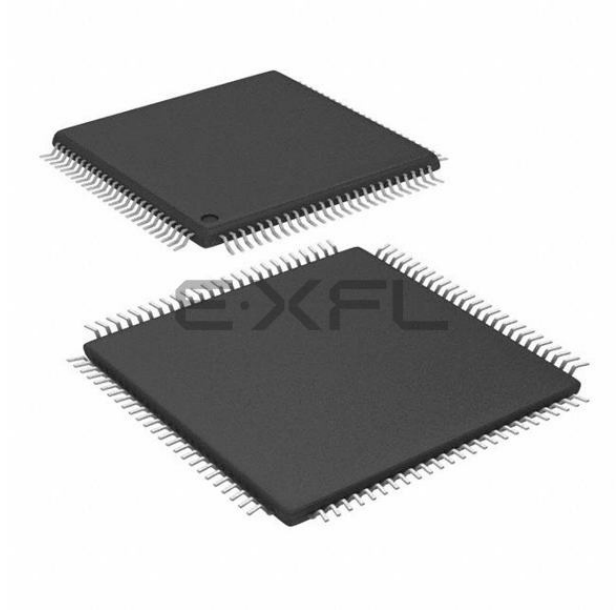
Microchip Technology - DSPIC33FJ256GP710A-I/PT Datasheet



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	dsPIC
Core Size	16-Bit
Speed	40 MIPs
Connectivity	CANbus, I²C, IrDA, LINbus, SPI, UART/USART
Peripherals	AC'97, Brown-out Detect/Reset, DMA, I²S, POR, PWM, WDT
Number of I/O	85
Program Memory Size	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	30K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 32x10b/12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	100-TQFP
Supplier Device Package	100-TQFP (12x12)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/dspic33fj256gp710a-i-pt

REGISTER 3-1: SR: CPU STATUS REGISTER

bit 8 **DC:** MCU ALU Half Carry/$\overline{\text{Borrow}}$ bit
1 = A carry-out from the 4th low-order bit (for byte sized data) or 8th low-order bit (for word sized data) of the result occurred
0 = No carry-out from the 4th low-order bit (for byte sized data) or 8th low-order bit (for word sized data) of the result occurred

bit 7-5 **IPL<2:0>:** CPU Interrupt Priority Level Status bits(2)
111 = CPU Interrupt Priority Level is 7 (15), user interrupts disabled
110 = CPU Interrupt Priority Level is 6 (14)
101 = CPU Interrupt Priority Level is 5 (13)
100 = CPU Interrupt Priority Level is 4 (12)
011 = CPU Interrupt Priority Level is 3 (11)
010 = CPU Interrupt Priority Level is 2 (10)
001 = CPU Interrupt Priority Level is 1 (9)
000 = CPU Interrupt Priority Level is 0 (8)

bit 4 **RA:** REPEAT Loop Active bit
1 = REPEAT loop in progress
0 = REPEAT loop not in progress

bit 3 **N:** MCU ALU Negative bit
1 = Result was negative
0 = Result was non-negative (zero or positive)

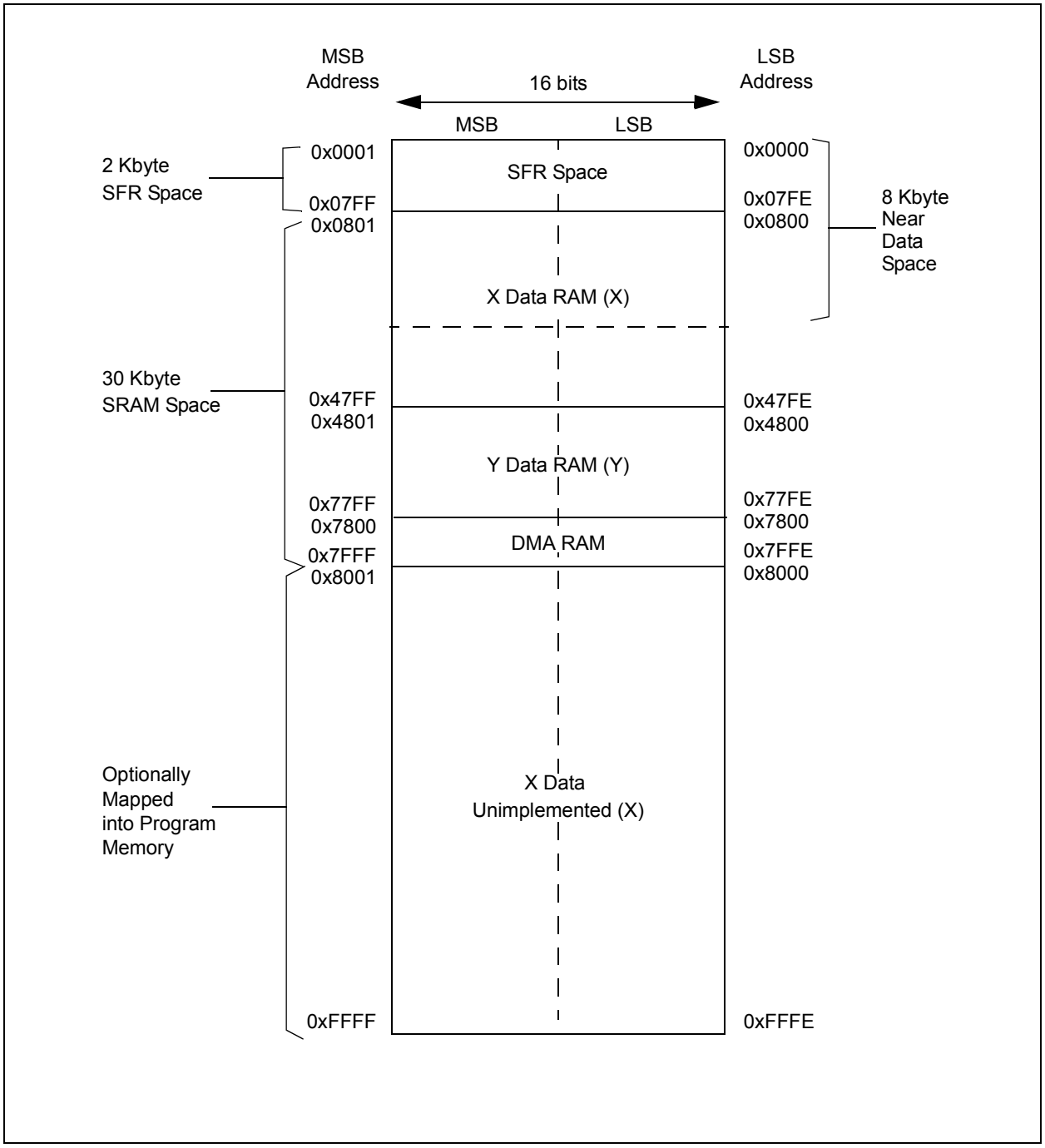
bit 2 **OV:** MCU ALU Overflow bit
This bit is used for signed arithmetic (2's complement). It indicates an overflow of the magnitude which causes the sign bit to change state.
1 = Overflow occurred for signed arithmetic (in this arithmetic operation)
0 = No overflow occurred

bit 1 **Z:** MCU ALU Zero bit
1 = An operation which affects the Z bit has set it at some time in the past
0 = The most recent operation which affects the Z bit has cleared it (i.e., a non-zero result)

bit 0 **C:** MCU ALU Carry/$\overline{\text{Borrow}}$ bit
1 = A carry-out from the Most Significant bit of the result occurred
0 = No carry-out from the Most Significant bit of the result occurred

Note 1: This bit may be read or cleared (not set).
2: The IPL<2:0> bits are concatenated with the IPL<3> bit (CORCON<3>) to form the CPU Interrupt Priority Level. The value in parentheses indicates the IPL if IPL<3> = 1. User interrupts are disabled when IPL<3> = 1.
3: The IPL<2:0> Status bits are read only when NSTDIS = 1 (INTCON1<15>).

