

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	30 MIPs
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, Motor Control PWM, QEI, POR, PWM, WDT
Number of I/O	20
Program Memory Size	12KB (4K x 24)
Program Memory Type	FLASH
EEPROM Size	1K × 8
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	2.5V ~ 5.5V
Data Converters	A/D 6x10b
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/dspic30f2010-30i-so

Email: info@E-XFL.COM

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

dsPIC30F2010

High-Performance, 16-bit Digital Signal Controller

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
- 83 base instructions with flexible addressing modes
- 24-bit wide instructions, 16-bit wide data path
- 12 Kbytes on-chip Flash program space
- 512 bytes on-chip data RAM
- 1 Kbyte 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)
- 27 interrupt sources
- Three external interrupt sources
- · Eight user-selectable priority levels for each interrupt
- · Four processor exceptions and software traps

DSP Engine Features:

- · Modulo and Bit-Reversed modes
- Two 40-bit wide accumulators with optional saturation logic
- 17-bit x 17-bit single-cycle hardware fractional/ integer multiplier
- Single-cycle Multiply-Accumulate (MAC) operation
- 40-stage Barrel Shifter
- Dual data fetch

Peripheral Features:

- High current sink/source I/O pins: 25 mA/25 mA
- Three 16-bit timers/counters; optionally pair up 16-bit timers into 32-bit timer modules
- Four 16-bit capture input functions
- Two 16-bit compare/PWM output functions
 Dual Compare mode available
- 3-wire SPI modules (supports 4 Frame modes)
- I²C[™] module supports Multi-Master/Slave mode
- and 7-bit/10-bit addressing
- Addressable UART modules with FIFO buffers

Motor Control PWM Module Features:

- Six PWM output channels
 - Complementary or Independent Output modes
 - Edge and Center-Aligned modes
- Four duty cycle generators
- · Dedicated time base with four modes
- · Programmable output polarity
- · Dead-time control for Complementary mode
- Manual output control
- Trigger for synchronized 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:
 - 1 Msps (for 10-bit A/D) conversion rate
 - Six input channels
 - Conversion available during Sleep and Idle
- Programmable Brown-out Reset

TABLE 3-3: CORE REGISTER MAP (CONTINUED)

SFR Name	Address (Home)	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	Reset State		
CORCON	0044	—		—	US	EDT	DL2	DL1	DL0	SATA	SATB	SATDW	ACCSAT	IPL3	PSV	RND	IF	0000 0000 0010 0000		
MODCON	0046	XMODEN	YMODEN	_	_		BWN	l<3:0>			YW	M<3:0>			XWM	<3:0>		0000 0000 0000 0000		
XMODSRT	0048							X	S<15:1>								0	uuuu uuuu uuuu uuu0		
XMODEND	004A							Х	E<15:1>								1	uuuu uuuu uuul		
YMODSRT	004C							Y	S<15:1>								0	uuuu uuuu uuuu uuu0		
YMODEND	004E		YE<15:1> 1 u								uuuu uuuu uuul									
XBREV	0050	BREN		XB<14:0> uu									uuuu uuuu uuuu							
DISICNT	0052	_	_	— DISICNT<13:0> 00									0000 0000 0000 0000							

Legend: u = uninitialized bit; — = unimplemented bit, read as '0'

Note: Refer to the "dsPIC30F Family Reference Manual" (DS70046) for descriptions of register bit fields.

4.0 ADDRESS GENERATOR UNITS

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).

The dsPIC DSC core contains two independent address generator units: the X AGU and Y AGU. The Y AGU supports word-sized data reads for the DSP MAC class of instructions only. The dsPIC DSC AGUs support three types of data addressing:

- Linear Addressing
- Modulo (Circular) Addressing
- Bit-Reversed Addressing

Linear and Modulo Data Addressing modes can be applied to data space or program space. Bit-Reversed Addressing is only applicable to data space addresses.

4.1 Instruction Addressing Modes

The Addressing modes in Table 4-1 form the basis of the Addressing modes optimized to support the specific features of individual instructions. The Addressing modes provided in the MAC class of instructions are somewhat different from those in the other instruction types.

