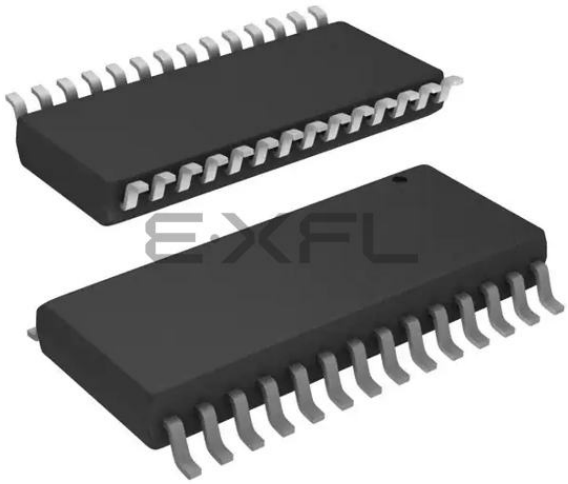
Microchip Technology - DSPIC33FJ64MC802T-I/SO Datasheet



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	Brown-out Detect/Reset, DMA, Motor Control PWM, POR, PWM, QEI, WDT
Number of I/O	21
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	16K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 6x10b/12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-SOIC (0.295", 7.50mm Width)
Supplier Device Package	28-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/dspic33fj64mc802t-i-so

TABLE 4-20: DMA REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
DMA0CON	0380	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—	MODE<1:0>		0000
DMA0REQ	0382	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>							0000
DMA0STA	0384	STA<15:0>																0000
DMA0STB	0386	STB<15:0>																0000
DMA0PAD	0388	PAD<15:0>																0000
DMA0CNT	038A	—	—	—	—	—	—	CNT<9:0>										0000
DMA1CON	038C	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—	MODE<1:0>		0000
DMA1REQ	038E	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>							0000
DMA1STA	0390	STA<15:0>																0000
DMA1STB	0392	STB<15:0>																0000
DMA1PAD	0394	PAD<15:0>																0000
DMA1CNT	0396	—	—	—	—	—	—	CNT<9:0>										0000
DMA2CON	0398	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—	MODE<1:0>		0000
DMA2REQ	039A	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>							0000
DMA2STA	039C	STA<15:0>																0000
DMA2STB	039E	STB<15:0>																0000
DMA2PAD	03A0	PAD<15:0>																0000
DMA2CNT	03A2	—	—	—	—	—	—	CNT<9:0>										0000
DMA3CON	03A4	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—	MODE<1:0>		0000
DMA3REQ	03A6	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>							0000
DMA3STA	03A8	STA<15:0>																0000
DMA3STB	03AA	STB<15:0>																0000
DMA3PAD	03AC	PAD<15:0>																0000
DMA3CNT	03AE	—	—	—	—	—	—	CNT<9:0>										0000
DMA4CON	03B0	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—	MODE<1:0>		0000
DMA4REQ	03B2	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>							0000
DMA4STA	03B4	STA<15:0>																0000
DMA4STB	03B6	STB<15:0>																0000
DMA4PAD	03B8	PAD<15:0>																0000
DMA4CNT	03BA	—	—	—	—	—	—	CNT<9:0>										0000
DMA5CON	03BC	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—	MODE<1:0>		0000
DMA5REQ	03BE	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>							0000
DMA5STA	03C0	STA<15:0>																0000
DMA5STB	03C2	STB<15:0>																0000

Legend: — = unimplemented, read as '0'.

4.8.2 DATA ACCESS FROM PROGRAM 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 space without going through data space. The TBLRDH and TBLWTH instructions are the only method to read or write the upper 8 bits of a program space word as data.

The PC is incremented by two 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-wide word address spaces, residing side by side, each with the same address range. The TBLRDL and TBLWTL access the space that contains the least significant data word. The TBLRDH and TBLWTH access the space that contains the upper data byte.

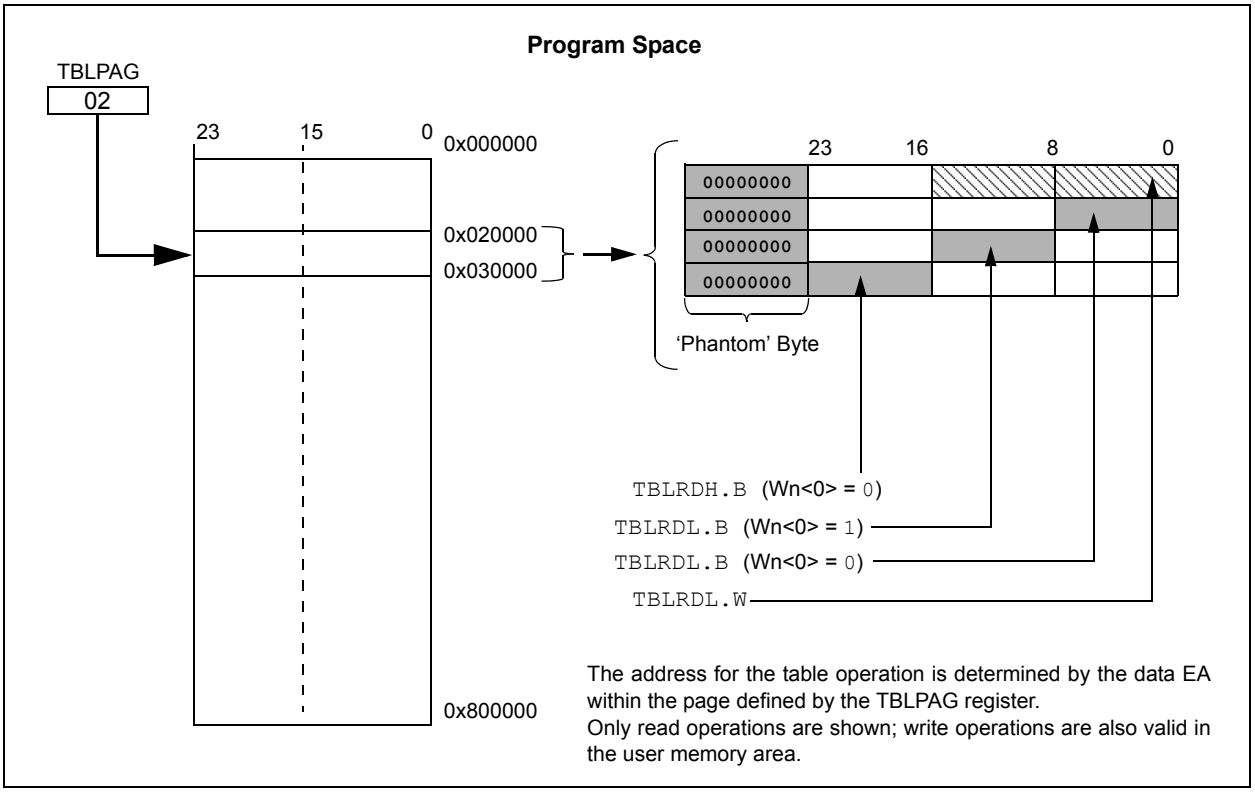
Two table instructions are provided to move byte or word-sized (16-bit) data to and from program space. Both function as either byte or word operations.