FIGURE 4-5: DATA MEMORY MAP FOR dsPIC33FJXXXGPX06A/X08A/X10A DEVICES WITH 30 KBS RAM



4.6.3 READING DATA FROM PROGRAM MEMORY USING PROGRAM SPACE VISIBILITY

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

Program space access through the data space occurs if the Most Significant bit of the data space EA is '1' and program space visibility is enabled by setting the PSV bit in the Core Control register (CORCON<2>). The location of the program memory space to be mapped into the data space is determined by the Program Space Visibility Page 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. Note that 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 to 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 4-11), 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 location 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.

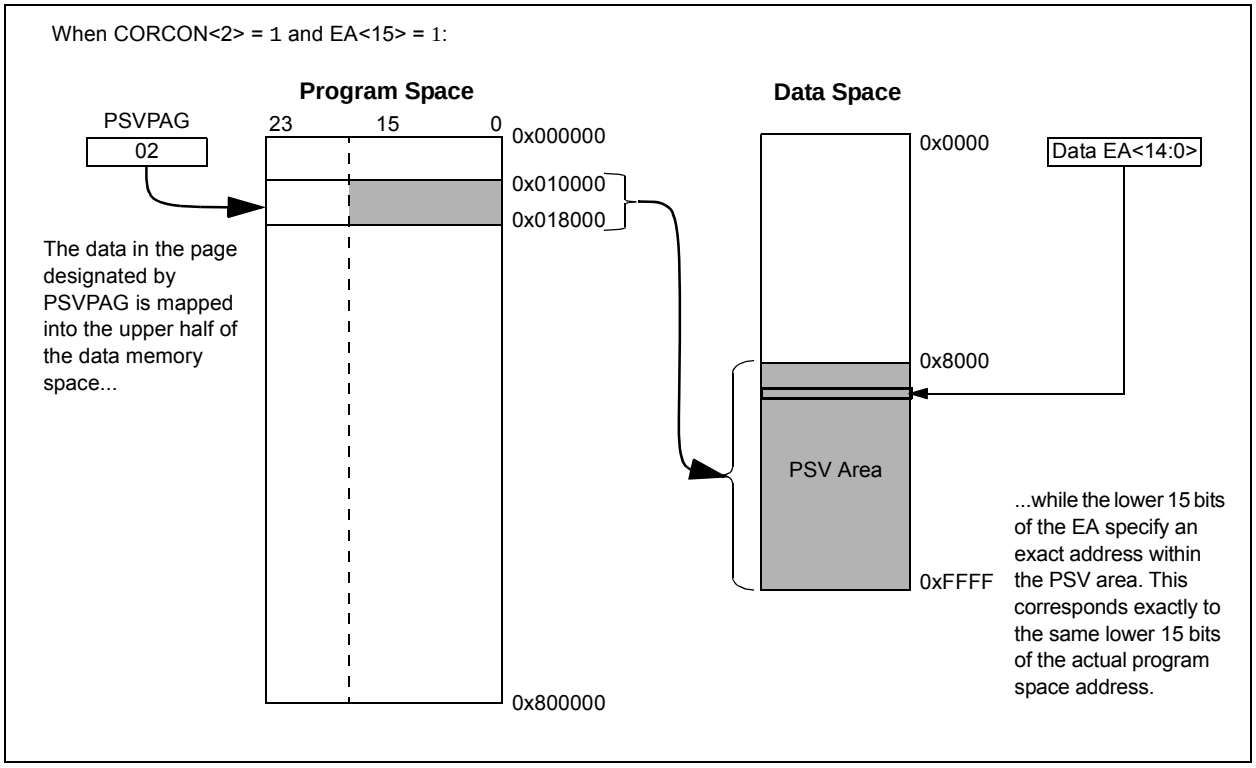
For operations that use PSV and are executed outside a REPEAT loop, the MOV and MOV.D instructions require one instruction cycle in addition to the specified execution time. All other instructions require two instruction cycles in addition to the specified execution time.

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 4-11: PROGRAM SPACE VISIBILITY OPERATION



EXAMPLE 5-2: LOADING THE WRITE BUFFERS

```
; Set up NVMCON for row programming operations
       MOV     #0x4001, W0                     ;
       MOV     W0, NVMCON                      ; Initialize NVMCON
; Set up a pointer to the first program memory location to be written
; program memory selected, and writes enabled
       MOV     #0x0000, W0                     ;
       MOV     W0, TBLPAG                      ; Initialize PM Page Boundary SFR
       MOV     #0x6000, W0                     ; An example program memory address
; Perform the TBLWT instructions to write the latches
; 0th_program_word
       MOV     #LOW_WORD_0, W2                 ;
       MOV     #HIGH_BYTE_0, W3                ;
       TBLWTL  W2, [W0]                        ; Write PM low word into program latch
       TBLWTH  W3, [W0++]                      ; Write PM high byte into program latch
; 1st_program_word
       MOV     #LOW_WORD_1, W2                 ;
       MOV     #HIGH_BYTE_1, W3                ;
       TBLWTL  W2, [W0]                        ; Write PM low word into program latch
       TBLWTH  W3, [W0++]                      ; Write PM high byte into program latch
;  2nd_program_word
       MOV     #LOW_WORD_2, W2                 ;
       MOV     #HIGH_BYTE_2, W3                ;
       TBLWTL  W2, [W0]                        ; Write PM low word into program latch
       TBLWTH  W3, [W0++]                      ; Write PM high byte into program latch
       •
       •
       •
; 63rd_program_word
       MOV     #LOW_WORD_31, W2                ;
       MOV     #HIGH_BYTE_31, W3               ;
       TBLWTL  W2, [W0]                        ; Write PM low word into program latch
       TBLWTH  W3, [W0++]                      ; Write PM high byte into program latch
```

EXAMPLE 5-3: INITIATING A PROGRAMMING SEQUENCE

```
       DISI    #5                      ; Block all interrupts with priority <7
                                       ; for next 5 instructions
       MOV     #0x55, W0
       MOV     W0, NVMKEY              ; Write the 55 key
       MOV     #0xAA, W1               ;
       MOV     W1, NVMKEY              ; Write the AA key
       BSET    NVMCON, #WR             ; Start the erase sequence
       NOP                             ; Insert two NOPs after the
       NOP                             ; erase command is asserted
```

REGISTER 7-1: SR: CPU STATUS REGISTER(1)

R-0	R-0	R/C-0	R/C-0	R-0	R/C-0	R -0	R/W-0
OA	OB	SA	SB	OAB	SAB	DA	DC
bit 15							bit 8

R/W-0(3)	R/W-0(3)	R/W-0(3)	R-0	R/W-0	R/W-0	R/W-0	R/W-0
IPL2(2)	**IPL1(2)**	**IPL0(2)**	RA	N	OV	Z	C
bit 7							bit 0

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

bit 7-5 **IPL<2:0>:** CPU Interrupt Priority Level Status bits(2)
111 = CPU Interrupt Priority Level is 7 (15), user interrupts disabled
110 = CPU Interrupt Priority Level is 6 (14)
101 = CPU Interrupt Priority Level is 5 (13)
100 = CPU Interrupt Priority Level is 4 (12)
011 = CPU Interrupt Priority Level is 3 (11)
010 = CPU Interrupt Priority Level is 2 (10)
001 = CPU Interrupt Priority Level is 1 (9)
000 = CPU Interrupt Priority Level is 0 (8)

Note 1: For complete register details, see Register 3-1.
2: The IPL<2:0> bits are concatenated with the IPL<3> bit (CORCON<3>) to form the CPU Interrupt Priority Level. The value in parentheses indicates the IPL if IPL<3> = 1. User interrupts are disabled when IPL<3> = 1.
3: The IPL<2:0> Status bits are read-only when NSTDIS (INTCON1<15>) = 1.