4.1.1 FILE REGISTER INSTRUCTIONS

Most file register instructions use a 13-bit address field (f) to directly address data present in the first 8192 bytes of data memory (near data space). Most file register instructions employ a working register W0, which is denoted as WREG in these instructions. The destination is typically either the same file register, or WREG (with the exception of the MUL instruction), which writes the result to a register or register pair. The MOV instruction allows additional flexibility and can access the entire data space.

Addressing Mode	Description
File Register Direct	The address of the file register is specified explicitly.
Register Direct	The contents of a register are accessed directly.
Register Indirect	The contents of Wn forms the Effective Address (EA).
Register Indirect Post-modified	The contents of Wn forms the EA. Wn is post-modified (incremented or decremented) by a constant value.
Register Indirect Pre-modified	Wn is pre-modified (incremented or decremented) by a signed constant value to form the EA.
Register Indirect with Register Offset	The sum of Wn and Wb forms the EA.
Register Indirect with Literal Offset	The sum of Wn and a literal forms the EA.

TABLE 4-1: FUNDAMENTAL ADDRESSING MODES SUPPORTED

5.0 INTERRUPTS

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).

The dsPIC30F2010 has 24 interrupt sources and four processor exceptions (traps), which must be arbitrated based on a priority scheme.

The CPU is responsible for reading the Interrupt Vector Table (IVT) and transferring the address contained in the interrupt vector to the program counter. The interrupt vector is transferred from the program data bus into the program counter, via a 24-bit wide multiplexer on the input of the program counter.

The Interrupt Vector Table (IVT) and Alternate Interrupt Vector Table (AIVT) are placed near the beginning of program memory (0x000004). The IVT and AIVT are shown in Figure 5-1.

The interrupt controller is responsible for preprocessing the interrupts and processor exceptions, prior to their being presented to the processor core. The peripheral interrupts and traps are enabled, prioritized and controlled using centralized special function registers:

- IFS0<15:0>, IFS1<15:0>, IFS2<15:0> All interrupt request flags are maintained in these three registers. The flags are set by their respective peripherals or external signals, and they are cleared via software.
- IEC0<15:0>, IEC1<15:0>, IEC2<15:0>
 All interrupt enable control bits are maintained in these three registers. These control bits are used to individually enable interrupts from the peripherals or external signals.
- IPC0<15:0>... IPC11<7:0> The user-assignable priority level associated with each of these interrupts is held centrally in these twelve registers.
- IPL<3:0> The current CPU priority level is explicitly stored in the IPL bits. IPL<3> is present in the CORCON register, whereas IPL<2:0> are present in the STATUS Register (SR) in the processor core.
- INTCON1<15:0>, INTCON2<15:0> Global interrupt control functions are derived from these two registers. INTCON1 contains the control and status flags for the processor exceptions. The INTCON2 register controls the external interrupt request signal behavior and the use of the alternate vector table.

Note: Interrupt flag bits get set when an interrupt condition occurs, regardless of the state of its corresponding enable bit. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

All interrupt sources can be user-assigned to one of seven priority levels, 1 through 7, via the IPCx registers. Each interrupt source is associated with an interrupt vector, as shown in Figure 5-1. Levels 7 and 1 represent the highest and lowest maskable priorities, respectively.

Note:	Assigning a priority level of 0 to an inter-
	rupt source is equivalent to disabling that
	interrupt.

If the NSTDIS bit (INTCON1<15>) is set, nesting of interrupts is prevented. Thus, if an interrupt is currently being serviced, processing of a new interrupt is prevented, even if the new interrupt is of higher priority than the one currently being serviced.

Note: The IPL bits become read-only whenever the NSTDIS bit has been set to '1'.

Certain interrupts have specialized control bits for features like edge or level triggered interrupts, interrupt-on-change, etc. Control of these features remains within the peripheral module which generates the interrupt.

The DISI instruction can be used to disable the processing of interrupts of priorities 6 and lower for a certain number of instructions, during which the DISI bit (INTCON2<14>) remains set.