- TBLRDL (Table Read Low):
  - In Word mode, this instruction 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'.
- TBLRDH (Table Read High):
  - In Word mode, this instruction maps the entire upper word of a program address (P<23:16>) to a data address. The phantom byte (D<15:8>), is always '0'.
  - In Byte mode, this instruction maps the upper or lower byte of the program word to D<7:0> of the data address, in the TBLRDL instruction. The data is always '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 Page register (TBLPAG). The TBLPAG covers the entire program memory space of the device, including user application 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.

FIGURE 4-10: ACCESSING PROGRAM MEMORY WITH TABLE INSTRUCTIONS



REGISTER 7-7: IFS2: INTERRUPT FLAG STATUS REGISTER 2

U-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0
—	DMA4IF	PMPIF	—	—	—	—	—
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
—	—	—	DMA3IF	C1IF(1)	C1RXIF(1)	SPI2IF	SPI2EIF
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 **DMA4IF:** DMA Channel 4 Data Transfer Complete Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 13 **PMPIF:** Parallel Master Port Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 12-5 **Unimplemented:** Read as '0'

bit 4 **DMA3IF:** DMA Channel 3 Data Transfer Complete Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 3 **C1IF:** ECAN1 Event Interrupt Flag Status bit(1)
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 2 **C1RXIF:** ECAN1 Receive Data Ready Interrupt Flag Status bit(1)
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 1 **SPI2IF:** SPI2 Event Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 0 **SPI2EIF:** SPI2 Error Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

Note 1: Interrupts are disabled on devices without an ECAN™ module.

REGISTER 7-25: IPC11: INTERRUPT PRIORITY CONTROL REGISTER 11

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	DMA4IP<2:0>		
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	PMPIP<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 15-11 **Unimplemented:** Read as '0'

bit 10-8 **DMA4IP<2:0>:** DMA Channel 4 Data Transfer Complete 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 **PMPIP<2:0>:** Parallel Master Port Interrupt Priority bits
111 = Interrupt is priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is priority 1
000 = Interrupt source is disabled

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

NOTES:

11.0 I/O PORTS

Note 1: This data sheet summarizes the features of the dsPIC33FJ32MC302/304, dsPIC33FJ64MCX02/X04 and dsPIC33FJ128MCX02/X04 family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 10. "I/O Ports"** (DS70193) of 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.

All of the device pins (except VDD, VSS, MCLR and OSC1/CLKI) are shared among the peripherals and the parallel I/O ports. All I/O input ports feature Schmitt Trigger inputs for improved noise immunity.

11.1 Parallel I/O (PIO) Ports

Generally, a parallel I/O port that shares a pin with a peripheral is subservient to the peripheral. The peripheral's output buffer data and control signals are provided to a pair of multiplexers. The multiplexers select whether the peripheral or the associated port has ownership of the output data and control signals of the I/O pin. The logic also prevents loop through, in which a port's digital output can drive the input of a peripheral that shares the same pin. Figure 11-1 shows how ports are shared with other peripherals and the associated I/O pin to which they are connected.

When a peripheral is enabled and the peripheral is actively driving an associated pin, the use of the pin as a general purpose output pin is disabled. The I/O pin can be read, but the output driver for the parallel port bit is disabled. If a peripheral is enabled, but the peripheral is not actively driving a pin, that pin can be driven by a port.

All port pins have three registers directly associated with their operation as digital I/O. The data direction register (TRISx) determines whether the pin is an input or an output. If the data direction bit is a '1', then the pin is an input. All port pins are defined as inputs after a Reset. Reads from the latch (LATx) read the latch. Writes to the latch write the latch. Reads from the port (PORTx) read the port pins, while writes to the port pins write the latch.

Any bit and its associated data and control registers that are not valid for a particular device is disabled. This means the corresponding LATx and TRISx registers and the port pin are read as zeros.

When a pin is shared with another peripheral or function that is defined as an input only, it is nevertheless regarded as a dedicated port because there is no other competing source of outputs.