REGISTER 7-2: CORCON: CORE CONTROL REGISTER(1)

U-0	U-0	U-0	R/W-0	R/W-0	R-0	R-0	R-0
—	—	—	US	EDT	DL<2:0>		
bit 15							bit 8

R/W-0	R/W-0	R/W-1	R/W-0	R/C-0	R/W-0	R/W-0	R/W-0
SATA	SATB	SATDW	ACCSAT	**IPL3(2)**	PSV	RND	IF
bit 7							bit 0

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

bit 3 **IPL3:** CPU Interrupt Priority Level Status bit 3(2)
1 = CPU interrupt priority level is greater than 7
0 = CPU interrupt priority level is 7 or less

Note 1: For complete register details, see Register 3-2.
2: The IPL3 bit is concatenated with the IPL<2:0> bits (SR<7:5>) to form the CPU Interrupt Priority Level.

REGISTER 7-4: INTCON2: INTERRUPT CONTROL REGISTER 2

R/W-0	R-0	U-0	U-0	U-0	U-0	U-0	U-0
ALTIVT	DISI	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	INT4EP	INT3EP	INT2EP	INT1EP	INT0EP
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 **ALTIVT:** Enable Alternate Interrupt Vector Table bit
1 = Use alternate vector table
0 = Use standard (default) vector table

bit 14 **DISI:** DISI Instruction Status bit
1 = DISI instruction is active
0 = DISI instruction is not active

bit 13-5 **Unimplemented:** Read as '0'

bit 4 **INT4EP:** External Interrupt 4 Edge Detect Polarity Select bit
1 = Interrupt on negative edge
0 = Interrupt on positive edge

bit 3 **INT3EP:** External Interrupt 3 Edge Detect Polarity Select bit
1 = Interrupt on negative edge
0 = Interrupt on positive edge

bit 2 **INT2EP:** External Interrupt 2 Edge Detect Polarity Select bit
1 = Interrupt on negative edge
0 = Interrupt on positive edge

bit 1 **INT1EP:** External Interrupt 1 Edge Detect Polarity Select bit
1 = Interrupt on negative edge
0 = Interrupt on positive edge

bit 0 **INT0EP:** External Interrupt 0 Edge Detect Polarity Select bit
1 = Interrupt on negative edge
0 = Interrupt on positive edge

REGISTER 7-27: IPC12: INTERRUPT PRIORITY CONTROL REGISTER 12

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	T8IP<2:0>			—	MI2C2IP<2:0>		
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	SI2C2IP<2:0>			—	T7IP<2:0>		
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 **Unimplemented:** Read as '0'

bit 14-12 **T8IP<2:0>:** Timer8 Interrupt Priority bits
111 = Interrupt is priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is priority 1
000 = Interrupt source is disabled

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

bit 10-8 **MI2C2IP<2:0>:** I2C2 Master Events Interrupt Priority bits
111 = Interrupt is priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is priority 1
000 = Interrupt source is disabled

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

bit 6-4 **SI2C2IP<2:0>:** I2C2 Slave Events Interrupt Priority bits
111 = Interrupt is priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is priority 1
000 = Interrupt source is disabled

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

bit 2-0 **T7IP<2:0>:** Timer7 Interrupt Priority bits
111 = Interrupt is priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is priority 1
000 = Interrupt source is disabled

REGISTER 7-33: INTTREG: INTERRUPT CONTROL AND STATUS REGISTER

U-0	U-0	U-0	U-0	R-0	R-0	R-0	R-0
—	—	—	—	ILR<3:0>			
bit 15							bit 8

U-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
—	VECNUM<6:0>						
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 15-12 **Unimplemented:** Read as '0'

bit 11-8 **ILR<3:0>:** New CPU Interrupt Priority Level bits
1111 = CPU Interrupt Priority Level is 15
•
•
•
0001 = CPU Interrupt Priority Level is 1
0000 = CPU Interrupt Priority Level is 0

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

bit 6-0 **VECNUM<6:0>:** Vector Number of Pending Interrupt bits
0111111 = Interrupt Vector pending is number 135
•
•
•
0000001 = Interrupt Vector pending is number 9
0000000 = Interrupt Vector pending is number 8

16.0 SERIAL PERIPHERAL INTERFACE (SPI)

Note 1: This data sheet summarizes the features of the dsPIC33FJXXXGPX06A/X08A/X10A family of devices. However, it is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 18. "Serial Peripheral Interface (SPI)"** (DS70206) in the *"dsPIC33F/PIC24H Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).

2: Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

The Serial Peripheral Interface (SPI) module is a synchronous serial interface useful for communicating with other peripheral or microcontroller devices. These peripheral devices may be serial EEPROMs, shift registers, display drivers, Analog-to-Digital Converters (ADC), etc. The SPI module is compatible with SPI and SIOP from Motorola®.

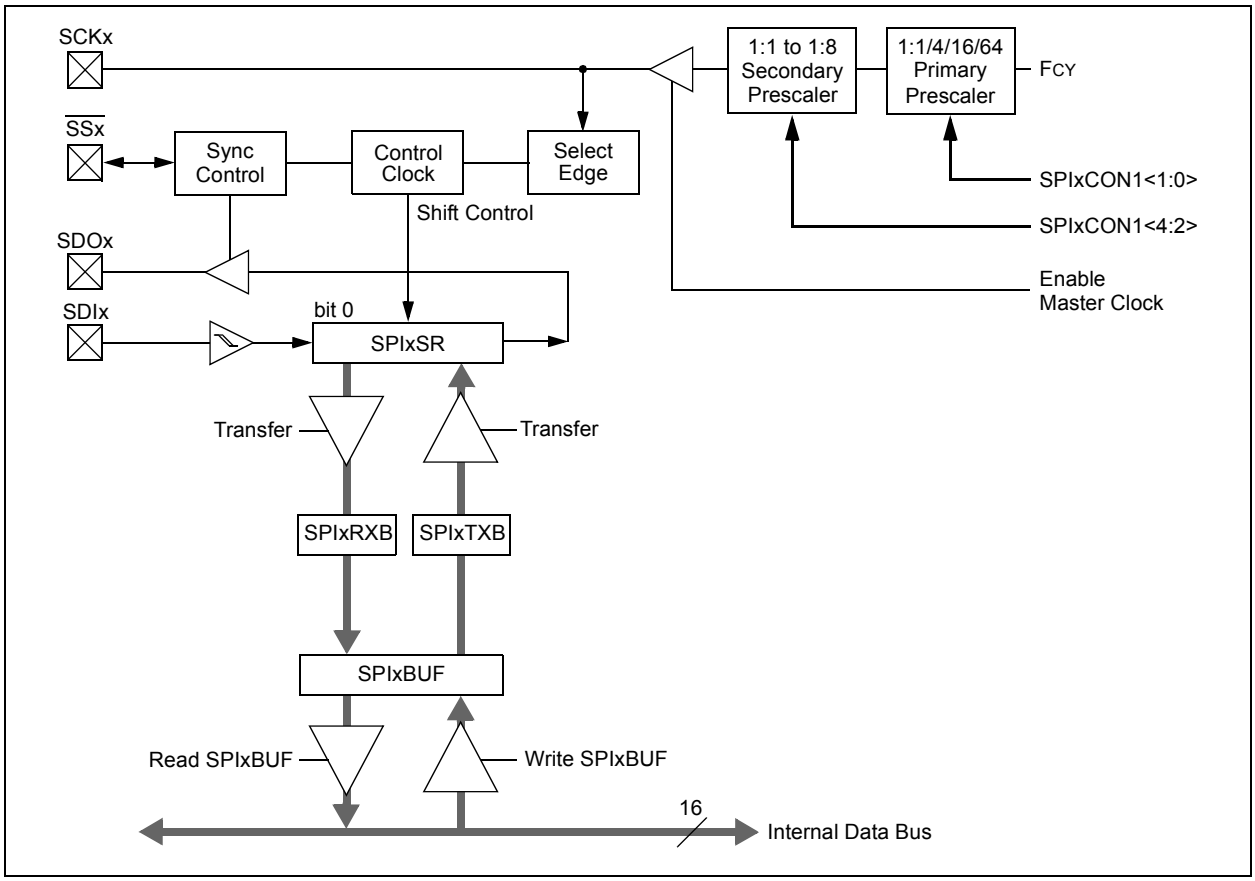
Note: In this section, the SPI modules are referred to together as SPIx, or separately as SPI1 and SPI2. Special Function Registers will follow a similar notation. For example, SPIxCON refers to the control register for the SPI1 or SPI2 module.

