

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 "<u>Embedded -</u> <u>Microcontrollers</u>"

Details

E·XFI

Product Status	Active
Core Processor	dsPIC
Core Size	16-Bit
Speed	20 MIPS
Connectivity	I ² C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, Motor Control PWM, QEI, POR, PWM, WDT
Number of I/O	20
Program Memory Size	24KB (8K x 24)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	2.5V ~ 5.5V
Data Converters	A/D 6x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°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/dspic30f3010-20e-so

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong

dsPIC30F3010/3011

High Performance, 16-Bit Digital Signal Controllers

Note: This data sheet summarizes features of this group of dsPIC30F devices and is not intended to be a complete reference source. For more information on the CPU, peripherals, register descriptions and general device functionality, refer to the "dsPIC30F Family Reference Manual" (DS70046). For more information on the device instruction set and programming, refer to the "16-bit MCU and DSC Programmer's Reference Manual" (DS70157).

High-Performance Modified RISC CPU:

- Modified Harvard Architecture
- C Compiler Optimized Instruction Set Architecture with Flexible Addressing modes
- 83 Base Instructions
- 24-bit Wide Instructions, 16-bit Wide Data Path
- 24 Kbytes On-Chip Flash Program Space (8K instruction words)
- 1 Kbyte of On-Chip Data RAM
- 1 Kbyte of Nonvolatile Data EEPROM
- 16 x 16-bit Working Register Array
- Up to 30 MIPs Operation:
 - DC to 40 MHz external clock input
 - 4 MHz-10 MHz oscillator input with PLL active (4x, 8x, 16x)
- 29 Interrupt Sources
 - 3 external interrupt sources
 - 8 user-selectable priority levels for each interrupt source
 - 4 processor trap sources

DSP Engine Features:

- Dual Data Fetch
- Accumulator Write Back for DSP Operations
- · Modulo and Bit-Reversed Addressing modes
- Two, 40-bit Wide Accumulators with Optional saturation Logic
- 17-bit x 17-bit Single-Cycle Hardware Fractional/ Integer Multiplier
- All DSP Instructions Single Cycle
- ±16-bit Single-Cycle Shift

Peripheral Features:

- High-Current Sink/Source I/O Pins: 25 mA/25 mA
- Timer module with Programmable Prescaler:
 - Five 16-bit timers/counters; optionally pair 16-bit timers into 32-bit timer modules
- 16-bit Capture Input Functions
- 16-bit Compare/PWM Output Functions
- 3-Wire SPI modules (supports 4 Frame modes)
- I²C[™] module Supports Multi-Master/Slave mode and 7-bit/10-bit Addressing
- 2 UART modules with FIFO Buffers

Motor Control PWM Module Features:

- 6 PWM Output Channels
 - Complementary or Independent Output modes
 - Edge and Center-Aligned modes
- 3 Duty Cycle Generators
- Dedicated Time Base
- Programmable Output Polarity
- Dead-Time Control for Complementary mode
- Manual Output Control
- Trigger for A/D Conversions

Quadrature Encoder Interface Module Features:

- Phase A, Phase B and Index Pulse Input
- 16-bit Up/Down Position Counter
- Count Direction Status
- Position Measurement (x2 and x4) mode
- Programmable Digital Noise Filters on Inputs
- Alternate 16-bit Timer/Counter mode
- Interrupt on Position Counter Rollover/Underflow

Analog Features:

- 10-bit Analog-to-Digital Converter (ADC) with 4 Sample and Hold (S&H) Inputs:
 - 1 Msps conversion rate
 - 9 input channels
 - Conversion available during Sleep and Idle
- Programmable Brown-out Reset

2.0 CPU ARCHITECTURE OVERVIEW

Note: This data sheet summarizes features of this group of dsPIC30F devices and is not intended to be a complete reference source. For more information on the CPU, peripherals, register descriptions and general device functionality, refer to the "dsPIC30F Family Reference Manual" (DS70046). For more information on the device instruction set and programming, refer to the "16-bit MCU and DSC Programmer's Reference Manual" (DS70157).

2.1 Core Overview

The core has a 24-bit instruction word. The Program Counter (PC) is 23 bits wide with the Least Significant bit (LSb) always clear (see **Section 3.1 "Program Address Space"**), and the Most Significant bit (MSb) is ignored during normal program execution, except for certain specialized instructions. Thus, the PC can address up to 4M instruction words of user program space. An instruction prefetch mechanism is used to help maintain throughput. Program loop constructs, free from loop count management overhead, are supported using the DO and REPEAT instructions, both of which are interruptible at any point.

The working register array consists of 16x16-bit registers, each of which can act as data, address or offset registers. One working register (W15) operates as a Software Stack Pointer (SP) for interrupts and calls.

The data space is 64 Kbytes (32K words) and is split into two blocks, referred to as X and Y data memory. Each block has its own independent Address Generation Unit (AGU). Most instructions operate solely through the X memory AGU, which provides the appearance of a single unified data space. The Multiply-Accumulate (MAC) class of dual source DSP instructions operate through both the X and Y AGUs, splitting the data address space into two parts (see **Section 3.2 "Data Address Space"**). The X and Y data space boundary is device specific and cannot be altered by the user. Each data word consists of 2 bytes, and most instructions can address data either as words or bytes.

There are two methods of accessing data stored in program memory:

• The upper 32 Kbytes of data space memory can be mapped into the lower half (user space) of program space at any 16K program word boundary, defined by the 8-bit Program Space Visibility Page (PSVPAG) register. This lets any instruction access program space as if it were data space, with a limitation that the access requires an additional cycle. Moreover, only the lower 16 bits of each instruction word can be accessed using this method. Linear indirect access of 32K word pages within program space is also possible using any working register, via table read and write instructions.
Table read and write instructions can be used to access all 24 bits of an instruction word.

Overhead-free circular buffers (Modulo Addressing) are supported in both X and Y address spaces. This is primarily intended to remove the loop overhead for DSP algorithms.

The X AGU also supports Bit-Reversed Addressing on destination effective addresses, to greatly simplify input or output data reordering for radix-2 FFT algorithms. Refer to **Section 4.0 "Address Generator Units"** for details on Modulo and Bit-Reversed addressing.

The core supports Inherent (no operand), Relative, Literal, Memory Direct, Register Direct, Register Indirect, Register Offset and Literal Offset Addressing modes. Instructions are associated with predefined addressing modes, depending upon their functional requirements.