FIGURE 11-1: BLOCK DIAGRAM OF A TYPICAL SHARED PORT STRUCTURE

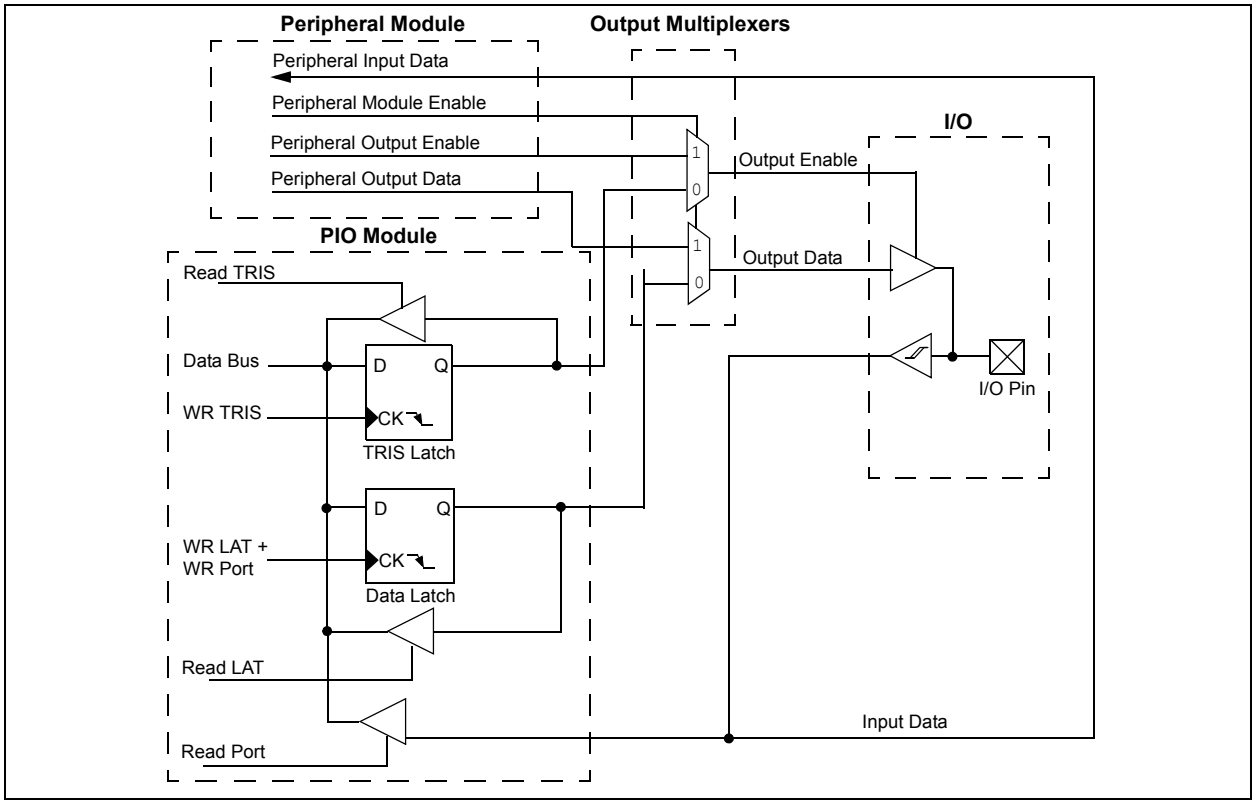


TABLE 11-1: SELECTABLE INPUT SOURCES (MAPS INPUT TO FUNCTION)(1)

Input Name	Function Name	Register	Configuration Bits
External Interrupt 1	INT1	RPINR0	INT1R<4:0>
External Interrupt 2	INT2	RPINR1	INT2R<4:0>
Timer2 External Clock	T2CK	RPINR3	T2CKR<4:0>
Timer3 External Clock	T3CK	RPINR3	T3CKR<4:0>
Timer4 External Clock	T4CK	RPINR4	T4CKR<4:0>
Timer5 External Clock	T5CK	RPINR4	T5CKR<4:0>
Input Capture 1	IC1	RPINR7	IC1R<4:0>
Input Capture 2	IC2	RPINR7	IC2R<4:0>
Input Capture 7	IC7	RPINR10	IC7R<4:0>
Input Capture 8	IC8	RPINR10	IC8R<4:0>
Output Compare Fault A	OCFA	RPINR11	OCFAR<4:0>
PWM1 Fault	$\overline{\text{FLTA1}}$	RPINR12	FLTA1R<4:0>
PWM2 Fault	$\overline{\text{FLTA2}}$	RPINR13	FLTA2R<4:0>
QEI1 Phase A	QEA1	RPINR14	QEA1R<4:0>
QEI1 Phase B	QEB1	RPINR14	QEB1R<4:0>
QEI1 Index	INDX1	RPINR15	INDX1R<4:0>
QEI2 Phase A	QEA2	RPINR16	QEA2R<4:0>
QEI2Phase B	QEB2	RPINR16	QEB2R<4:0>
QEI2 Index	INDX2	RPINR17	INDX2R<4:0>
UART1 Receive	U1RX	RPINR18	U1RXR<4:0>
UART1 Clear To Send	$\overline{\text{U1CTS}}$	RPINR18	U1CTSR<4:0>
UART2 Receive	U2RX	RPINR19	U2RXR<4:0>
UART2 Clear To Send	$\overline{\text{U2CTS}}$	RPINR19	U2CTSR<4:0>
SPI1 Data Input	SDI1	RPINR20	SDI1R<4:0>
SPI1 Clock Input	SCK1	RPINR20	SCK1R<4:0>
SPI1 Slave Select Input	$\overline{\text{SS1}}$	RPINR21	SS1R<4:0>
SPI2 Data Input	SDI2	RPINR22	SDI2R<4:0>
SPI2 Clock Input	SCK2	RPINR22	SCK2R<4:0>
SPI2 Slave Select Input	$\overline{\text{SS2}}$	RPINR23	SS2R<4:0>
ECAN1 Receive	CIRX	RPINR26	CIRXR<4:0>

Note 1: Unless otherwise noted, all inputs use Schmitt input buffers.

11.6.3 CONTROLLING CONFIGURATION CHANGES

Because peripheral remapping can be changed during run time, some restrictions on peripheral remapping are needed to prevent accidental configuration changes. The dsPIC33F devices include three features to prevent alterations to the peripheral map:

- Control register lock sequence
- Continuous state monitoring
- Configuration bit pin select lock

11.6.3.1 Control Register Lock

Under normal operation, writes to the RPINRx and RPORx registers are not allowed. Attempted writes appear to execute normally, but the contents of the registers remain unchanged. To change these registers, they must be unlocked in hardware. The register lock is controlled by the IOLOCK bit (OSCCON<6>). Setting IOLOCK prevents writes to the control registers;
clearing IOLOCK allows writes.

To set or clear the IOLOCK bit, a specific command sequence must be executed:

1. Write 0x46 to OSCCON<7:0>.
2. Write 0x57 to OSCCON<7:0>.
3. Clear (or set) the IOLOCK bit as a single operation.

Note: MPLAB® C30 provides built-in C language functions for unlocking the OSCCON register:

```
__builtin_write_OSCCONL(value)
__builtin_write_OSCCONH(value)
```

See MPLAB IDE Help for more information.

Unlike the similar sequence with the oscillator's LOCK bit, IOLOCK remains in one state until changed. This allows all of the peripheral pin selects to be configured with a single unlock sequence followed by an update to all control registers, then locked with a second lock sequence.