Each SPI module consists of a 16-bit shift register, SPIxSR (where x = 1 or 2), used for shifting data in and out, and a buffer register, SPIxBUF. A control register, SPIxCON, configures the module. Additionally, a status register, SPIxSTAT, indicates various status conditions.

The serial interface consists of 4 pins: SDIx (serial data input), SDOx (serial data output), SCKx (shift clock input or output), and $\overline{SSx}$ (active-low slave select).

In Master mode operation, SCK is a clock output but in Slave mode, it is a clock input.

FIGURE 16-1: SPI MODULE BLOCK DIAGRAM



REGISTER 18-2: UxSTA: UARTx STATUS AND CONTROL REGISTER (CONTINUED)

bit 5 **ADDEN:** Address Character Detect bit (bit 8 of received data = 1)
- 1 = Address Detect mode enabled. If 9-bit mode is not selected, this does not take effect
- 0 = Address Detect mode 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 (read/clear 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 UxRSR to the empty state

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

Note 1: Refer to **Section 17. "UART"** (DS70188) in the *"dsPIC33F/PIC24H Family Reference Manual"* for information on enabling the UART module for transmit operation.

REGISTER 19-12: CiBUFPNT1: ECAN™ FILTER 0-3 BUFFER POINTER REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F3BP<3:0>				F2BP<3:0>			
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
F1BP<3:0>				F0BP<3:0>			
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 15-12 **F3BP<3:0>:** RX Buffer Written when Filter 3 Hits bits
1111 = Filter hits received in RX FIFO buffer
1110 = Filter hits received in RX Buffer 14
•
•
•
0001 = Filter hits received in RX Buffer 1
0000 = Filter hits received in RX Buffer 0

bit 11-8 **F2BP<3:0>:** RX Buffer Written when Filter 2 Hits bits
1111 = Filter hits received in RX FIFO buffer
1110 = Filter hits received in RX Buffer 14
•
•
•
0001 = Filter hits received in RX Buffer 1
0000 = Filter hits received in RX Buffer 0

bit 7-4 **F1BP<3:0>:** RX Buffer Written when Filter 1 Hits bits
1111 = Filter hits received in RX FIFO buffer
1110 = Filter hits received in RX Buffer 14
•
•
•
0001 = Filter hits received in RX Buffer 1
0000 = Filter hits received in RX Buffer 0

bit 3-0 **F0BP<3:0>:** RX Buffer Written when Filter 0 Hits bits
1111 = Filter hits received in RX FIFO buffer
1110 = Filter hits received in RX Buffer 14
•
•
•
0001 = Filter hits received in RX Buffer 1
0000 = Filter hits received in RX Buffer 0

22.4 Watchdog Timer (WDT)

For dsPIC33FJXXXGPX06A/X08A/X10A devices, the WDT is driven by the LPRC oscillator. When the WDT is enabled, the clock source is also enabled.

The nominal WDT clock source from LPRC is 32 kHz. This feeds a prescaler and then can be configured for either 5-bit (divide-by-32) or 7-bit (divide-by-128) operation. The prescaler is set by the WDTPRE Configuration bit. With a 32 kHz input, the prescaler yields a nominal WDT time-out period (TWDT) of 1 ms in 5-bit mode, or 4 ms in 7-bit mode.

A variable postscaler divides down the WDT prescaler output and allows for a wide range of time-out periods. The postscaler is controlled by the WDTPOST<3:0> Configuration bits (FWDT<3:0>) which allow the selection of a total of 16 settings, from 1:1 to 1:32,768. Using the prescaler and postscaler, time-out periods ranging from 1 ms to 131 seconds can be achieved.

The WDT, prescaler and postscaler are reset:

- On any device Reset
- On the completion of a clock switch, whether invoked by software (i.e., setting the OSWEN bit after changing the NOSC bits) or by hardware (i.e., Fail-Safe Clock Monitor)
- When a PWRSAV instruction is executed (i.e., Sleep or Idle mode is entered)
- When the device exits Sleep or Idle mode to resume normal operation
- By a CLRWDT instruction during normal execution

If the WDT is enabled, it will continue to run during Sleep or Idle modes. When the WDT time-out occurs, the device will wake the device and code execution will continue from where the PWRSAV instruction was executed. The corresponding SLEEP or IDLE bits (RCON<3,2>) will need to be cleared in software after the device wakes up.

The WDT flag bit, WDTO (RCON<4>), is not automatically cleared following a WDT time-out. To detect subsequent WDT events, the flag must be cleared in software.

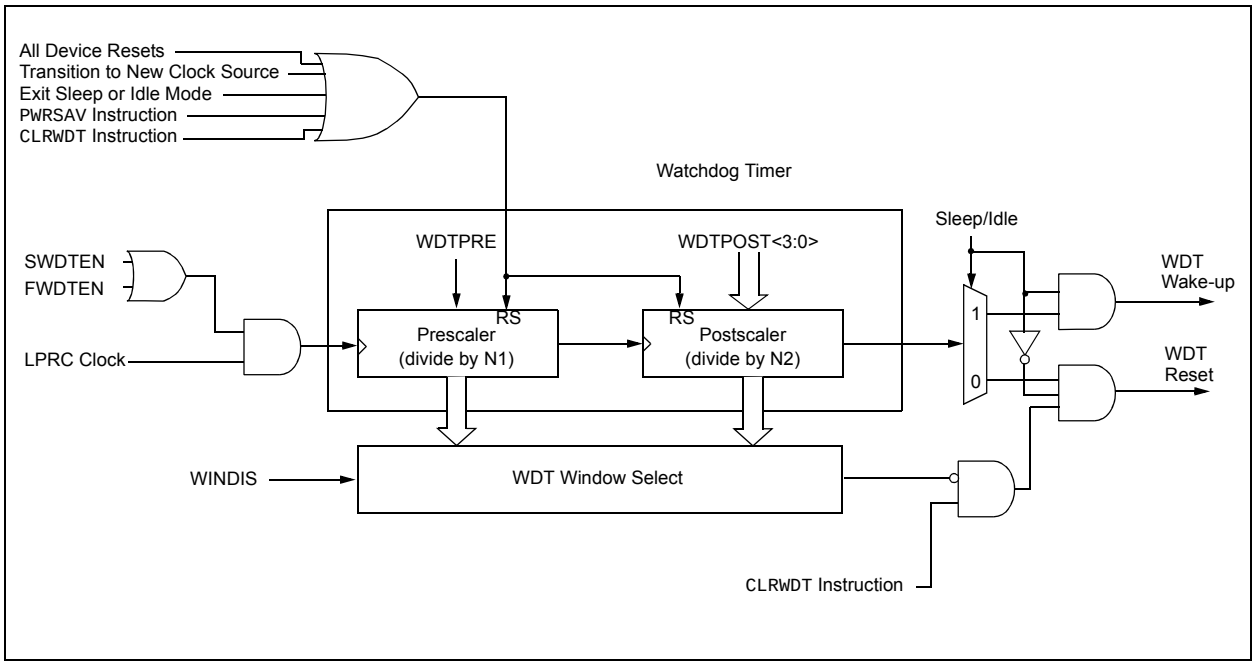
Note:	The CLRWDT and PWRSAV instructions clear the prescaler and postscaler counts when executed.