When an interrupt is serviced, the PC is loaded with the address stored in the vector location in Program Memory that corresponds to the interrupt. There are 63 different vectors within the IVT (refer to Figure 5-1). These vectors are contained in locations 0x000004 through 0x0000FE of program memory (refer to Figure 5-1). These locations contain 24-bit addresses, and in order to preserve robustness, an address error trap will take place should the PC attempt to fetch any of these words during normal execution. This prevents execution of random data as a result of accidentally decrementing a PC into vector space, accidentally mapping a data space address into vector space or the PC rolling over to 0x000000 after reaching the end of implemented program memory space. Execution of a GOTO instruction to this vector space will also generate an address error trap.

7.3 Writing to the Data EEPROM

To write an EEPROM data location, the following sequence must be followed:

- 1. Erase data EEPROM word.
 - a) Select word, data EEPROM, erase and set WREN bit in NVMCON register.
 - b) Write address of word to be erased into NVMADRU/NVMADR.
 - c) Enable NVM interrupt (optional).
 - d) Write 0x55 to NVMKEY.
 - e) Write 0xAA to NVMKEY.
 - f) Set the WR bit. This will begin erase cycle.
 - g) Either poll NVMIF bit or wait for NVMIF interrupt.
 - h) The WR bit is cleared when the erase cycle ends.
- 2. Write data word into data EEPROM write latches.
- 3. Program 1 data word into data EEPROM.
 - a) Select word, data EEPROM, program and set WREN bit in NVMCON register.
 - b) Enable NVM write done interrupt (optional).
 - c) Write 0x55 to NVMKEY.
 - d) Write 0xAA to NVMKEY.
 - e) Set The WR bit. This will begin program cycle.
 - f) Either poll NVMIF bit or wait for NVM interrupt.
 - g) The WR bit is cleared when the write cycle ends.

EXAMPLE 7-4: DATA EEPROM WORD WRITE

;	Point to data me	emory	
	MOV	#LOW_ADDR_WORD,W0	; Init pointer
	MOV	#HIGH_ADDR_WORD,W1	
	MOV	W1 TBLPAG	
	MOV	#LOW(WORD),W2	; Get data
	TBLWTL	W2 [W0]	; Write data
;	The NVMADR captu	res last table access address	
;	Select data EEPR	COM for 1 word op	
	MOV	#0x4004,W0	
	MOV	W0 NVMCON	
		,	
;	Operate key to a	llow write operation	
	DISI #5	; Block al	l interrupts with priority <7
		; for next	5 instructions
	MOV	#0x55,W0	
	MOV	WONVMKEY	; Write the 0x55 key
	MOV	#0xAA,W1	
	MOV	W1 NVMKEY	; Write the OxAA key
	BSET	NVMCON, #WR	; Initiate program sequence
	NOP		
	NOP		
;	Write cycle will	complete in 2mS. CPU is not stal.	led for the Data Write Cycle
;	User can poll WR	bit, use NVMIF or Timer IRQ to de	etermine write complete

The write will not initiate if the above sequence is not exactly followed (write 0x55 to NVMKEY, write 0xAA to NVMCON, then set WR bit) for each word. It is strongly recommended that interrupts be disabled during this code segment.

Additionally, the WREN bit in NVMCON must be set to enable writes. This mechanism prevents accidental writes to data EEPROM, due to unexpected code execution. The WREN bit should be kept clear at all times, except when updating the EEPROM. The WREN bit is not cleared by hardware.

After a write sequence has been initiated, clearing the WREN bit will not affect the current write cycle. The WR bit will be inhibited from being set unless the WREN bit is set. The WREN bit must be set on a previous instruction. Both WR and WREN cannot be set with the same instruction.

At the completion of the write cycle, the WR bit is cleared in hardware and the Nonvolatile Memory Write Complete Interrupt Flag bit (NVMIF) is set. The user may either enable this interrupt, or poll this bit. NVMIF must be cleared by software.

7.3.1 WRITING A WORD OF DATA EEPROM

Once the user has erased the word to be programmed, then a table write instruction is used to write one write latch, as shown in Example 7-4.