11.6.3.2 Continuous State Monitoring

In addition to being protected from direct writes, the contents of the RPINRx and RPORx registers are constantly monitored in hardware by shadow registers. If an unexpected change in any of the registers occurs (such as cell disturbances caused by ESD or other external events), a configuration mismatch Reset is triggered.

11.6.3.3 Configuration Bit Pin Select Lock

As an additional level of safety, the device can be configured to prevent more than one write session to the RPINRx and RPORx registers. The IOL1WAY Configuration bit (FOSC<5>) blocks the IOLOCK bit from being cleared after it has been set once. If IOLOCK remains set, the register unlock procedure does not execute, and the peripheral pin select control registers cannot be written to. The only way to clear the bit and re-enable peripheral remapping is to perform a device Reset.

In the default (unprogrammed) state, IOL1WAY is set, restricting users to one write session. Programming IOL1WAY allows user applications unlimited access (with the proper use of the unlock sequence) to the peripheral pin select registers.

11.7 I/O Helpful Tips

1. In some cases, certain pins as defined in **TABLE 31-9: "DC Characteristics: I/O Pin Input Specifications"** under "Injection Current", have internal protection diodes to VDD and VSS. The term "Injection Current" is also referred to as "Clamp Current". On designated pins, with sufficient external current limiting precautions by the user, I/O pin input voltages are allowed to be greater or less than the data sheet absolute maximum ratings with nominal VDD with respect to the VSS and VDD supplies. Note that when the user application forward biases either of the high or low side internal input clamp diodes, that the resulting current being injected into the device that is clamped internally by the VDD and VSS power rails, may affect the ADC accuracy by four to six counts.
2. I/O pins that are shared with any analog input pin, (i.e., ANx), are always analog pins by default after any reset. Consequently, any pin(s) configured as an analog input pin, automatically disables the digital input pin buffer. As such, any attempt to read a digital input pin will always return a '0' regardless of the digital logic level on the pin if the analog pin is configured. To use a pin as a digital I/O pin on a shared ANx pin, the user application needs to configure the analog pin configuration registers in the ADC module, (i.e., ADxPCFGL, AD1PCFGH), by setting the appropriate bit that corresponds to that I/O port pin to a '1'. On devices with more than one ADC, both analog pin configurations for both ADC modules must be configured as a digital I/O pin for that pin to function as a digital I/O pin.

Note:	Although it is not possible to use a digital input pin when its analog function is enabled, it is possible to use the digital I/O output function, TRISx = 0x0, while the analog function is also enabled. However, this is not recommended, particularly if the analog input is connected to an external analog voltage source, which would create signal contention between the analog signal and the output pin driver.

3. Most I/O pins have multiple functions. Referring to the device pin diagrams in the data sheet, the priorities of the functions allocated to any pins are indicated by reading the pin name from left-to-right. The left most function name takes precedence over any function to its right in the naming convention. For example: AN16/T2CK/T7CK/RC1. This indicates that AN16 is the highest priority in this example and will supersede all other functions to its right in the list. Those other functions to its right, even if enabled, would not work as long as any other function to its left was enabled. This rule applies to all of the functions listed for a given pin.
4. Each CN pin has a configurable internal weak pull-up resistor. The pull-ups act as a current source connected to the pin, and eliminates the need for external resistors in certain applications. The internal pull-up is to ~(VDD-0.8) not VDD. This is still above the minimum VIH of CMOS and TTL devices.
5. When driving LEDs directly, the I/O pin can source or sink more current than what is specified in the VOH/IOH and VOL/IOL DC characteristic specification. The respective IOH and IOL current rating only applies to maintaining the corresponding output at or above the VOH and at or below the VOL levels. However, for LEDs unlike digital inputs of an externally connected device, they are not governed by the same minimum VIH/VIL levels. An I/O pin output can safely sink or source any current less than that listed in the absolute maximum rating section of the data sheet. For example:

   VOH = 2.4v @ IOH = -8 mA and VDD = 3.3V

   The maximum output current sourced by any 8 mA I/O pin = 12 mA.

LED source current < 12 mA is technically permitted. Refer to the VOH/IOH graphs in **Section 31.0 "Electrical Characteristics"** for additional information.

11.8 I/O Resources

Many useful resources related to Resets are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page, which can be accessed using this link, contains the latest updates and additional information.

Note:	In the event you are not able to access the product page using the link above, enter this URL in your browser: http://www.microchip.com/wwwproducts/Devices.aspx?dDocName=en532315

11.8.1 KEY RESOURCES

- **Section 10. "I/O Ports"** (DS70193)
- Code Samples
- Application Notes
- Software Libraries
- Webinars
- All related dsPIC33F/PIC24H Family Reference Manuals Sections
- Development Tools

12.0 TIMER1

Note 1: This data sheet summarizes the features of the dsPIC33FJ32MC302/304, dsPIC33FJ64MCX02/X04 and dsPIC33FJ128MCX02/X04 family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 11. "Timers"** (DS70205) of 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 Timer1 module is a 16-bit timer, which can serve as the time counter for the real-time clock, or operate as a free-running interval timer/counter.

The Timer1 module has the following unique features over other timers:

- Can be operated from the low power 32 kHz crystal oscillator available on the device.
- Can be operated in Asynchronous Counter mode from an external clock source.
- The external clock input (T1CK) can optionally be synchronized to the internal device clock and the clock synchronization is performed after the prescaler.

The unique features of Timer1 allow it to be used for Real-Time Clock (RTC) applications. A block diagram of Timer1 is shown in Figure 12-1.

The Timer1 module can operate in one of the following modes:

- Timer mode
- Gated Timer mode
- Synchronous Counter mode
- Asynchronous Counter mode

In Timer and Gated Timer modes, the input clock is derived from the internal instruction cycle clock (FCY). In Synchronous and Asynchronous Counter modes, the input clock is derived from the external clock input at the T1CK pin.