For most instructions, the core is capable of executing a data (or program data) memory read, a working register (data) read, a data memory write and a program (instruction) memory read per instruction cycle. As a result, 3 operand instructions are supported, allowing C = A + B operations to be executed in a single cycle.

A DSP engine has been included to significantly enhance the core arithmetic capability and throughput. It features a high-speed 17-bit by 17-bit multiplier, a 40-bit ALU, two 40-bit saturating accumulators and a 40-bit bidirectional barrel shifter. Data in the accumulator or any working register can be shifted up to 16 bits right or 16 bits left in a single cycle. The DSP instructions operate seamlessly with all other instructions and have been designed for optimal real-time performance. The MAC class of instructions can concurrently fetch two data operands from memory, while multiplying two W registers. To enable this concurrent fetching of data operands, the data space has been split for these instructions and linear for all others. This has been achieved in a transparent and flexible manner, by dedicating certain working registers to each address space for the MAC class of instructions.

The core does not support a multi-stage instruction pipeline. However, a single stage instruction prefetch mechanism is used, which accesses and partially decodes instructions a cycle ahead of execution, in order to maximize available execution time. Most instructions execute in a single cycle, with certain exceptions.

The core features a vectored exception processing structure for traps and interrupts, with 62 independent vectors. The exceptions consist of up to 8 traps (of which 4 are reserved) and 54 interrupts. Each interrupt is prioritized based on a user assigned priority between 1 and 7 (1 being the lowest priority and 7 being the highest) in conjunction with a predetermined 'natural order'. Traps have fixed priorities, ranging from 8 to 15.

2.2 Programmer's Model

The programmer's model is shown in Figure 2-1 and consists of 16x16-bit working registers (W0 through W15), 2x40-bit accumulators (ACCA and ACCB), STATUS Register (SR), Data Table Page register (TBLPAG), Program Space Visibility Page register (PSVPAG), DO and REPEAT registers (DOSTART, DOEND, DCOUNT and RCOUNT) and Program Counter (PC). The working registers can act as Data, Address or Offset registers. All registers are memory mapped. W0 acts as the W register for file register addressing.

Some of these registers have a Shadow register associated with each of them, as shown in Figure 2-1. The Shadow register is used as a temporary holding register and can transfer its contents to or from its host register upon the occurrence of an event. None of the Shadow registers are accessible directly. The following rules apply for transfer of registers into and out of shadows.

- PUSH.S and POP.S W0, W1, W2, W3, SR (DC, N, OV, Z and C bits only) are transferred.
- DO instruction DOSTART, DOEND, DCOUNT shadows are pushed on loop start, and popped on loop end.

When a byte operation is performed on a working register, only the Least Significant Byte (LSB) of the target register is affected. However, a benefit of memory mapped working registers is that both the Least and Most Significant Bytes can be manipulated through byte-wide data memory space accesses.

2.2.1 SOFTWARE STACK POINTER/ FRAME POINTER

The dsPIC DSC devices contain a software stack. W15 is the dedicated Software Stack Pointer, and will be automatically modified by exception processing and subroutine calls and returns. However, W15 can be referenced by any instruction in the same manner as all other W registers. This simplifies the reading, writing and manipulation of the Stack Pointer (e.g., creating stack frames).

Note:	In order to protect against misaligned
	stack accesses, W15<0> is always clear.

W15 is initialized to 0x0800 during a Reset. The user may reprogram the SP during initialization to any location within data space.

W14 has been dedicated as a Stack Frame Pointer as defined by the LNK and ULNK instructions. However, W14 can be referenced by any instruction in the same manner as all other W registers.

2.2.2 STATUS REGISTER

The dsPIC DSC core has a 16-bit STATUS Register (SR), the LSB of which is referred to as the SR Low Byte (SRL) and the MSB as the SR High Byte (SRH). See Figure 2-1 for SR layout.

SRL contains all the MCU ALU operation status flags (including the Z bit), as well as the CPU Interrupt Priority Level status bits, IPL<2:0>, and the Repeat Active status bit, RA. During exception processing, SRL is concatenated with the MSB of the PC to form a complete word value which is then stacked.

The upper byte of the SR register contains the DSP adder/subtracter status bits, the DO Loop Active bit (DA) and the Digit Carry (DC) status bit.

2.2.3 PROGRAM COUNTER

The Program Counter is 23 bits wide. Bit 0 is always clear. Therefore, the PC can address up to 4M instruction words.

2.4.1 MULTIPLIER

The 17x17-bit multiplier is capable of signed or unsigned operation and can multiplex its output using a scaler to support either 1.31 fractional (Q31) or 32-bit integer results. Unsigned operands are zero-extended into the 17th bit of the multiplier input value. Signed operands are sign-extended into the 17th bit of the multiplier input value. The output of the 17x17-bit multiplier/scaler is a 33-bit value, which is signextended to 40 bits. Integer data is inherently represented as a signed two's complement value, where the MSB is defined as a sign bit. Generally speaking, the range of an N-bit two's complement integer is -2^{N-1} to $2^{N-1} - 1$. For a 16-bit integer, the data range is -32768 (0x8000) to 32767 (0x7FFF), including 0. For a 32-bit integer, the data range is -2,147,483,648 (0x8000 0000) to 2,147,483,645 (0x7FFF FFFF).

When the multiplier is configured for fractional multiplication, the data is represented as a two's complement fraction, where the MSB is defined as a sign bit and the radix point is implied to lie just after the sign bit (QX format). The range of an N-bit two's complement fraction with this implied radix point is -1.0 to $(1-2^{1-N})$. For a 16-bit fraction, the Q15 data range is -1.0 (0x8000) to 0.999969482 (0x7FFF), including 0 and has a precision of 3.01518x10⁻⁵. In Fractional mode, a 16x16 multiply operation generates a 1.31 product, which has a precision of 4.65661x10⁻¹⁰.

The same multiplier is used to support the MCU multiply instructions, which includes integer 16-bit signed, unsigned and mixed sign multiplies.

The MUL instruction may be directed to use byte or word-sized operands. Byte operands will direct a 16-bit result, and word operands will direct a 32-bit result to the specified register(s) in the W array.

2.4.2 DATA ACCUMULATORS AND ADDER/SUBTRACTER

The data accumulator consists of a 40-bit adder/subtracter with automatic sign extension logic. It can select one of two accumulators (A or B) as its preaccumulation source and post-accumulation destination. For the ADD and LAC instructions, the data to be accumulated or loaded can be optionally scaled via the barrel shifter, prior to accumulation.