The WDT is enabled or disabled by the FWDTEN Configuration bit in the FWDT Configuration register. When the FWDTEN Configuration bit is set, the WDT is always enabled.

The WDT can be optionally controlled in software when the FWDTEN Configuration bit has been programmed to '0'. The WDT is enabled in software by setting the SWDTEN control bit (RCON<5>). The SWDTEN control bit is cleared on any device Reset. The software WDT option allows the user to enable the WDT for critical code segments and disable the WDT during non-critical segments for maximum power savings.

Note:	If the WINDIS bit (FWDT<6>) is cleared, the CLRWDT instruction should be executed by the application software only during the last 1/4 of the WDT period. This CLRWDT window can be determined by using a timer. If a CLRWDT instruction is executed before this window, a WDT Reset occurs.

FIGURE 22-2: WDT BLOCK DIAGRAM



24.11 PICkit 2 Development Programmer/Debugger and PICkit 2 Debug Express

The PICkit™ 2 Development Programmer/Debugger is a low-cost development tool with an easy to use interface for programming and debugging Microchip's Flash families of microcontrollers. The full featured Windows® programming interface supports baseline (PIC10F, PIC12F5xx, PIC16F5xx), midrange (PIC12F6xx, PIC16F), PIC18F, PIC24, dsPIC30, dsPIC33, and PIC32 families of 8-bit, 16-bit, and 32-bit microcontrollers, and many Microchip Serial EEPROM products. With Microchip's powerful MPLAB Integrated Development Environment (IDE) the PICkit™ 2 enables in-circuit debugging on most PIC® microcontrollers. In-Circuit-Debugging runs, halts and single steps the program while the PIC microcontroller is embedded in the application. When halted at a breakpoint, the file registers can be examined and modified.

The PICkit 2 Debug Express include the PICkit 2, demo board and microcontroller, hookup cables and CDROM with user's guide, lessons, tutorial, compiler and MPLAB IDE software.

24.12 MPLAB PM3 Device Programmer

The MPLAB PM3 Device Programmer is a universal, CE compliant device programmer with programmable voltage verification at V_{DDMIN} and V_{DDMAX} for maximum reliability. It features a large LCD display (128 x 64) for menus and error messages and a modular, detachable socket assembly to support various package types. The ICSP™ cable assembly is included as a standard item. In Stand-Alone mode, the MPLAB PM3 Device Programmer can read, verify and program PIC devices without a PC connection. It can also set code protection in this mode. The MPLAB PM3 connects to the host PC via an RS-232 or USB cable. The MPLAB PM3 has high-speed communications and optimized algorithms for quick programming of large memory devices and incorporates an MMC card for file storage and data applications.

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

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

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

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

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

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

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

25.1 DC Characteristics

TABLE 25-1: OPERATING MIPS VS. VOLTAGE

Characteristic	VDD Range (in Volts)	Temp Range (in °C)	Max MIPS
			dsPIC33FJXXXGPX06A/X08A/X10A
—	VBOR-3.6V(1)	-40°C to +85°C	40
—	VBOR-3.6V(1)	-40°C to +125°C	40

Note 1: Device is functional at VBORMIN < VDD < VDDMIN. Analog modules such as the ADC will have degraded performance. Device functionality is tested but not characterized. Refer to parameter BO10 in Table 25-11 for the minimum and maximum BOR values.

TABLE 25-2: THERMAL OPERATING CONDITIONS

Rating	Symbol	Min	Typ	Max	Unit
dsPIC33FJXXXGPX06A/X08A/X10A					
Operating Junction Temperature Range	TJ	-40	—	+125	°C
Operating Ambient Temperature Range	TA	-40	—	+85	°C
Extended Temperature Devices					
Operating Junction Temperature Range	TJ	-40	—	+150	°C
Operating Ambient Temperature Range	TA	-40	—	+125	°C
Power Dissipation: Internal chip power dissipation: PINT = VDD x (IDD - Σ IOH) I/O Pin Power Dissipation: I/O = Σ ({VDD - VOH} x IOH) + Σ (VOL x IOL)	PD	PINT + PI/O			W
Maximum Allowed Power Dissipation	PDMAX	(TJ - TA)/θJA			W

TABLE 25-3: THERMAL PACKAGING CHARACTERISTICS

Characteristic	Symbol	Typ	Max	Unit	Notes
Package Thermal Resistance, 100-pin TQFP (14x14x1 mm)	θJA	40	—	°C/W	1
Package Thermal Resistance, 100-pin TQFP (12x12x1 mm)	θJA	40	—	°C/W	1
Package Thermal Resistance, 80-pin TQFP (12x12x1 mm)	θJA	40	—	°C/W	1
Package Thermal Resistance, 64-pin TQFP (10x10x1 mm)	θJA	40	—	°C/W	1
Package Thermal Resistance, 64-pin QFN (9x9x0.9 mm)	θJA	28	—	°C/W	1

Note 1: Junction to ambient thermal resistance, Theta-JA (θJA) numbers are achieved by package simulations.

FIGURE 25-2: EXTERNAL CLOCK TIMING

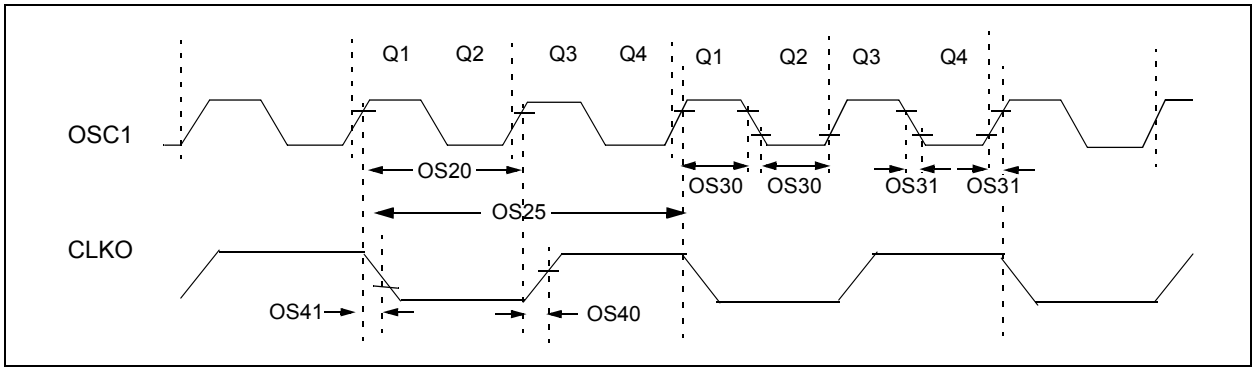


TABLE 25-16: EXTERNAL CLOCK TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristic**	**Min**	**Typ(1)**	**Max**	**Units**	**Conditions**
OS10	FIN	External CLKI Frequency (External clocks allowed only in EC and ECPLL modes)	DC	—	40	MHz	EC
		Oscillator Crystal Frequency	3.5 10 —	— — —	10 40 33	MHz MHz kHz	XT HS SOSC
OS20	TOSC	TOSC = 1/FOSC	12.5	—	DC	ns	—
OS25	TCY	Instruction Cycle Time(2)	25	—	DC	ns	—
OS30	TosL, TosH	External Clock in (OSC1) High or Low Time	0.375 x TOSC	—	0.625 x TOSC	ns	EC
OS31	TosR, TosF	External Clock in (OSC1) Rise or Fall Time	—	—	20	ns	EC
OS40	TckR	CLKO Rise Time(3)	—	5.2	—	ns	—
OS41	TckF	CLKO Fall Time(3)	—	5.2	—	ns	—
OS42	GM	External Oscillator Transconductance(4)	14	16	18	mA/V	VDD = 3.3V TA = +25ºC