The Timer modes are determined by the following bits:

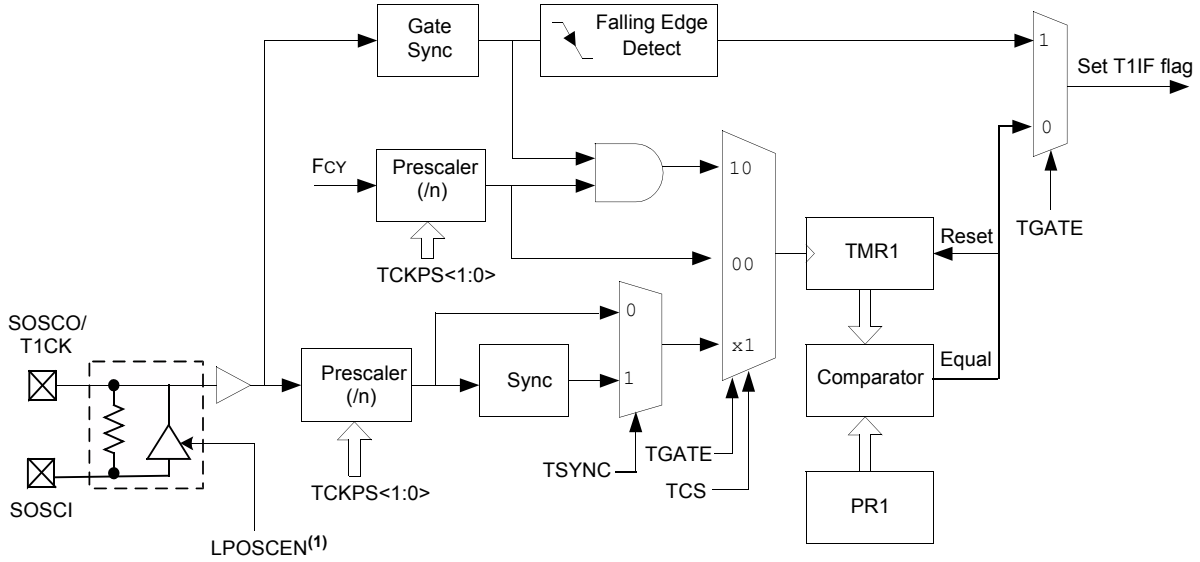
- Timer Clock Source Control bit (TCS): T1CON<1>
- Timer Synchronization Control bit (TSYNC): T1CON<2>
- Timer Gate Control bit (TGATE): T1CON<6>

Timer control bit setting for different operating modes are given in the Table 12-1.

TABLE 12-1: TIMER MODE SETTINGS

Mode	TCS	TGATE	TSYNC
Timer	0	0	x
Gated timer	0	1	x
Synchronous Counter	1	x	1
Asynchronous Counter	1	x	0

FIGURE 12-1: 16-BIT TIMER1 MODULE BLOCK DIAGRAM



Note 1: Refer to **Section 9.0 "Oscillator Configuration"** for information on enabling the secondary oscillator.

18.1 SPI Helpful Tips

1. In Frame mode, if there is a possibility that the master may not be initialized before the slave:
   a) If FRMPOL (SPIxCON2<13>) = 1, use a pull-down resistor on SSx.
   b) If FRMPOL = 0, use a pull-up resistor on SSx.

Note:	This insures that the first frame transmission after initialization is not shifted or corrupted.

2. In non-framed 3-wire mode, (i.e., not using SSx from a master):
   a) If CKP (SPIxCON1<6>) = 1, always place a pull-up resistor on SSx.
   b) If CKP = 0, always place a pull-down resistor on SSx.

Note:	This will insure that during power-up and initialization the master/slave will not lose sync due to an errant SCK transition that would cause the slave to accumulate data shift errors for both transmit and receive appearing as corrupted data.

3. FRMEN (SPIxCON2<15>) = 1 and SSEN (SPIxCON1<7>) = 1 are exclusive and invalid. In Frame mode, SCKx is continuous and the Frame sync pulse is active on the SSx pin, which indicates the start of a data frame.

Note:	Not all third-party devices support Frame mode timing. Refer to the SPI electrical characteristics for details.

4. In Master mode only, set the SMP bit (SPIxCON1<9>) to a '1' for the fastest SPI data rate possible. The SMP bit can only be set at the same time or after the MSTEN bit (SPIxCON1<5>) is set.

To avoid invalid slave read data to the master, the user's master software must guarantee enough time for slave software to fill its write buffer before the user application initiates a master write/read cycle. It is always advisable to preload the SPIxBUF transmit register in advance of the next master transaction cycle. SPIxBUF is transferred to the SPI shift register and is empty once the data transmission begins.

18.2 SPI Resources

Many useful resources related to SPI are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page, which can be accessed using this link, contains the latest updates and additional information.

Note:	In the event you are not able to access the product page using the link above, enter this URL in your browser: http://www.microchip.com/wwwproducts/Devices.aspx?dDocName=en532315

18.2.1 KEY RESOURCES

- **Section 18. "Serial Peripheral Interface (SPI)"** (DS70206)
- Code Samples
- Application Notes
- Software Libraries
- Webinars
- All related dsPIC33F/PIC24H Family Reference Manuals Sections
- Development Tools

21.2 Frame Types

The ECAN module transmits various types of frames which include data messages, or remote transmission requests initiated by the user, as other frames that are automatically generated for control purposes. The following frame types are supported:

- Standard Data Frame:

  A standard data frame is generated by a node when the node wishes to transmit data. It includes an 11-bit Standard Identifier (SID), but not an 18-bit Extended Identifier (EID).

- Extended Data Frame:
- An extended data frame is similar to a standard data frame, but includes an extended identifier as well.
- Remote Frame:
- It is possible for a destination node to request the data from the source. For this purpose, the destination node sends a remote frame with an identifier that matches the identifier of the required data frame. The appropriate data source node sends a data frame as a response to this remote request.
- Error Frame:
- An error frame is generated by any node that detects a bus error. An error frame consists of two fields: an error flag field and an error delimiter field.
- Overload Frame:
- An overload frame can be generated by a node as a result of two conditions. First, the node detects a dominant bit during interframe space which is an illegal condition. Second, due to internal conditions, the node is not yet able to start reception of the next message. A node can generate a maximum of 2 sequential overload frames to delay the start of the next message.
- Interframe Space:
- Interframe space separates a proceeding frame (of whatever type) from a following data or remote frame.