2.4.2.1 Adder/Subtracter, Overflow and Saturation

The adder/subtracter is a 40-bit adder with an optional zero input into one side and either true or complement data into the other input. In the case of addition, the carry/borrow input is active-high and the other input is true data (not complemented), whereas in the case of subtraction, the carry/borrow input is active-low and the other input is complemented. The adder/subtracter generates overflow status bits, SA/SB and OA/OB, which are latched and reflected in the STATUS register.

- Overflow from bit 39: this is a catastrophic overflow in which the sign of the accumulator is destroyed.
- Overflow into guard bits 32 through 39: this is a recoverable overflow. This bit is set whenever all the guard bits are not identical to each other.

The adder has an additional saturation block which controls accumulator data saturation, if selected. It uses the result of the adder, the overflow status bits described above, and the SATA/B (CORCON<7:6>) and ACCSAT (CORCON<4>) mode control bits to determine when and to what value to saturate.

Six STATUS register bits have been provided to support saturation and overflow; they are:

- 1. OA: ACCA overflowed into guard bits
- 2. OB: ACCB overflowed into guard bits
- 3. SA: ACCA saturated (bit 31 overflow and saturation) or

ACCA overflowed into guard bits and saturated (bit 39 overflow and saturation)

SB: ACCB saturated (bit 31 overflow and saturation)

ACCB overflowed into guard bits and saturated (bit 39 overflow and saturation)

5. OAB: Logical OR of OA and OB

4

or

6. SAB: Logical OR of SA and SB

The OA and OB bits are modified each time data passes through the adder/subtracter. When set, they indicate that the most recent operation has overflowed into the accumulator guard bits (bits 32 through 39). The OA and OB bits can also optionally generate an arithmetic warning trap when set and the corresponding overflow trap flag enable bit (OVATE, OVBTE) in the INTCON1 register (refer to **Section 5.0 "Interrupts"**) is set. This allows the user to take immediate action, for example, to correct system gain.

4.2.3 MODULO ADDRESSING APPLICABILITY

Modulo Addressing can be applied to the Effective Address (EA) calculation associated with any W register. It is important to realize that the address boundaries check for addresses less than or greater than the upper (for incrementing buffers) and lower (for decrementing buffers) boundary addresses (not just equal to). Address changes may, therefore, jump beyond boundaries and still be adjusted correctly.

Note: The modulo corrected effective address is written back to the register only when Pre-Modify or Post-Modify Addressing mode is used to compute the effective address. When an address offset (e.g., [W7 + W2]) is used, Modulo Addressing correction is performed, but the contents of the register remains unchanged.

4.3 Bit-Reversed Addressing

Bit-Reversed Addressing is intended to simplify data re-ordering for radix-2 FFT algorithms. It is supported by the X AGU for data writes only.

The modifier, which may be a constant value or register contents, is regarded as having its bit order reversed. The address source and destination are kept in normal order. Thus, the only operand requiring reversal is the modifier.

4.3.1 BIT-REVERSED ADDRESSING IMPLEMENTATION

Bit-Reversed Addressing is enabled when:

- BWM (W register selection) in the MODCON register is any value other than 15 (the stack can not be accessed using Bit-Reversed Addressing) and
- 2. the BREN bit is set in the XBREV register and
- 3. the addressing mode used is Register Indirect with Pre-Increment or Post-Increment.

If the length of a bit-reversed buffer is $M = 2^N$ bytes, then the last 'N' bits of the data buffer start address must be zeros.

XB<14:0> is the bit-reversed address modifier or 'pivot point' which is typically a constant. In the case of an FFT computation, its value is equal to half of the FFT data buffer size.

Note:	All bit-reversed EA calculations assume							
	word-sized data (LSb of every EA is							
	always clear). The XB value is scaled							
	accordingly to generate compatible (byte)							
	addresses.							

When enabled, Bit-Reversed Addressing will only be executed for Register Indirect with Pre-Increment or Post-Increment Addressing and word-sized data writes. It will not function for any other addressing mode or for byte-sized data, and normal addresses will be generated instead. When Bit-Reversed Addressing is active, the W Address Pointer will always be added to the address modifier (XB) and the offset associated with the Register Indirect Addressing mode will be ignored. In addition, as word-sized data is a requirement, the LSb of the EA is ignored (and always clear).

Note: Modulo Addressing and Bit-Reversed Addressing should not be enabled together. In the event that the user attempts to do this, Bit-Reversed Addressing will assume priority when active for the X WAGU, and X WAGU Modulo Addressing will be disabled. However, Modulo Addressing will continue to function in the X RAGU.

If Bit-Reversed Addressing has already been enabled by setting the BREN (XBREV<15>) bit, then a write to the XBREV register should not be immediately followed by an indirect read operation using the W register that has been designated as the Bit-Reversed Pointer.



FIGURE 4-2: BIT-REVERSED ADDRESS EXAMPLE

7.3.2 WRITING A BLOCK OF DATA EEPROM

To write a block of data EEPROM, write to all sixteen latches first, then set the NVMCON register and program the block.

		_	
MOV	<pre>#LOW_ADDR_WORD,W0</pre>	;	Init pointer
MOV	#HIGH_ADDR_WORD,W1		
MOV	W1 TBLPAG		
MOV	#data1,W2	;	Get 1st data
TBLWTL	W2 [W0]++	;	write data
MOV	#data2,W2	;	Get 2nd data
TBLWTL	W2 [W0]++	;	write data
MOV	#data3,W2	;	Get 3rd data
TBLWTL	W2 [W0]++	;	write data
MOV	#data4,W2	;	Get 4th data
TBLWTL	W2 [W0]++	;	write data
MOV	#data5,W2	;	Get 5th data
TBLWTL	W2 [W0]++	;	write data
MOV	#data6,W2	;	Get 6th data
TBLWTL	W2,[W0]++	;	write data
MOV	#data7,W2	;	Get 7th data
TBLWTL	W2,[W0]++	;	write data
MOV	#data8,W2	;	Get 8th data
TBLWTL	W2,[W0]++	;	write data
MOV	#data9,W2	;	Get 9th data
TBLWTL	W2,[W0]++	;	write data
MOV	#data10,W2	;	Get 10th data
TBLWTL	W2,[W0]++	;	write data
MOV	#datal1,W2	;	Get 11th data
TBLWTL	W2,[W0]++	;	write data
MOV	#data12,W2	;	Get 12th data
TBLWTL	W2,[W0]++	;	write data
MOV	#data13,W2	;	Get 13th data
TBLWTL	W2,[W0]++	;	write data
MOV	#data14,W2	;	Get 14th data
TBLWTL	W2,[W0]++	;	write data
MOV	#data15,W2	;	Get 15th data
TBLWTL	W2,[W0]++	;	write data
MOV	#data16,W2	;	Get 16th data
TBLWTL	W2,[W0]++	;	write data. The NVMADR captures last table access address.
MOV	#0x400A,W0	;	Select data EEPROM for multi word op
MOV	WU NVMCON	;	Operate Key to allow program operation
DISI	#5	;	Block all interrupts with priority <7
		;	for next 5 instructions
MOV	#0x55,W0		
MOV	WU NVMKEY	;	Write the Ux55 Key
MOV	#UXAA,W⊥		White the Oran least
MOV	WI NVMKEY	;	Write the UXAA Key
BSET	NVMCON, #WR	i	Start write cycle
NOP			
NOP			