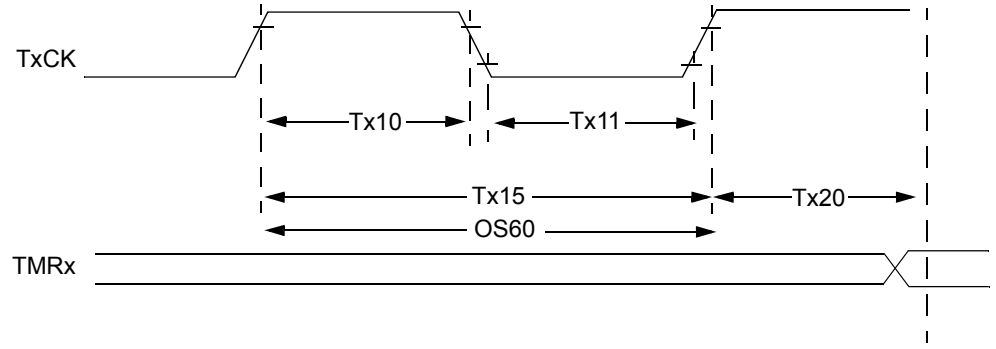
Note 1: Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.

2: Instruction cycle period (TCY) equals two times the input oscillator time-base period. All specified values are based on characterization data for that particular oscillator type under standard operating conditions with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at “min.” values with an external clock applied to the OSC1/CLKI pin. When an external clock input is used, the “max.” cycle time limit is “DC” (no clock) for all devices.

3: Measurements are taken in EC mode. The CLKO signal is measured on the OSC2 pin.

4: Data for this parameter is Preliminary. This parameter is characterized, but not tested in manufacturing.

FIGURE 25-5: TIMER1, 2, 3, 4, 5, 6, 7, 8 AND 9 EXTERNAL CLOCK TIMING CHARACTERISTICS



Note: Refer to Figure 25-1 for load conditions.

TABLE 25-22: TIMER1 EXTERNAL CLOCK TIMING REQUIREMENTS(1)

AC CHARACTERISTICS				Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristic**		**Min**	**Typ**	**Max**	**Units**	**Conditions**
TA10	TTXH	TxCK High Time	Synchronous, no prescaler	TCY + 20	—	—	ns	Must also meet parameter TA15
			Synchronous, with prescaler	(TCY + 20)/N	—	—	ns	
			Asynchronous	20	—	—	ns	
TA11	TTXL	TxCK Low Time	Synchronous, no prescaler	(TCY + 20)/N	—	—	ns	Must also meet parameter TA15
			Synchronous, with prescaler	20	—	—	ns	N = prescale value (1,8,64,256)
			Asynchronous	20	—	—	ns	
TA15	TTXP	TxCK Input Period	Synchronous, no prescaler	2TCY + 40	—	—	ns	—
			Synchronous, with prescaler	Greater of 40 ns or (2TCY + 40)/N	—	—	—	N = prescale value (1, 8, 64, 256)
			Asynchronous	40	—	—	ns	—
OS60	Ft1	SOSC1/T1CK Oscillator Input frequency Range (oscillator enabled by setting TCS bit (T1CON<1>))		DC	—	50	kHz	—
TA20	TCKEXTMRL	Delay from External TxCK Clock Edge to Timer Increment		0.75TCY+40	—	1.75TCY+40	ns	—

Note 1: Timer1 is a Type A.

FIGURE 25-8: OC/PWM MODULE TIMING CHARACTERISTICS

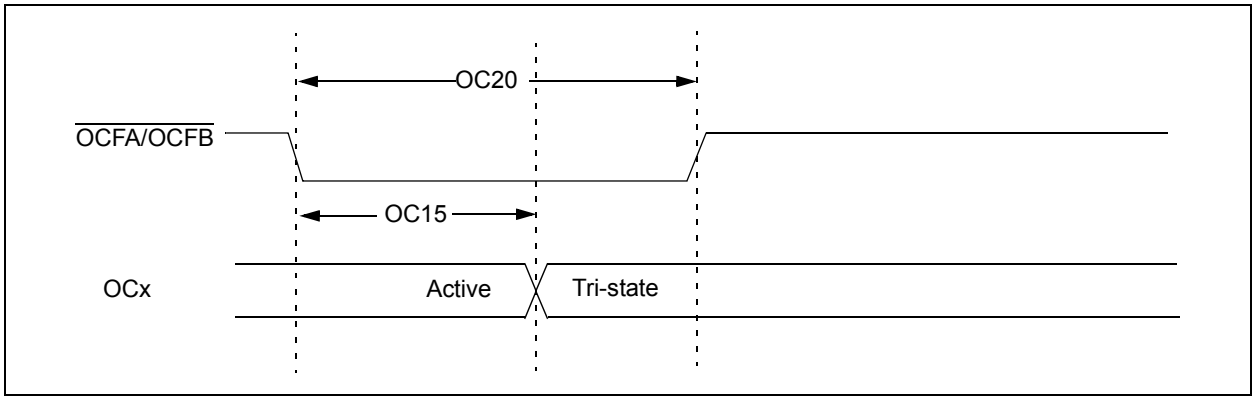
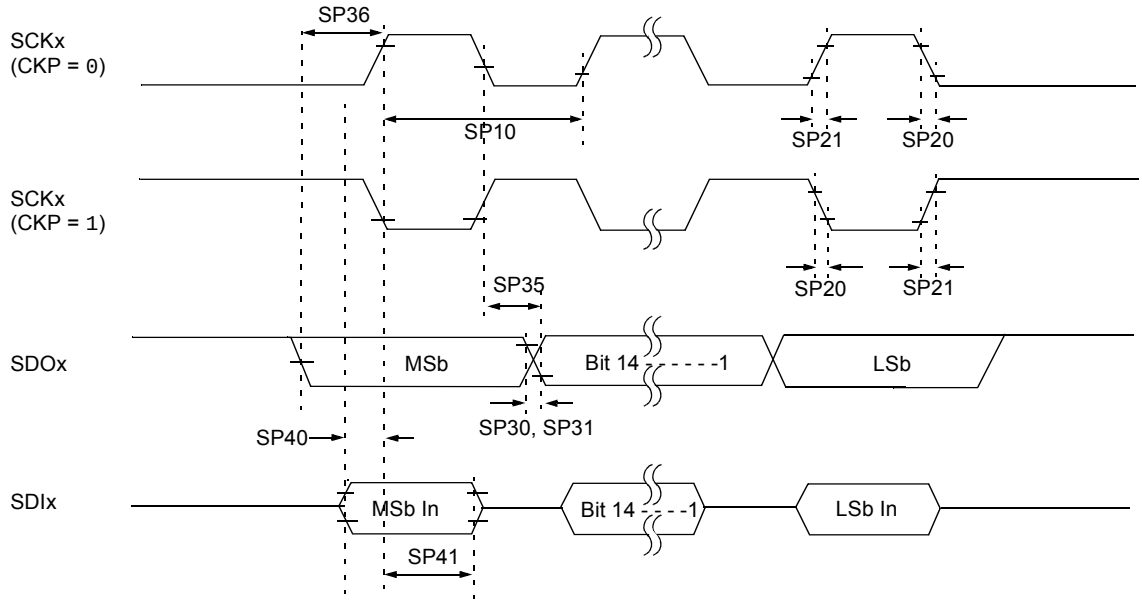


TABLE 25-27: SIMPLE OC/PWM MODE TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristic(1)**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
OC15	TFD	Fault Input to PWM I/O Change	—	—	TCY+20	ns	—
OC20	TFLT	Fault Input Pulse-Width	TCY+20	—	—	ns	—

Note 1: These parameters are characterized but not tested in manufacturing.