REGISTER 21-5: CiFIFO: ECAN™ FIFO STATUS REGISTER

U-0	U-0	R-0	R-0	R-0	R-0	R-0	R-0
—	—	FBP<5:0>					
bit 15							bit 8

U-0	U-0	R-0	R-0	R-0	R-0	R-0	R-0
—	—	FNRB<5:0>					
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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **FBP<5:0>:** FIFO Buffer Pointer bits

011111 = RB31 buffer
011110 = RB30 buffer
•
•
•
000001 = TRB1 buffer
000000 = TRB0 buffer

bit 7-6 **Unimplemented:** Read as '0'

bit 5-0 **FNRB<5:0>:** FIFO Next Read Buffer Pointer bits

011111 = RB31 buffer
011110 = RB30 buffer
•
•
•
000001 = TRB1 buffer
000000 = TRB0 buffer

25.1 RTCC Module Registers

The RTCC module registers are organized into three categories:

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

25.1.1 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 RTCPTR bits (RCFGCAL<9:8>) to select the desired timer register pair (see Table 25-1).

By writing the RTCVALH byte, the RTCC Pointer value, 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 25-1: 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 ALRMPTR bits (ALCFGRPT<9:8>) to select the desired Alarm register pair (see Table 25-2).

By writing the ALRMVALH byte, the Alarm Pointer value, ALRMPTR<1:0> bits, 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 25-2: ALRMVAL REGISTER MAPPING

ALRMPTR <1:0>	Alarm Value Register Window	
	ALRMVAL<15:8>	ALRMVAL<7:0>
00	ALRMMIN	ALRMSEC
01	ALRMWD	ALRMHR
10	ALRMMNTH	ALRMDAY
11	—	—

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 will decrement the ALRMPTR<1:0> value. 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.

25.1.2 WRITE LOCK

In order to perform a write to any of the RTCC Timer registers, the RTCWREN bit (RCFGCAL<13>) must be set (refer to Example 25-1).

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 1 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 25-1.

EXAMPLE 25-1: SETTING THE RTCWREN BIT

```
MOV     #NVMKEY, W1         ;move the address of NVMKEY into W1
MOV     #0x55, W2
MOV     #0xAA, W3
MOV     W2, [W1]            ;start 55/AA sequence
MOV     W3, [W1]
BSET    RCFGCAL, #13        ;set the RTCWREN bit
```

REGISTER 25-3: ALCFGRPT: ALARM CONFIGURATION 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
ALRMEN	CHIME	AMASK<3:0>				ALRMPTR<1:0>	
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
ARPT<7:0>							
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 **ALRMEN:** Alarm Enable bit
1 = Alarm is enabled (cleared automatically after an alarm event whenever ARPT<7:0> = 0x00 and CHIME = 0)
0 = Alarm is disabled

bit 14 **CHIME:** Chime Enable bit
1 = Chime is enabled; ARPT<7:0> bits are allowed to roll over from 0x00 to 0xFF
0 = Chime is disabled; ARPT<7:0> bits stop once they reach 0x00

bit 13-10 **AMASK<3:0>:** Alarm Mask Configuration bits
11xx = Reserved – do not use
101x = Reserved – do not use
1001 = Once a year (except when configured for February 29th, once every 4 years)
1000 = Once a month
0111 = Once a week
0110 = Once a day
0101 = Every hour
0100 = Every 10 minutes
0011 = Every minute
0010 = Every 10 seconds
0001 = Every second
0000 = Every half second

bit 9-8 **ALRMPTR<1:0>:** Alarm Value Register Window Pointer bits
Points to the corresponding Alarm Value registers when reading ALRMVALH and ALRMVALL registers; the ALRMPTR<1:0> value decrements on every read or write of ALRMVALH until it reaches '00'.
ALRMVAL<15:8>:
11 = Unimplemented
10 = ALRMMNTH
01 = ALRMWD
00 = ALRMMIN
ALRMVAL<7:0>:
11 = Unimplemented
10 = ALRMDAY
01 = ALRMHR
00 = ALRMSEC

bit 7-0 **ARPT<7:0>:** Alarm Repeat Counter Value bits
11111111 = Alarm will repeat 255 more times
•
•
•
00000000 = Alarm will not repeat
The counter decrements on any alarm event. The counter is prevented from rolling over from 0x00 to 0xFF unless CHIME = 1.

REGISTER 25-4: RTCVAL (WHEN RTCPTR<1:0> = 11): YEAR VALUE REGISTER(1)

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

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
YRTEN<3:0>				YRONE<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 15-8 **Unimplemented:** Read as '0'

bit 7-4 **YRTEN<3:0>:** Binary Coded Decimal Value of Year's Tens Digit; contains a value from 0 to 9

bit 3-0 **YRONE<3:0>:** Binary Coded Decimal Value of Year's Ones Digit; contains a value from 0 to 9

Note 1: A write to the YEAR register is only allowed when RTCWREN = 1.

REGISTER 25-5: RTCVAL (WHEN RTCPTR<1:0> = 10): MONTH AND DAY VALUE REGISTER(1)

U-0	U-0	U-0	R-x	R-x	R-x	R-x	R-x
—	—	—	MTHTEN0	MTHONE<3:0>			
bit 15							bit 8

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	DAYTEN<1:0>		DAYONE<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 15-13 **Unimplemented:** Read as '0'

bit 12 **MTHTEN0:** Binary Coded Decimal Value of Month's Tens Digit; contains a value of 0 or 1

bit 11-8 **MTHONE<3:0>:** Binary Coded Decimal Value of Month's Ones Digit; contains a value from 0 to 9

bit 7-6 **Unimplemented:** Read as '0'

bit 5-4 **DAYTEN<1:0>:** Binary Coded Decimal Value of Day's Tens Digit; contains a value from 0 to 3

bit 3-0 **DAYONE<3:0>:** Binary Coded Decimal Value of Day's Ones Digit; contains a value from 0 to 9

Note 1: A write to this register is only allowed when RTCWREN = 1.