EXAMPLE 7-5: DATA EEPROM BLOCK WRITE

7.4 Write Verify

Depending on the application, good programming practice may dictate that the value written to the memory should be verified against the original value. This should be used in applications where excessive writes can stress bits near the specification limit.

7.5 Protection Against Spurious Write

There are conditions when the device may not want to write to the data EEPROM memory. To protect against spurious EEPROM writes, various mechanisms have been built-in. On power-up, the WREN bit is cleared; also, the Power-up Timer prevents EEPROM write.

The write initiate sequence, and the WREN bit together, help prevent an accidental write during brown-out, power glitch or software malfunction.

10.1 Timer Gate Operation

The 32-bit timer can be placed in the Gated Time Accumulation mode. This mode allows the internal TCY to increment the respective timer when the gate input signal (T2CK pin) is asserted high. Control bit, TGATE (T2CON<6>), must be set to enable this mode. When in this mode, Timer2 is the originating clock source. The TGATE setting is ignored for Timer3. The timer must be enabled (TON = 1) and the timer clock source set to internal (TCS = 0).

The falling edge of the external signal terminates the count operation, but does not reset the timer. The user must reset the timer in order to start counting from zero.

10.2 ADC Event Trigger

When a match occurs between the 32-bit timer (TMR3/ TMR2) and the 32-bit combined Period register (PR3/ PR2), a special ADC trigger event signal is generated by Timer3.

10.3 Timer Prescaler

The input clock (FOSC/4 or external clock) to the timer has a prescale option of 1:1, 1:8, 1:64 and 1:256, selected by control bits, TCKPS<1:0> (T2CON<5:4> and T3CON<5:4>). For the 32-bit timer operation, the originating clock source is Timer2. The prescaler operation for Timer3 is not applicable in this mode. The prescaler counter is cleared when any of the following occurs:

- A write to the TMR2/TMR3 register
- Clearing either of the TON (T2CON<15> or T3CON<15>) bits to '0'
- A device Reset such as a POR and BOR

However, if the timer is disabled (TON = 0), the Timer2 prescaler cannot be reset, since the prescaler clock is halted.

TMR2/TMR3 is not cleared when T2CON/T3CON is written.

10.4 Timer Operation During Sleep Mode

During CPU Sleep mode, the timer will not operate, because the internal clocks are disabled.

10.5 Timer Interrupt

The 32-bit timer module can generate an interrupt-onperiod match, or on the falling edge of the external gate signal. When the 32-bit timer count matches the respective 32-bit Period register, or the falling edge of the external "gate" signal is detected, the T3IF bit (IFS0<7>) is asserted and an interrupt will be generated if enabled. In this mode, the T3IF interrupt flag is used as the source of the interrupt. The T3IF bit must be cleared in software.

Enabling an interrupt is accomplished via the respective Timer Interrupt Enable bit, T3IE (IEC0<7>).

15.0 MOTOR CONTROL PWM MODULE

Note: This data sheet summarizes features of this group of dsPIC30F devices and is not intended to be a complete reference source. For more information on the CPU, peripherals, register descriptions and general device functionality, refer to the "dsPIC30F Family Reference Manual" (DS70046).

This module simplifies the task of generating multiple, synchronized Pulse-Width Modulated (PWM) outputs. In particular, the following power and motion control applications are supported by the PWM module:

- Three-Phase AC Induction Motor
- Switched Reluctance (SR) Motor
- Brushless DC (BLDC) Motor
- Uninterruptible Power Supply (UPS)

The PWM module has the following features:

- 6 PWM I/O pins with 3 duty cycle generators
- Up to 16-bit resolution

- 'On-the-Fly' PWM frequency changes
- Edge and Center-Aligned Output modes
- Single Pulse Generation mode
- Interrupt support for asymmetrical updates in Center-Aligned mode
- Output override control for Electrically Commutative Motor (ECM) operation
- 'Special Event' comparator for scheduling other peripheral events
- Fault pins to optionally drive each of the PWM output pins to a defined state

This module contains 3 duty cycle generators, numbered 1 through 3. The module has 6 PWM output pins, numbered PWM1H/PWM1L through PWM3H/ PWM3L. The six I/O pins are grouped into high/low numbered pairs, denoted by the suffix H or L, respectively. For complementary loads, the low PWM pins are always the complement of the corresponding high I/O pins.

The PWM module allows several modes of operation which are beneficial for specific power control applications.

FIGURE 15-4: DEAD-TIME TIMING DIAGRAM



15.8 Independent PWM Output

An Independent PWM Output mode is required for driving certain types of loads. A particular PWM output pair is in the Independent Output mode when the corresponding PMOD bit in the PWMCON1 register is set. No dead-time control is implemented between adjacent PWM I/O pins when the module is operating in the Independent mode and both I/O pins are allowed to be active simultaneously.

In the Independent mode, each duty cycle generator is connected to both of the PWM I/O pins in an output pair. By using the associated Duty Cycle register and the appropriate bits in the OVDCON register, the user may select the following signal output options for each PWM I/O pin operating in the Independent mode:

- I/O pin outputs PWM signal
- I/O pin inactive
- I/O pin active

15.9 Single Pulse PWM Operation

The PWM module produces single pulse outputs when the PTCON control bits, PTMOD<1:0> = 10. Only edge-aligned outputs may be produced in the Single Pulse mode. In Single Pulse mode, the PWM I/O pin(s) are driven to the active state when the PTEN bit is set. When a match with a Duty Cycle register occurs, the PWM I/O pin is driven to the inactive state. When a match with the PTPER register occurs, the PTMR register is cleared, all active PWM I/O pins are driven to the inactive state, the PTEN bit is cleared and an interrupt is generated.