10.0 TIMER2/3 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 section describes the 32-bit general purpose Timer module (Timer2/3) and associated operational modes. Figure 10-1 depicts the simplified block diagram of the 32-bit Timer2/3 module. Figure 10-2 and Figure 10-3 show Timer2/3 configured as two independent 16-bit timers; Timer2 and Timer3, respectively.

Note: Timer2 is a 'Type B' timer and Timer3 is a 'Type C' timer. Please refer to the appropriate timer type in Section 22.0 "Electrical Characteristics" for details.

The Timer2/3 module is a 32-bit timer, which can be configured as two 16-bit timers, with selectable operating modes. These timers are utilized by other peripheral modules such as:

- Input Capture
- Output Compare/Simple PWM

The following sections provide a detailed description, including setup and control registers, along with associated block diagrams for the operational modes of the timers.

The 32-bit timer has the following modes:

- Two independent 16-bit timers (Timer2 and Timer3) with all 16-bit operating modes (except Asynchronous Counter mode)
- Single 32-bit Timer operation
- Single 32-bit Synchronous Counter

Further, the following operational characteristics are supported:

- ADC Event Trigger
- Timer Gate Operation
- Selectable Prescaler Settings
- Timer Operation during Idle and Sleep modes
- Interrupt on a 32-bit Period Register Match

These operating modes are determined by setting the appropriate bit(s) in the 16-bit T2CON and T3CON SFRs.

For 32-bit timer/counter operation, Timer2 is the least significant word and Timer3 is the most significant word of the 32-bit timer.

Note: For 32-bit timer operation, T3CON control bits are ignored. Only T2CON control bits are used for setup and control. Timer 2 clock and gate inputs are utilized for the 32-bit timer module, but an interrupt is generated with the Timer3 interrupt flag (T3IF), and the interrupt is enabled with the Timer3 interrupt enable bit (T3IE).

16-bit Mode: In the 16-bit mode, Timer2 and Timer3 can be configured as two independent 16-bit timers. Each timer can be set up in either 16-bit Timer mode or 16-bit Synchronous Counter mode. See **Section 9.0 "Timer1 Module"** for details on these two operating modes.

The only functional difference between Timer2 and Timer3 is that Timer2 provides synchronization of the clock prescaler output. This is useful for high frequency external clock inputs.

32-bit Timer Mode: In the 32-bit Timer mode, the timer increments on every instruction cycle up to a match value, preloaded into the combined 32-bit period register PR3/PR2, then resets to '0' and continues to count.

For synchronous 32-bit reads of the Timer2/Timer3 pair, reading the least significant word (TMR2 register) will cause the most significant word (msw) to be read and latched into a 16-bit holding register, termed TMR3HLD.

For synchronous 32-bit writes, the holding register (TMR3HLD) must first be written to. When followed by a write to the TMR2 register, the contents of TMR3HLD will be transferred and latched into the MSB of the 32-bit timer (TMR3).

32-bit Synchronous Counter Mode: In the 32-bit Synchronous Counter mode, the timer increments on the rising edge of the applied external clock signal, which is synchronized with the internal phase clocks. The timer counts up to a match value preloaded in the combined 32-bit period register PR3/PR2, then resets to '0' and continues.

When the timer is configured for the Synchronous Counter mode of operation and the CPU goes into the Idle mode, the timer will stop incrementing, unless the TSIDL (T2CON<13>) bit = '0'. If TSIDL = '1', the timer module logic will resume the incrementing sequence upon termination of the CPU Idle mode.

TABLE 12-1: OUTPUT COMPARE REGISTER MAP

SFR 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	Reset State
OC1RS	0180		Output Compare 1 Master Register								0000 0000 0000 0000							
OC1R	0182		Output Compare 1 Slave Register										0000 0000 0000 0000					
OC1CON	0184	_	OCFRZ	OCSIDL	—	_	—	_	_	_	_		OCFLT1	OCTSEL1		OCM<2:0	>	0000 0000 0000 0000
OC2RS	0186		-					Out	put Comp	are 2 Mas	ter Registe	ər	-	-				0000 0000 0000 0000
OC2R	0188		Output Compare 2 Slave Register								0000 0000 0000 0000							
OC2CON	018A	_	OCFRZ	OCSIDL	_	_	_	_	_	_	_	_	OCFLT2	OCTSEL2		OCM<2:0	>	0000 0000 0000 0000

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

Note: Refer to the "dsPIC30F Family Reference Manual" (DS70046) for descriptions of register bit fields.