Register 27-2: PMMODE: PARALLEL PORT MODE REGISTER

R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BUSY	IRQM<1:0>		INCM<1:0>		MODE16	MODE<1:0>	
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
WAITB<1:0>(1)		WAITM<3:0>				WAITE<1:0>(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 **BUSY:** Busy bit (Master mode only)
1 = Port is busy (not useful when the processor stall is active)
0 = Port is not busy

bit 14-13 **IRQM<1:0>:** Interrupt Request Mode bits
11 = Interrupt generated when Read Buffer 3 is read or Write Buffer 3 is written (Buffered PSP mode) or on a read or write operation when PMA<1:0> = 11 (Addressable PSP mode only)
10 = No interrupt generated, processor stall activated
01 = Interrupt generated at the end of the read/write cycle
00 = No interrupt generated

bit 12-11 **INCM<1:0>:** Increment Mode bits
11 = PSP read and write buffers auto-increment (Legacy PSP mode only)
10 = Decrement ADDR<10:0> by 1 every read/write cycle
01 = Increment ADDR<10:0> by 1 every read/write cycle
00 = No increment or decrement of address

bit 10 **MODE16:** 8/16-bit Mode bit
1 = 16-bit mode: data register is 16 bits, a read or write to the data register invokes two 8-bit transfers
0 = 8-bit mode: data register is 8 bits, a read or write to the data register invokes one 8-bit transfer

bit 9-8 **MODE<1:0>:** Parallel Port Mode Select bits
11 = Master mode 1 (PMCS1, PMRD/$\overline{\text{PMWR}}$, PMENB, PMBE, PMA<x:0> and PMD<7:0>)
10 = Master mode 2 (PMCS1, PMRD, $\overline{\text{PMWR}}$, $\overline{\text{PMBE}}$, PMA<x:0> and PMD<7:0>)
01 = Enhanced PSP, control signals ($\overline{\text{PMRD}}$, $\overline{\text{PMWR}}$, $\overline{\text{PMCS1}}$, PMD<7:0> and PMA<1:0>)
00 = Legacy Parallel Slave Port, control signals ($\overline{\text{PMRD}}$, $\overline{\text{PMWR}}$, $\overline{\text{PMCS1}}$ and PMD<7:0>)

bit 7-6 **WAITB<1:0>:** Data Setup to Read/Write Wait State Configuration bits(1)
11 = Data wait of 4 TCY; multiplexed address phase of 4 TCY
10 = Data wait of 3 TCY; multiplexed address phase of 3 TCY
01 = Data wait of 2 TCY; multiplexed address phase of 2 TCY
00 = Data wait of 1 TCY; multiplexed address phase of 1 TCY

bit 5-2 **WAITM<3:0>:** Read to Byte Enable Strobe Wait State Configuration bits
1111 = Wait of additional 15 TCY
•
•
•
0001 = Wait of additional 1 TCY
0000 = No additional wait cycles (operation forced into one TCY)

bit 1-0 **WAITE<1:0>:** Data Hold After Strobe Wait State Configuration bits(1)
11 = Wait of 4 TCY
10 = Wait of 3 TCY
01 = Wait of 2 TCY
00 = Wait of 1 TCY

Note 1: WAITB and WAITE bits are ignored whenever WAITM3:WAITM0 = 0000.

FIGURE 31-9: OC/PWM MODULE TIMING CHARACTERISTICS

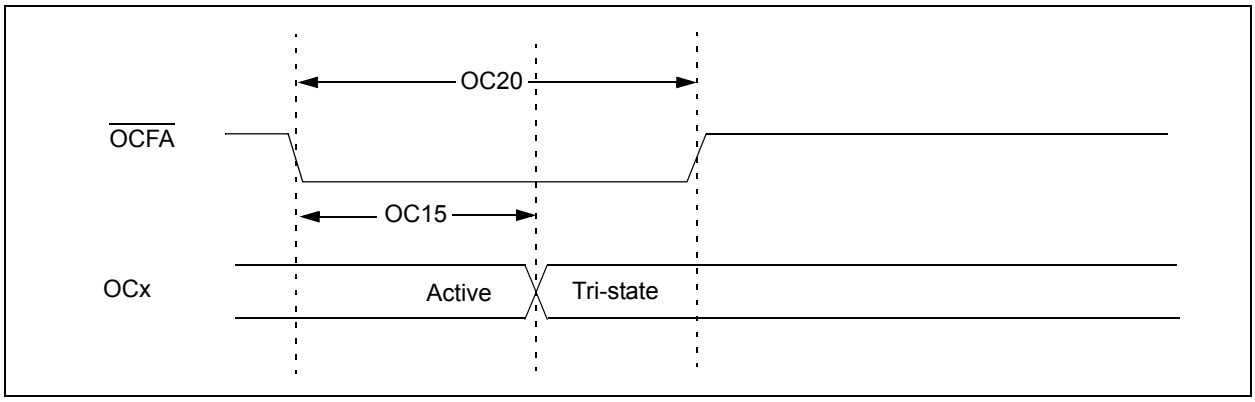


TABLE 31-28: SIMPLE OC/PWM MODE TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -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.

TABLE 31-41: I2Cx BUS DATA TIMING REQUIREMENTS (SLAVE MODE)

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.	**Symbol**	**Characteristic**		**Min**	**Max**	**Units**	**Conditions**
IS10	TLO:SCL	Clock Low Time	100 kHz mode	4.7	—	µs	Device must operate at a minimum of 1.5 MHz
			400 kHz mode	1.3	—	µs	Device must operate at a minimum of 10 MHz
			1 MHz mode(1)	0.5	—	µs	—
IS11	THI:SCL	Clock High Time	100 kHz mode	4.0	—	µs	Device must operate at a minimum of 1.5 MHz
			400 kHz mode	0.6	—	µs	Device must operate at a minimum of 10 MHz
			1 MHz mode(1)	0.5	—	µs	—
IS20	TF:SCL	SDAx and SCLx Fall Time	100 kHz mode	—	300	ns	CB is specified to be from 10 to 400 pF
			400 kHz mode	20 + 0.1 CB	300	ns	
			1 MHz mode(1)	—	100	ns	
IS21	TR:SCL	SDAx and SCLx Rise Time	100 kHz mode	—	1000	ns	CB is specified to be from 10 to 400 pF
			400 kHz mode	20 + 0.1 CB	300	ns	
			1 MHz mode(1)	—	300	ns	
IS25	TSU:DAT	Data Input Setup Time	100 kHz mode	250	—	ns	—
			400 kHz mode	100	—	ns	
			1 MHz mode(1)	100	—	ns	
IS26	THD:DAT	Data Input Hold Time	100 kHz mode	0	—	µs	—
			400 kHz mode	0	0.9	µs	
			1 MHz mode(1)	0	0.3	µs	
IS30	TSU:STA	Start Condition Setup Time	100 kHz mode	4.7	—	µs	Only relevant for Repeated Start condition
			400 kHz mode	0.6	—	µs	
			1 MHz mode(1)	0.25	—	µs	
IS31	THD:STA	Start Condition Hold Time	100 kHz mode	4.0	—	µs	After this period, the first clock pulse is generated
			400 kHz mode	0.6	—	µs	
			1 MHz mode(1)	0.25	—	µs	
IS33	TSU:STO	Stop Condition Setup Time	100 kHz mode	4.7	—	µs	—
			400 kHz mode	0.6	—	µs	
			1 MHz mode(1)	0.6	—	µs	
IS34	THD:STO	Stop Condition Hold Time	100 kHz mode	4000	—	ns	—
			400 kHz mode	600	—	ns	
			1 MHz mode(1)	250		ns	
IS40	TAA:SCL	Output Valid From Clock	100 kHz mode	0	3500	ns	—
			400 kHz mode	0	1000	ns	
			1 MHz mode(1)	0	350	ns	
IS45	TBF:SDA	Bus Free Time	100 kHz mode	4.7	—	µs	Time bus must be free before a new transmission can start
			400 kHz mode	1.3	—	µs	
			1 MHz mode(1)	0.5	—	µs	
IS50	CB	Bus Capacitive Loading		—	400	pF	—