15.10 PWM Output Override

The PWM output override bits allow the user to manually drive the PWM I/O pins to specified logic states, independent of the duty cycle comparison units.

All control bits associated with the PWM output override function are contained in the OVDCON register. The upper half of the OVDCON register contains six bits, POVDxH<3:1> and POVDxL<3:1>, that determine which PWM I/O pins will be overridden. The lower half of the OVDCON register contains six bits, POUTxH<3:1> and POUTxL<3:1>, that determine the state of the PWM I/O pins when a particular output is overridden via the POVD bits.

15.10.1 COMPLEMENTARY OUTPUT MODE

When a PWMxL pin is driven active via the OVDCON register, the output signal is forced to be the complement of the corresponding PWMxH pin in the pair. Dead-time insertion is still performed when PWM channels are overridden manually.

15.10.2 OVERRIDE SYNCHRONIZATION

If the OSYNC bit in the PWMCON2 register is set, all output overrides performed via the OVDCON register are synchronized to the PWM time base. Synchronous output overrides occur at the following times:

- Edge-Aligned mode, when PTMR is zero.
- Center-Aligned modes, when PTMR is zero and when the value of PTMR matches PTPER.

17.4.1 10-BIT MODE SLAVE TRANSMISSION

Once a slave is addressed in this fashion, with the full 10-bit address (we will refer to this state as "PRIOR_ADDR_MATCH"), the master can begin sending data bytes for a slave reception operation.

17.4.2 10-BIT MODE SLAVE RECEPTION

Once addressed, the master can generate a Repeated Start, reset the high byte of the address and set the R_W bit without generating a Stop bit, thus initiating a slave transmit operation.

17.5 Automatic Clock Stretch

In the Slave modes, the module can synchronize buffer reads and writes to the master device by clock stretching.

17.5.1 TRANSMIT CLOCK STRETCHING

Both 10-Bit and 7-Bit Transmit modes implement clock stretching by asserting the SCLREL bit after the falling edge of the ninth clock if the TBF bit is cleared, indicating the buffer is empty.

In Slave Transmit modes, clock stretching is always performed, irrespective of the STREN bit.

Clock synchronization takes place following the ninth clock of the transmit sequence. If the device samples an ACK on the falling edge of the ninth clock, and if the TBF bit is still clear, then the SCLREL bit is automatically cleared. The SCLREL being cleared to '0' will assert the SCL line low. The user's ISR must set the SCLREL bit before transmission is allowed to continue. By holding the SCL line low, the user has time to service the ISR and load the contents of the I2CTRN before the master device can initiate another transmit sequence.

- Note 1: If the user loads the contents of I2CTRN, setting the TBF bit before the falling edge of the ninth clock, the SCLREL bit will not be cleared and clock stretching will not occur.
 - **2:** The SCLREL bit can be set in software, regardless of the state of the TBF bit.

17.5.2 RECEIVE CLOCK STRETCHING

The STREN bit in the I2CCON register can be used to enable clock stretching in Slave Receive mode. When the STREN bit is set, the SCL pin will be held low at the end of each data receive sequence.

17.5.3 CLOCK STRETCHING DURING 7-BIT ADDRESSING (STREN = 1)

When the STREN bit is set in Slave Receive mode, the SCL line is held low when the buffer register is full. The method for stretching the SCL output is the same for both 7 and 10-Bit Addressing modes.

Clock stretching takes place following the ninth clock of the receive sequence. <u>On</u> the falling edge of the ninth clock at the end of the ACK sequence, if the RBF bit is set, the SCLREL bit is automatically cleared, forcing the SCL output to be held low. The user's ISR must set the SCLREL bit before reception is allowed to continue. By holding the SCL line low, the user has time to service the ISR and read the contents of the I2CRCV before the master device can initiate another receive sequence. This will prevent buffer overruns from occurring.

Note 1:	If the user reads the contents of the
	I2CRCV, clearing the RBF bit before the
	falling edge of the ninth clock, the
	SCLREL bit will not be cleared and clock
	stretching will not occur.

2: The SCLREL bit can be set in software, regardless of the state of the RBF bit. The user should be careful to clear the RBF bit in the ISR before the next receive sequence in order to prevent an overflow condition.

17.5.4 CLOCK STRETCHING DURING 10-BIT ADDRESSING (STREN = 1)

Clock stretching takes place automatically during the addressing sequence. Because this module has a register for the entire address, it is not necessary for the protocol to wait for the address to be updated.

After the address phase is complete, clock stretching will occur on each data receive or transmit sequence as was described earlier.

17.6 Software Controlled Clock Stretching (STREN = 1)

When the STREN bit is '1', the SCLREL bit may be cleared by software to allow software to control the clock stretching. The logic will synchronize writes to the SCLREL bit with the SCL clock. Clearing the SCLREL bit will not assert the SCL output until the module detects a falling edge on the SCL output and SCL is sampled low. If the SCLREL bit is cleared by the user while the SCL line has been sampled low, the SCL output will be asserted (held low). The SCL output will remain low until the SCLREL bit is set, and all other devices on the I²C bus have deasserted SCL. This ensures that a write to the SCLREL bit will not violate the minimum high time requirement for SCL.

If the STREN bit is '0', a software write to the SCLREL bit will be disregarded and have no effect on the SCLREL bit.

17.7 Interrupts

The I²C module generates two interrupt flags, MI2CIF (I²C Master Interrupt Flag) and SI2CIF (I²C Slave Interrupt Flag). The MI2CIF interrupt flag is activated on completion of a master message event. The SI2CIF interrupt flag is activated on detection of a message directed to the slave.

17.8 Slope Control

The I²C standard requires slope control on the SDA and SCL signals for Fast mode (400 kHz). The control bit, DISSLW, enables the user to disable slew rate control, if desired. It is necessary to disable the slew rate control for 1 MHz mode.

17.9 IPMI Support

The control bit, IPMIEN, enables the module to support Intelligent Peripheral Management Interface (IPMI). When this bit is set, the module accepts and acts upon all addresses.

17.10 General Call Address Support

The general call address can address all devices. When this address is used, all devices should, in theory, respond with an Acknowledgement.