13.3 Position Measurement Mode

There are two Measurement modes which are supported and are termed x2 and x4. These modes are selected by the QEIM<2:0> mode select bits located in SFR QEICON<10:8>.

When control bits QEIM < 2:0 > = 100 or 101, the x2 Measurement mode is selected and the QEI logic only looks at the Phase A input for the position counter increment rate. Every rising and falling edge of the Phase A signal causes the position counter to be incremented or decremented. The Phase B signal is still utilized for the determination of the counter direction, just as in the x4 mode.

Within the x2 Measurement mode, there are two variations of how the position counter is reset:

- Position counter reset by detection of index pulse, QEIM<2:0> = 100.
- Position counter reset by match with MAXCNT, QEIM<2:0> = 101.

When control bits QEIM<2:0> = 110 or 111, the x4 Measurement mode is selected and the QEI logic looks at both edges of the Phase A and Phase B input signals. Every edge of both signals causes the position counter to increment or decrement.

Within the x4 Measurement mode, there are two variations of how the position counter is reset:

- 1. Position counter reset by detection of index pulse, QEIM<2:0> = 110.
- Position counter reset by match with MAXCNT, QEIM<2:0> = 111.

The x4 Measurement mode provides for finer resolution data (more position counts) for determining motor position.

13.4 Programmable Digital Noise Filters

The digital noise filter section is responsible for rejecting noise on the incoming quadrature signals. Schmitt Trigger inputs and a three-clock cycle delay filter combine to reject low level noise and large, short duration noise spikes that typically occur in noise prone applications, such as a motor system.

The filter ensures that the filtered output signal is not permitted to change until a stable value has been registered for three consecutive clock cycles.

For the QEA, QEB and INDX pins, the clock divide frequency for the digital filter is programmed by bits QECK<2:0> (DFLTCON<6:4>) and are derived from the base instruction cycle TcY.

To enable the filter output for channels QEA, QEB and INDX, the QEOUT bit must be '1'. The filter network for all channels is disabled on POR and BOR.

13.5 Alternate 16-bit Timer/Counter

When the QEI module is not configured for the QEI mode QEIM<2:0> = 001, the module can be configured as a simple 16-bit timer/counter. The setup and control of the auxiliary timer is accomplished through the QEI-CON SFR register. This timer functions identically to Timer1. The QEA pin is used as the timer clock input.

When configured as a timer, the POSCNT register serves as the Timer Count Register and the MAXCNT register serves as the Period Register. When a timer/ period register match occur, the QEI interrupt flag will be asserted.

The only exception between the general purpose timers and this timer is the added feature of external Up/ Down input select. When the UPDN pin is asserted high, the timer will increment up. When the UPDN pin is asserted low, the timer will be decremented.

Note:	Char	Changing the Operational mode (i.e., from							
	QEI	QEI to Timer or vice versa), will not affect							
	the	Timer/Position	Count	Register					
	conte	ents.							

The UPDN control/status bit (QEICON<11>) can be used to select the count direction state of the Timer register. When UPDN = 1, the timer will count up. When UPDN = 0, the timer will count down.

In addition, control bit UPDN_SRC (QEICON<0>) determines whether the timer count direction state is based on the logic state, written into the UPDN control/ status bit (QEICON<11>), or the QEB pin state. When UPDN_SRC = 1, the timer count direction is controlled from the QEB pin. Likewise, when UPDN_SRC = 0, the timer count direction is controlled by the UPDN bit.

Note: This Timer does not support the External Asynchronous Counter mode of operation. If using an external clock source, the clock will automatically be synchronized to the internal instruction cycle.

13.6 QEI Module Operation During CPU Sleep Mode

13.6.1 QEI OPERATION DURING CPU SLEEP MODE

The QEI module will be halted during the CPU Sleep mode.