FIGURE 25-11: SPIx MASTER MODE (FULL-DUPLEX, CKE = 1, CKP = X, SMP = 1) TIMING CHARACTERISTICS



Note: Refer to Figure 25-1 for load conditions.

TABLE 25-30: SPIx MASTER MODE (FULL-DUPLEX, CKE = 1, CKP = x, SMP = 1) TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristic(1)**	**Min**	**Typ(2)**	**Max**	**Units**	**Conditions**
SP10	TscP	Maximum SCK Frequency	—	—	10	MHz	See **Note 3**
SP20	TscF	SCKx Output Fall Time	—	—	—	ns	See parameter DO32 and **Note 4**
SP21	TscR	SCKx Output Rise Time	—	—	—	ns	See parameter DO31 and **Note 4**
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See parameter DO32 and **Note 4**
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See parameter DO31 and **Note 4**
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	6	20	ns	—
SP36	TdoV2sc, TdoV2scL	SDOx Data Output Setup to First SCKx Edge	30	—	—	ns	—
SP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	30	—	—	ns	—
SP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	30	—	—	ns	—

Note 1: These parameters are characterized, but are not tested in manufacturing.

2: Data in "Typ" column is at 3.3V, 25°C unless otherwise stated.

3: The minimum clock period for SCKx is 100 ns. The clock generated in Master mode must not violate this specification.

4: Assumes 50 pF load on all SPIx pins.

FIGURE 27-5: VOL – 2x DRIVER PINS

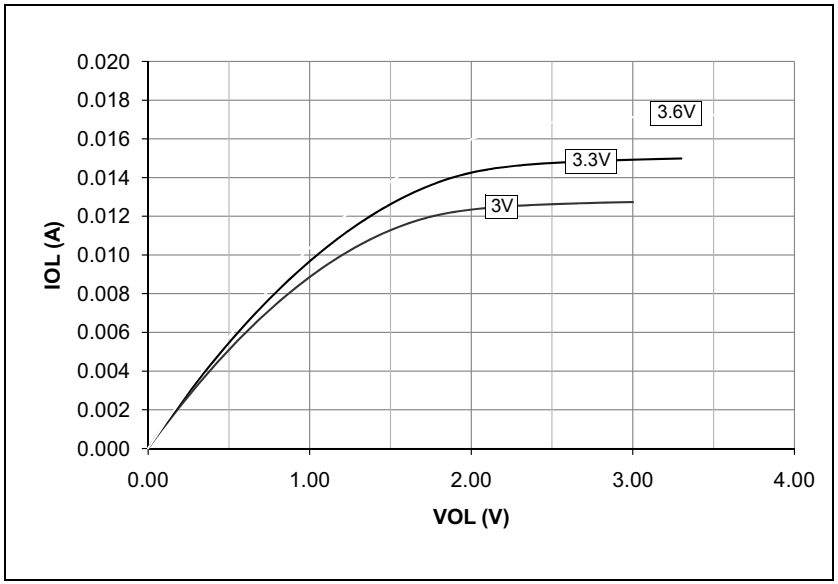


FIGURE 27-6: VOL – 4x DRIVER PINS

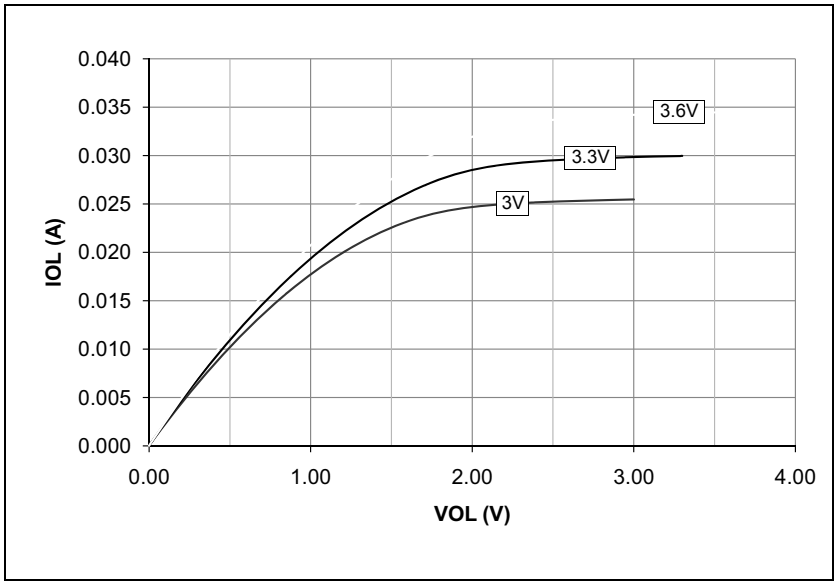


FIGURE 27-7: VOL – 8x DRIVER PINS

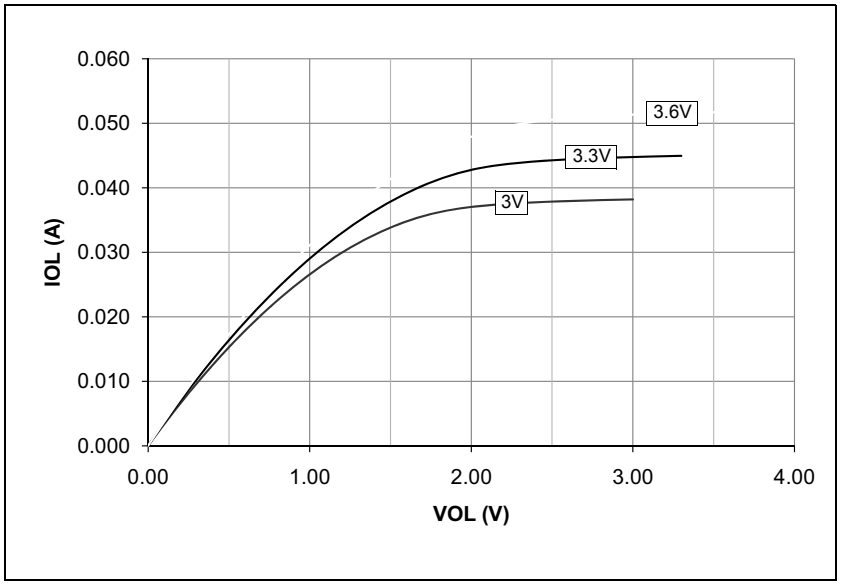
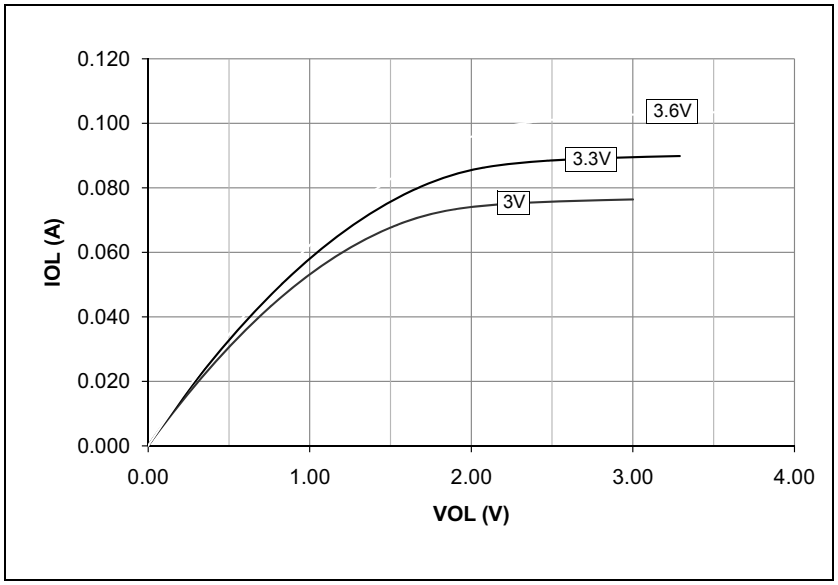