Note 1: Maximum pin capacitance = 10 pF for all I2Cx pins (for 1 MHz mode only).

32.0 DC AND AC DEVICE CHARACTERISTICS GRAPHS

Note: The graphs provided following this note are a statistical summary based on a limited number of samples and are provided for design guidance purposes only. The performance characteristics listed herein are not tested or guaranteed. In some graphs, the data presented may be outside the specified operating range (e.g., outside specified power supply range) and therefore, outside the warranted range.

FIGURE 32-1: VOH – 2x DRIVER PINS

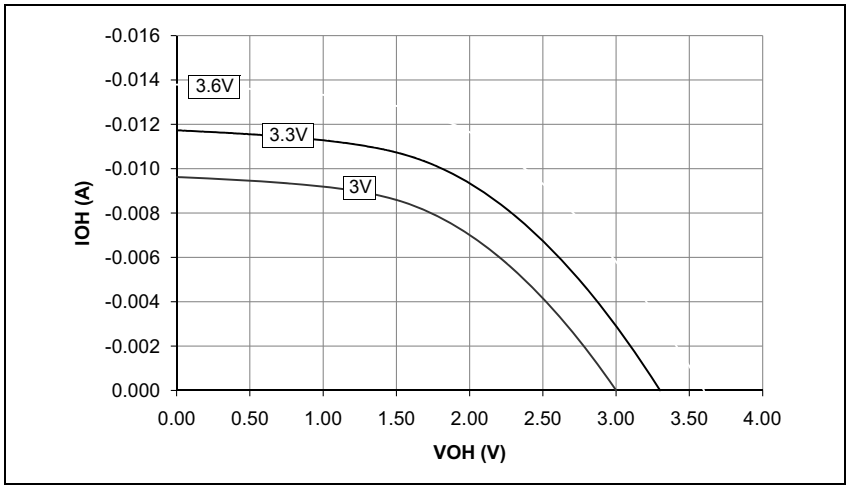


FIGURE 32-2: VOH – 4x DRIVER PINS

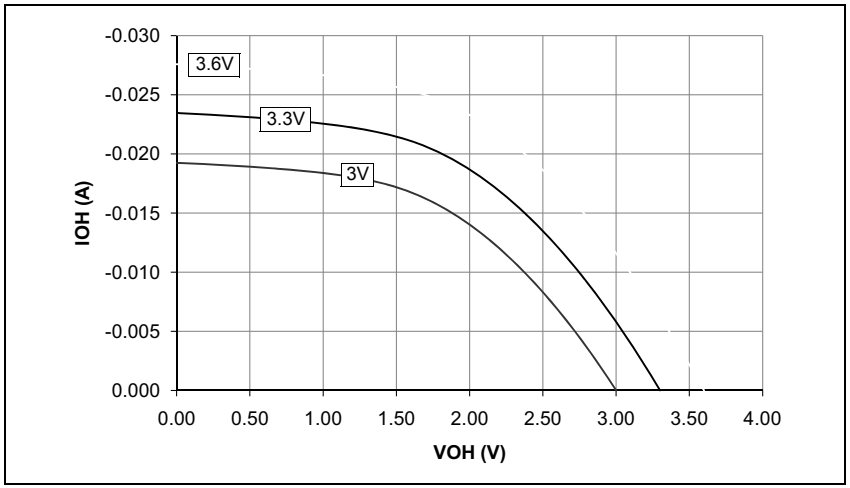


FIGURE 32-3: VOH – 8x DRIVER PINS

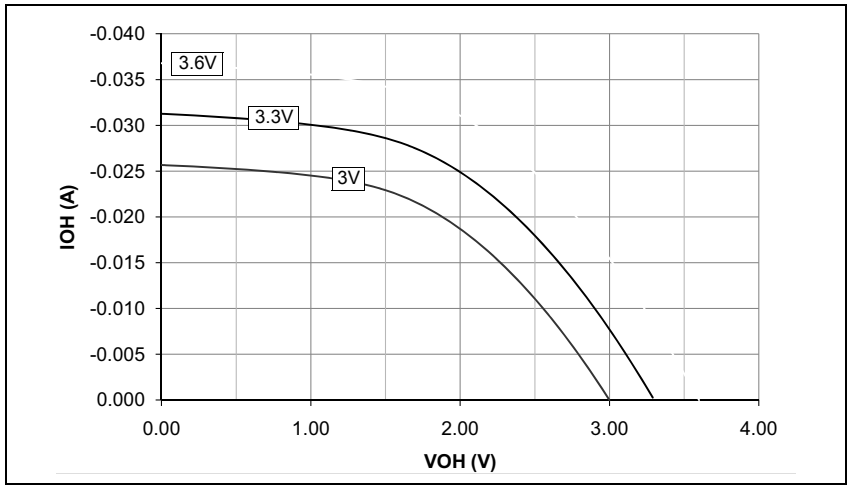


FIGURE 32-4: VOH – 16x DRIVER PINS