The general call address is one of eight addresses reserved for specific purposes by the I^2C protocol. It consists of all '0's with R_W = 0.

The general call address is recognized when the General Call Enable (GCEN) bit is set (I2CCON<7> = 1). Following a Start bit detection, 8 bits are shifted into I2CRSR and the address is compared with I2CADD, and is also compared with the general call address which is fixed in hardware.

If a general call address match occurs, the I2CRSR is transferred to the I2CRCV after the eighth clock, the RBF flag is set, and on the falling edge of the ninth bit (ACK bit), the Master Event Interrupt Flag (MI2CIF) is set.

When the interrupt is serviced, the source for the interrupt can be checked by reading the contents of the I2CRCV to determine if the address was device-specific, or a general call address.

17.11 I²C Master Support

As a master device, six operations are supported:

- Assert a Start condition on SDA and SCL
- Assert a Restart condition on SDA and SCL
- Write to the I2CTRN register initiating transmission of data/address
- Generate a Stop condition on SDA and SCL
- Configure the I²C port to receive data
- Generate an ACK condition at the end of a received byte of data

17.12 I²C Master Operation

The master device generates all of the serial clock pulses and the Start and Stop conditions. A transfer is ended with a Stop condition or with a Repeated Start condition. Since the Repeated Start condition is also the beginning of the next serial transfer, the I²C bus will not be released.

In Master Transmitter mode, serial data is output through SDA, while SCL outputs the serial clock. The first byte transmitted contains the slave address of the receiving device (7 bits) and the data direction bit. In this case, the data direction bit (R_W) is logic '0'. Serial data is transmitted 8 bits at a time. After each byte is transmitted, an ACK bit is received. Start and Stop conditions are output to indicate the beginning and the end of a serial transfer.

In Master Receive mode, the first byte transmitted contains the slave address of the transmitting device (7 bits) and the data direction bit. In this case, the data direction bit (R_W) is logic '1'. Thus, the first byte transmitted is a 7-bit slave address, followed by a '1' to indicate the receive bit. Serial data is received via SDA, while SCL outputs the serial clock. Serial data is received 8 bits at a time. After each byte is received, an ACK bit is transmitted. Start and Stop conditions indicate the beginning and end of transmission.

17.12.1 I²C MASTER TRANSMISSION

Transmission of a data byte, a 7-bit address or the second half of a 10-bit address is accomplished by simply writing a value to I2CTRN register. The user should only write to I2CTRN when the module is in a Wait state. This action will set the Buffer Full Flag (TBF) and allow the Baud Rate Generator to begin counting and start the next transmission. Each bit of address/ data will be shifted out onto the SDA pin after the falling edge of SCL is asserted. The Transmit Status Flag, TRSTAT (I2CSTAT<14>), indicates that a master transmit is in progress.

17.12.2 I²C MASTER RECEPTION

Master mode reception is enabled by programming the Receive Enable (RCEN) bit (I2CCON<3>). The I²C module must be Idle before the RCEN bit is set; otherwise, the RCEN bit will be disregarded. The Baud Rate Generator begins counting, and on each rollover, the state of the SCL pin toggles, and data is shifted into the I2CRSR on the rising edge of each clock.

17.12.3 BAUD RATE GENERATOR (BRG)

In I²C Master mode, the reload value for the BRG is located in the I2CBRG register. When the BRG is loaded with this value, the BRG counts down to '0' and stops until another reload has taken place. If clock arbitration is taking place, for instance, the BRG is reloaded when the SCL pin is sampled high.

As per the I²C standard, FSCL may be 100 kHz or 400 kHz. However, the user can specify any baud rate up to 1 MHz. I2CBRG values of '0' or '1' are illegal.

EQUATION 17-1: I2CBRG VALUE

$$I2CBRG = \left(\frac{FCY}{FSCL} - \frac{FCY}{1,111,111}\right) - 1$$

17.12.4 CLOCK ARBITRATION

Clock arbitration occurs when the master deasserts the SCL pin (SCL allowed to float high) during any receive, transmit or Restart/Stop condition. When the SCL pin is allowed to float high, the Baud Rate Generator is suspended from counting until the SCL pin is actually sampled high. When the SCL pin is sampled high, the Baud Rate Generator is reloaded with the contents of I2CBRG and begins counting. This ensures that the SCL high time will always be at least one BRG rollover count in the event that the clock is held low by an external device.

17.12.5 MULTI-MASTER COMMUNICATION, BUS COLLISION AND BUS ARBITRATION

Multi-master operation support is achieved by bus arbitration. When the master outputs address/data bits onto the SDA pin, arbitration takes place when the master outputs a '1' on SDA, by letting SDA float high while another master asserts a '0'. When the SCL pin floats high, data should be stable. If the expected data on SDA is a '1' and the data sampled on the SDA pin = 0, then a bus collision has taken place. The master will set the MI2CIF pulse and reset the master portion of the I²C port to its Idle state.

If a transmit was in progress when the bus collision occurred, the transmission is halted, the TBF flag is cleared, the SDA and SCL lines are deasserted and a value can now be written to I2CTRN. When the user services the I^2C master event Interrupt Service Routine, if the I^2C bus is free (i.e., the P bit is set), the user can resume communication by asserting a Start condition.

If a Start, Restart, Stop or Acknowledge condition was in progress when the bus collision occurred, the condition is aborted, the SDA and SCL lines are deasserted and the respective control bits in the I2CCON register are cleared to '0'. When the user services the bus collision Interrupt Service Routine, and if the I²C bus is free, the user can resume communication by asserting a Start condition.

The Master will continue to monitor the SDA and SCL pins, and if a Stop condition occurs, the MI2CIF bit will be set.

A write to the I2CTRN will start the transmission of data at the first data bit, regardless of where the transmitter left off when bus collision occurred.

In a Multi-Master environment, the interrupt generation on the detection of Start and Stop conditions allows the determination of when the bus is free. Control of the I^2C bus can be taken when the P bit is set in the I2CSTAT register, or the bus is Idle and the S and P bits are cleared.

17.13 I²C Module Operation During CPU Sleep and Idle Modes

17.13.1 I²C OPERATION DURING CPU SLEEP MODE

When the device enters Sleep mode, all clock sources to the module are shut down and stay at logic '0'. If Sleep occurs in the middle of a transmission, and the state machine is partially into a transmission as the clocks stop, then the transmission is aborted. Similarly, if Sleep occurs in the middle of a reception, then the reception is aborted.

