setup Time of SDIx Data Input to SCKx Edge	20	—	ns	
74	TsCH2DiL, TsCL2DiL	Hold Time of SDIx Data Input to SCKx Edge	40	—	ns	
75	TdoR	SDOx Data Output Rise Time	—	25	ns	
76	TdoF	SDOx Data Output Fall Time	—	25	ns	
78	TscR	SCKx Output Rise Time (Master mode)	—	25	ns	
79	TscF	SCKx Output Fall Time (Master mode)	—	25	ns	
	Fsck	SCKx Frequency	—	10	MHz	

PIC24FV16KM204 FAMILY

28-Lead SPDIP (.300")



Example



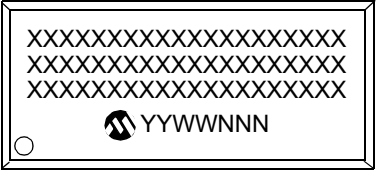
28-Lead SSOP (5.30 mm)



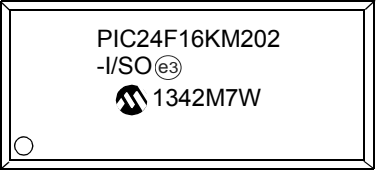
Example



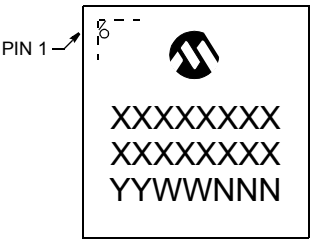
28-Lead SOIC (7.50 mm)



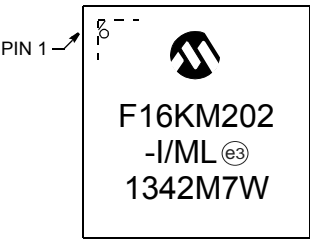
Example



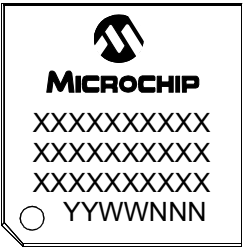
28-Lead QFN (6x6 mm)



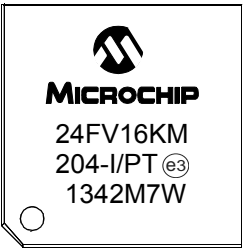
Example



44-Lead TQFP (10x10x1 mm)



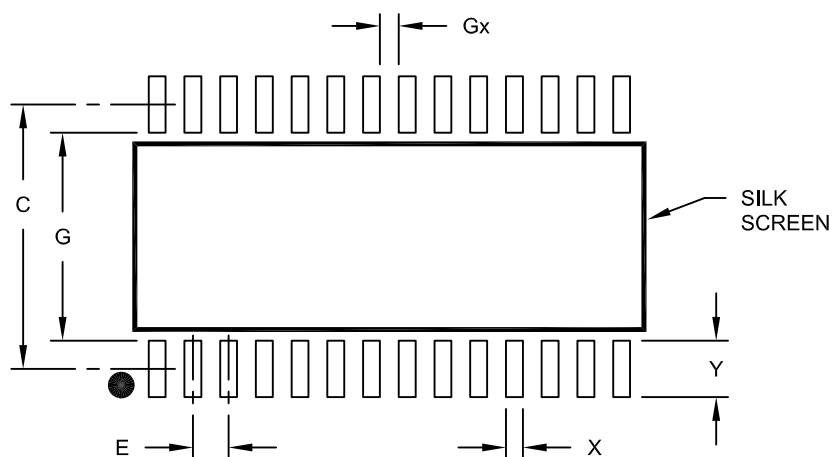
Example



PIC24FV16KM204 FAMILY

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>



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E	1.27 BSC		
Contact Pad Spacing	C		9.40	
Contact Pad Width (X28)	X			0.60
Contact Pad Length (X28)	Y			2.00
Distance Between Pads	Gx	0.67		
Distance Between Pads	G	7.40		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

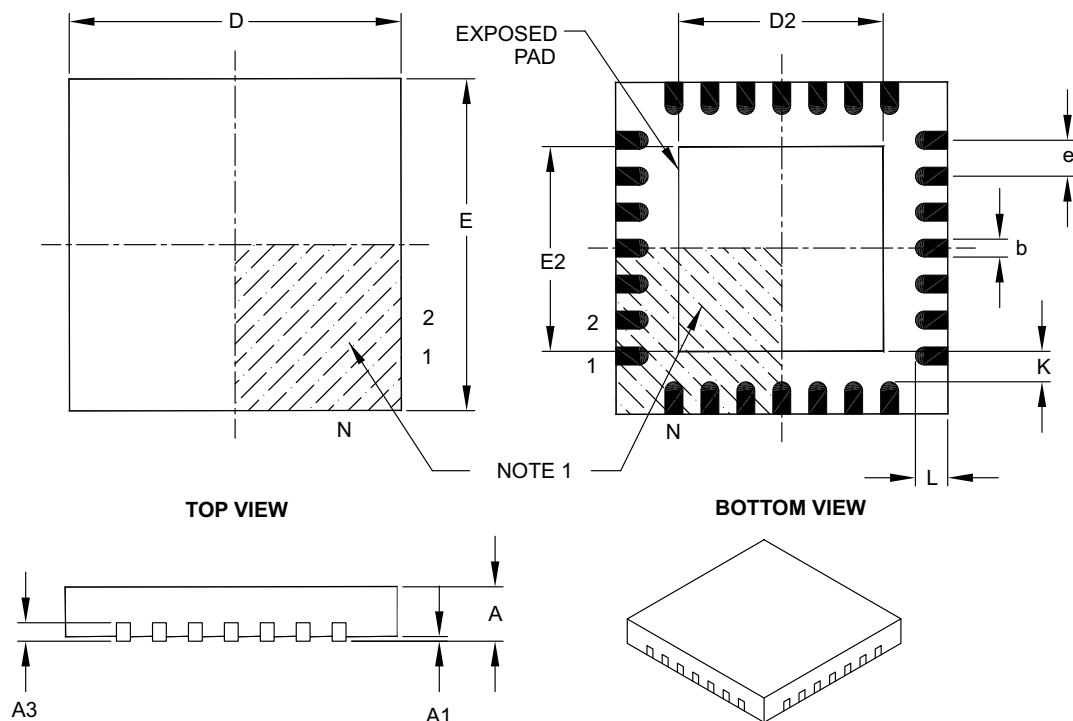
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2052A

PIC24FV16KM204 FAMILY

28-Lead Plastic Quad Flat, No Lead Package (ML) – 6x6 mm Body [QFN] with 0.55 mm Contact Length

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		28		
Pitch	e		0.65 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		6.00 BSC		
Exposed Pad Width	E2		3.65	3.70	4.20
Overall Length	D		6.00 BSC		
Exposed Pad Length	D2		3.65	3.70	4.20
Contact Width	b		0.23	0.30	0.35
Contact Length	L		0.50	0.55	0.70
Contact-to-Exposed Pad	K		0.20	–	–

Notes:

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- Package is saw singulated.
- 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 C04-105B