FIGURE 27-8: VOL – 16x DRIVER PINS



PORTA
Register Map ... 64
PORTB
Register Map ... 64
PORTC
Register Map ... 65
PORTD
Register Map ... 65
PORTE
Register Map ... 65
PORTF
Register Map ... 65
PORTG
Register Map ... 66
Power-Saving Features ... 155
Clock Frequency and Switching ... 155
Program Address Space ... 39
Construction ... 72
Data Access from Program Memory
Using Program Space Visibility ... 75
Data Access from Program Memory
Using Table Instructions ... 74
Data Access from, Address Generation ... 73
Memory Map ... 39
Table Read Instructions
TBLRDH ... 74
TBLRDL ... 74
Visibility Operation ... 75
Program Memory
Interrupt Vector ... 40
Organization ... 40
Reset Vector ... 40

R

Reader Response ... 358
Registers
ADxCHS0 (ADCx Input Channel 0 Select ... 247
ADxCHS123 (ADCx Input Channel 1, 2, 3 Select) ... 246
ADxCON1 (ADCx Control 1) ... 241
ADxCON2 (ADCx Control 2) ... 243
ADxCON3 (ADCx Control 3) ... 244
ADxCON4 (ADCx Control 4) ... 245
ADxCSSH (ADCx Input Scan Select High) ... 248
ADxCSSL (ADCx Input Scan Select Low) ... 248
ADxPCFGH (ADCx Port Configuration High) ... 249
ADxPCFGL (ADCx Port Configuration Low) ... 249
CiBUFPNT1 (ECAN Filter 0-3 Buffer Pointer) ... 215
CiBUFPNT2 (ECAN Filter 4-7 Buffer Pointer) ... 216
CiBUFPNT3 (ECAN Filter 8-11 Buffer Pointer) ... 217
CiBUFPNT4 (ECAN Filter 12-15 Buffer Pointer) ... 218
CiCFG1 (ECAN Baud Rate Configuration 1) ... 212
CiCFG2 (ECAN Baud Rate Configuration 2) ... 213
CiCTRL1 (ECAN Control 1) ... 204
CiCTRL2 (ECAN Control 2) ... 205
CiEC (ECAN Transmit/Receive Error Count) ... 211
CiFCTRL (ECAN FIFO Control) ... 207
CiFEN1 (ECAN Acceptance Filter Enable) ... 214
CiFIFO (ECAN FIFO Status) ... 208
CiFMSKSEL1 (ECAN Filter 7-0 Mask
Selection) ... 220, 221
CiINTE (ECAN Interrupt Enable) ... 210
CiINTF (ECAN Interrupt Flag) ... 209
CiRXFnEID (ECAN Acceptance Filter n
Extended Identifier) ... 219
CiRXFnSID (ECAN Acceptance Filter n
Standard Identifier) ... 219
CiRXFUL1 (ECAN Receive Buffer Full 1) ... 223
CiRXFUL2 (ECAN Receive Buffer Full 2) ... 223
CiRXMnEID (ECAN Acceptance Filter Mask n
Extended Identifier) ... 222
CiRXMnSID (ECAN Acceptance Filter Mask n
Standard Identifier) ... 222
CiRXOVF1 (ECAN Receive Buffer Overflow 1) ... 224
CiRXOVF2 (ECAN Receive Buffer Overflow 2) ... 224
CiTRBnDLC (ECAN Buffer n Data
Length Control) ... 227
CiTRBnDm (ECAN Buffer n Data Field Byte m) ... 227
CiTRBnEID (ECAN Buffer n Extended Identifier) ... 226
CiTRBnSID (ECAN Buffer n Standard Identifier) ... 226
CiTRBnSTAT (ECAN Receive Buffer n Status) ... 228
CiTRmnCON (ECAN TX/RX Buffer m Control) ... 225
CiVEC (ECAN Interrupt Code) ... 206
CLKDIV (Clock Divisor) ... 150
CORCON (Core Control) ... 32, 94
DCICON1 (DCI Control 1) ... 231
DCICON2 (DCI Control 2) ... 232
DCICON3 (DCI Control 3) ... 233
DCISTAT (DCI Status) ... 234
DMACS0 (DMA Controller Status 0) ... 141
DMACS1 (DMA Controller Status 1) ... 143
DMAxCNT (DMA Channel x Transfer Count) ... 140
DMAxCON (DMA Channel x Control) ... 137
DMAxPAD (DMA Channel x Peripheral Address) ... 140
DMAxREQ (DMA Channel x IRQ Select) ... 138
DMAxSTA (DMA Channel x RAM Start
Address A) ... 139
DMAxSTB (DMA Channel x RAM Start
Address B) ... 139
DSADR (Most Recent DMA RAM Address) ... 144
I2CxCON (I2Cx Control) ... 190
I2CxMSK (I2Cx Slave Mode Address Mask) ... 194
I2CxSTAT (I2Cx Status) ... 192
ICxCON (Input Capture x Control) ... 176
IEC0 (Interrupt Enable Control 0) ... 106
IEC1 (Interrupt Enable Control 1) ... 108
IEC2 (Interrupt Enable Control 2) ... 110
IEC3 (Interrupt Enable Control 3) ... 112
IEC4 (Interrupt Enable Control 4) ... 113
IFS0 (Interrupt Flag Status 0) ... 98
IFS1 (Interrupt Flag Status 1) ... 100
IFS2 (Interrupt Flag Status 2) ... 102
IFS3 (Interrupt Flag Status 3) ... 104
IFS4 (Interrupt Flag Status 4) ... 105
INTCON1 (Interrupt Control 1) ... 95
INTCON2 (Interrupt Control 2) ... 97
INTTREG Interrupt Control and Status Register ... 132
IPC0 (Interrupt Priority Control 0) ... 114
IPC1 (Interrupt Priority Control 1) ... 115
IPC10 (Interrupt Priority Control 10) ... 124
IPC11 (Interrupt Priority Control 11) ... 125
IPC12 (Interrupt Priority Control 12) ... 126
IPC13 (Interrupt Priority Control 13) ... 127
IPC14 (Interrupt Priority Control 14) ... 128
IPC15 (Interrupt Priority Control 15) ... 129
IPC16 (Interrupt Priority Control 16) ... 130
IPC17 (Interrupt Priority Control 17) ... 131
IPC2 (Interrupt Priority Control 2) ... 116
IPC3 (Interrupt Priority Control 3) ... 117
IPC4 (Interrupt Priority Control 4) ... 118
IPC5 (Interrupt Priority Control 5) ... 119
IPC6 (Interrupt Priority Control 6) ... 120