17.13.2 I²C OPERATION DURING CPU IDLE MODE

For the I^2C , the I2CSIDL bit selects if the module will stop on Idle or continue on Idle. If I2CSIDL = 0, the module will continue operation on assertion of the Idle mode. If I2CSIDL = 1, the module will stop on Idle.

20.0 SYSTEM INTEGRATION

Note: This data sheet summarizes features of this group of dsPIC30F devices and is not intended to be a complete reference source. For more information on the CPU, peripherals, register descriptions and general device functionality, refer to the "dsPIC30F Family Reference Manual" (DS70046). For more information on the device instruction set and programming, refer to the "16-bit MCU and DSC Programmer's Reference Manual" (DS70157).

There are several features intended to maximize system reliability, minimize cost through elimination of external components, provide power-saving operating modes and offer code protection:

- Oscillator Selection
- Reset
 - Power-on Reset (POR)
 - Power-up Timer (PWRT)
 - Oscillator Start-up Timer (OST)
 - Programmable Brown-out Reset (BOR)
- Watchdog Timer (WDT)
- Power-Saving modes (Sleep and Idle)
- Code Protection
- Unit ID Locations
- In-Circuit Serial Programming (ICSP)

dsPIC30F devices have a Watchdog Timer, which is permanently enabled via the Configuration bits, or can be software controlled. It runs off its own RC oscillator for added reliability. There are two timers that offer necessary delays on power-up. One is the Oscillator Startup Timer (OST), intended to keep the chip in Reset until the crystal oscillator is stable. The other is the Powerup Timer (PWRT), which provides a delay on power-up only, designed to keep the part in Reset while the power supply stabilizes. With these two timers on-chip, most applications need no external Reset circuitry.

Sleep mode is designed to offer a very low-current Power-Down mode. The user can wake-up from Sleep through external Reset, Watchdog Timer wake-up or through an interrupt. Several oscillator options are also made available to allow the part to fit a wide variety of applications. In the Idle mode, the clock sources are still active, but the CPU is shut off. The RC oscillator option saves system cost, while the LP crystal option saves power.

20.1 Oscillator System Overview

The dsPIC30F oscillator system has the following modules and features:

- Various external and internal oscillator options as clock sources
- An on-chip PLL to boost internal operating frequency
- A clock switching mechanism between various clock sources
- Programmable clock postscaler for system power savings
- A Fail-Safe Clock Monitor (FSCM) that detects clock failure and takes fail-safe measures
- Clock Control register, OSCCON
- · Configuration bits for main oscillator selection

Configuration bits determine the clock source upon Power-on Reset and Brown-out Reset. Thereafter, the clock source can be changed between permissible clock sources. The OSCCON register controls the clock switching and reflects system clock related status bits.

Table 20-1 provides a summary of the dsPIC30F oscillator operating modes. A simplified diagram of the oscillator system is shown in Figure 20-1.

dsPIC30F3010/3011

FIGURE 23-8: TIMERQ (QEI MODULE) EXTERNAL CLOCK TIMING CHARACTERISTICS



TABLE 23-25: QEI MODULE EXTERNAL CLOCK TIMING REQUIREMENTS

AC CHARACTERISTICS

Standard Operating Conditions: 2.5V to 5.5V(unless otherwise stated)Operating temperature $-40^{\circ}C \le TA \le +85^{\circ}C$ for Industrial

-40°C \leq TA \leq +125°C for Extended

Param No.	Symbol	Character	Min	Тур	Max	Units	Conditions	
TQ10	TtQH	TQCK High Time	Synchronous, with prescaler	TCY + 20	_	_	ns	Must also meet parameter TQ15
TQ11	TtQL	TQCK Low Time	Synchronous, with prescaler	Tcy + 20	—	—	ns	Must also meet parameter TQ15
TQ15	TtQP	TQCP Input Period	Synchronous, with prescaler	2 * TCY + 40	—	—	ns	
TQ20	TCKEXTMRL	Delay from External TQCK Clock Edge to Timer Increment		0.5 TCY	—	1.5 TCY	—	

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

AC CHARACTERISTICS			$\begin{array}{l} \mbox{Standard Operating Conditions: 2.5V to 5.5V} \\ \mbox{(unless otherwise stated)} \\ \mbox{Operating temperature} & -40^\circ C \leq TA \leq +85^\circ C \mbox{ for Industria} \\ & -40^\circ C \leq TA \leq +125^\circ C \mbox{ for Extended} \\ \end{array}$				
Param No.	Symbol	Characteristic ⁽¹⁾	Min Typ ⁽²⁾ Max Units				Conditions
SP70	TscL	SCKx Input Low Time	30	—	_	ns	
SP71	TscH	SCKx Input High Time	30	—	_	ns	
SP72	TscF	SCKx Input Fall Time ⁽³⁾	—	10	25	ns	
SP73	TscR	SCKx Input Rise Time ⁽³⁾	—	10	25	ns	
SP30	TdoF	SDOx Data Output Fall Time ⁽³⁾	—			ns	See parameter DO32
SP31	TdoR	SDOx Data Output Rise Time ⁽³⁾	—	_	_	ns	See parameter DO31
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	_	30	ns	
SP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	20	_		ns	
SP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	20	_	_	ns	
SP50	TssL2scH, TssL2scL	SSx↓ to SCKx↑ or SCKx↓ Input	120	—	_	ns	
SP51	TssH2doZ	SSx↑ to SDOx Output High-Impedance ⁽³⁾	10	—	50	ns	
SP52	TscH2ssH TscL2ssH	SSx after SCKx Edge	1.5 Tcy + 40	—	_	ns	—

TABLE 23-34: SPI MODULE SLAVE MODE (CKE = 0) TIMING REQUIREMENTS

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

2: Data in "Typ" column is at 5V, 25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

3: Assumes 50 pF load on all SPI pins.



FIGURE 23-19: SPI MODULE SLAVE MODE (CKE = 1) TIMING CHARACTERISTICS

AC CHARACTERISTICS			$\begin{array}{l} \mbox{Standard Operating Conditions: 2.5V to 5.5V} \\ \mbox{(unless otherwise stated)} \\ \mbox{Operating temperature} & -40^{\circ}C \leq TA \leq +85^{\circ}C \mbox{ for Industrial} \\ & -40^{\circ}C \leq TA \leq +125^{\circ}C \mbox{ for Extended} \end{array}$				
Param No.	Symbol	Characteristic ⁽¹⁾	Min	Тур ⁽²⁾	Max	Units	Conditions
SP70	TscL	SCKx Input Low Time	30			ns	
SP71	TscH	SCKx Input High Time	30	_	_	ns	
SP72	TscF	SCKx Input Fall Time ⁽³⁾	—	10	25	ns	
SP73	TscR	SCKx Input Rise Time ⁽³⁾	—	10	25	ns	
SP30	TdoF	SDOx Data Output Fall Time ⁽³⁾	_			ns	See parameter DO32
SP31	TdoR	SDOx Data Output Rise Time ⁽³⁾	_	_		ns	See parameter DO31
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	_	—	30	ns	
SP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	20	_	_	ns	
SP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	20	_	_	ns	
SP50	TssL2scH, TssL2scL	$\overline{SSx}\downarrow$ to SCKx \downarrow or SCKx \uparrow Input	120	—	—	ns	
SP51	TssH2doZ	SSx [↑] to SDOx Output High-Impedance ⁽⁴⁾	10	—	50	ns	
SP52	TscH2ssH TscL2ssH	SSx [↑] after SCKx Edge	1.5 Tcy + 40	—	—	ns	
SP60	TssL2doV	SDOx Data Output Valid after	_	_	50	ns	

TABLE 23-35: SPI MODULE SLAVE MODE (CKE = 1) TIMING REQUIREMENTS

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

2: Data in "Typ" column is at 5V, 25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

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

4: Assumes 50 pF load on all SPI pins.

TABLE 23-36: I²C[™] BUS DATA TIMING REQUIREMENTS (MASTER MODE)

AC CHARACTERISTICS				$\begin{array}{l} \mbox{Standard Operating Conditions: 2.5V to 5.5V} \\ \mbox{(unless otherwise stated)} \\ \mbox{Operating temperature} & -40^{\circ}C \leq TA \leq +85^{\circ}C \mbox{ for Industrial} \\ & -40^{\circ}C \leq TA \leq +125^{\circ}C \mbox{ for Extended} \end{array}$				
Param No.	Symbol	Characteristic		Min ⁽¹⁾	Max	Units	Conditions	
IM10	TLO:SCL	Clock Low Time	100 kHz mode	Tcy/2 (BRG + 1)	_	μS		
			400 kHz mode	Tcy/2 (BRG + 1)		μS		
			1 MHz mode ⁽²⁾	Tcy/2 (BRG + 1)	_	μS		
IM11	THI:SCL	Clock High Time	100 kHz mode	Tcy/2 (BRG + 1)	—	μS		
			400 kHz mode	Tcy/2 (BRG + 1)	—	μs		
			1 MHz mode ⁽²⁾	Tcy/2 (BRG + 1)		μS		
IM20	TF:SCL	SDA and SCL	100 kHz mode		300	ns	CB is specified to be	
		Fall Time	400 kHz mode	20 + 0.1 Св	300	ns	from 10 to 400 pF	
			1 MHz mode ⁽²⁾	—	100	ns		
IM21	TR:SCL	SDA and SCL	100 kHz mode	_	1000	ns	CB is specified to be	
		Rise Time	400 kHz mode	20 + 0.1 Св	300	ns	from 10 to 400 pF	
			1 MHz mode ⁽²⁾		300	ns		
IM25	TSU:DAT	Data Input Setup Time	100 kHz mode	250	_	ns		
			400 kHz mode	100	_	ns		
			1 MHz mode ⁽²⁾		—	ns		
IM26	THD:DAT	Data Input Hold Time	100 kHz mode	0	_	ns		
			400 kHz mode	0	0.9	μS		
			1 MHz mode ⁽²⁾			ns		
IM30	TSU:STA	Start Condition	100 kHz mode	Tcy/2 (BRG + 1)	_	μS	Only relevant for	
		Setup Time	400 kHz mode	Tcy/2 (BRG + 1)	—	μS	Repeated Start	
			1 MHz mode ⁽²⁾	Tcy/2 (BRG + 1)	—	μS	condition	
IM31	THD:STA	Start Condition	100 kHz mode	Tcy/2 (BRG + 1)	_	μS	After this period the	
		Hold Time	400 kHz mode	Tcy/2 (BRG + 1)	—	μS	first clock pulse is	
			1 MHz mode ⁽²⁾	Tcy/2 (BRG + 1)	_	μS	generated	
IM33	TSU:STO	Stop Condition	100 kHz mode	Tcy/2 (BRG + 1)		μS		
		Setup Time	400 kHz mode	Tcy/2 (BRG + 1)		μS		
			1 MHz mode ⁽²⁾	Tcy/2 (BRG + 1)		μS		
IM34	THD:STO	Stop Condition	100 kHz mode	Tcy/2 (BRG + 1)		ns		
		Hold Time	400 kHz mode	Tcy/2 (BRG + 1)		ns		
			1 MHz mode ⁽²⁾	Tcy/2 (BRG + 1)		ns		
IM40	TAA:SCL	Output Valid	100 kHz mode	_	3500	ns		
		From Clock	400 kHz mode	—	1000	ns		
			1 MHz mode ⁽²⁾		_	ns		
IM45	TBF:SDA	Bus Free Time	100 kHz mode	4.7	—	μs	Time the bus must be	
			400 kHz mode	1.3	—	μS	free before a new	
			1 MHz mode ⁽²⁾	—	—	μS	transmission can start	
IM50	Св	Bus Capacitive L	oading	—	400	pF		

Note 1: BRG is the value of the I²C[™] Baud Rate Generator. Refer to Section 21. "Inter-Integrated Circuit (I²C)" in the "dsPIC30F Family Reference Manual" (DS70046). 2: Maximum pin capacitance = 10 pF for all I^2C pins (for 1 MHz mode only).

dsPIC30F3010/3011



FIGURE 23-24: 10-BIT HIGH-SPEED ADC TIMING CHARACTERISTICS

44-Lead Plastic Quad Flat, No Lead Package (ML) – 8x8 mm Body [QFN]

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



	MILLIMETERS			
Dimension	MIN	NOM	MAX	
Contact Pitch	0.65 BSC			
Optional Center Pad Width	W2			6.80
Optional Center Pad Length	T2			6.80
Contact Pad Spacing	C1		8.00	
Contact Pad Spacing	C2		8.00	
Contact Pad Width (X44)				0.35
Contact Pad Length (X44)				0.80
Distance Between Pads	G	0.25		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2103A