13.6.2 TIMER OPERATION DURING CPU SLEEP MODE

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

16.4.2 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.

16.5 Automatic Clock Stretch

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

16.5.1 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.

16.5.2 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.

16.5.3 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. On 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.

16.5.4 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.

16.6 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 de-asserted 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.

16.7 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.

17.0 UNIVERSAL ASYNCHRONOUS RECEIVER TRANSMITTER (UART) 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 section describes the Universal Asynchronous Receiver/Transmitter Communications module.

Note: Since dsPIC30F2010 devices have only one UART, all references to Ux... imply that x = 1 only.

17.1 UART Module Overview

The key features of the UART module are:

- Full-duplex, 8- or 9-bit data communication
- Even, Odd or No Parity options (for 8-bit data)
- · One or two Stop bits
- Fully integrated Baud Rate Generator with 16-bit prescaler
- Baud rates range from 38 bps to 1.875 Mbps at a 30 MHz instruction rate
- 4-word deep transmit data buffer
- 4-word deep receive data buffer
- Parity, Framing and Buffer Overrun error detection
- Support for Interrupt only on Address Detect (9th bit = 1)
- Separate Transmit and Receive Interrupts
- Loopback mode for diagnostic support

FIGURE 17-1: UART TRANSMITTER BLOCK DIAGRAM



17.10 UART Operation During CPU Sleep and Idle Modes

17.10.1 UART OPERATION DURING CPU SLEEP MODE

When the device enters Sleep mode, all clock sources to the module are shutdown and stay at logic '0'. If entry into Sleep mode occurs while a transmission is in progress, then the transmission is aborted. The UxTX pin is driven to logic '1'. Similarly, if entry into Sleep mode occurs while a reception is in progress, then the reception is aborted. The UxSTA, UxMODE, transmit and receive registers and buffers, and the UxBRG register are not affected by Sleep mode.

If the WAKE bit (UxMODE<7>) is set before the device enters Sleep mode, then a falling edge on the UxRX pin will generate a receive interrupt. The Receive Interrupt Select mode bit (URXISEL) has no effect for this function. If the receive interrupt is enabled, then this will wake-up the device from Sleep. The UARTEN bit must be set in order to generate a wake-up interrupt.

17.10.2 UART OPERATION DURING CPU IDLE MODE

For the UART, the USIDL bit selects if the module will stop operation when the device enters Idle mode, or whether the module will continue on Idle. If USIDL = 0, the module will continue operation during Idle mode. If USIDL = 1, the module will stop on Idle.

dsPIC30F2010

NOTES:

18.9 Module Power-Down Modes

The module has three internal power modes. When the ADON bit is '1', the module is in Active mode; it is fully powered and functional. When ADON is '0', the module is in Off mode. The digital and analog portions of the circuit are disabled for maximum current savings. In order to return to the Active mode from Off mode, the user must wait for the ADC circuitry to stabilize.

18.10 A/D Operation During CPU Sleep and Idle Modes

18.10.1 A/D OPERATION DURING CPU SLEEP MODE

When the device enters Sleep mode, all clock sources to the module are shutdown and stay at logic '0'.

If Sleep occurs in the middle of a conversion, the conversion is aborted. The converter will not continue with a partially completed conversion on exit from Sleep mode.

Register contents are not affected by the device entering or leaving Sleep mode.

The A/D module can operate during Sleep mode if the A/D clock source is set to RC (ADRC = 1). When the RC clock source is selected, the A/D module waits one instruction cycle before starting the conversion. This allows the SLEEP instruction to be executed, which eliminates all digital switching noise from the conversion. When the conversion is complete, the DONE bit will be set and the result loaded into the ADCBUF register.

If the A/D interrupt is enabled, the device will wake-up from Sleep. If the A/D interrupt is not enabled, the A/D module will then be turned off, although the ADON bit will remain set.

18.10.2 A/D OPERATION DURING CPU IDLE MODE

The ADSIDL bit selects if the module will stop on Idle or continue on Idle. If ADSIDL = 0, the module will continue operation on assertion of Idle mode. If ADSIDL = 1, the module will stop on Idle.

18.11 Effects of a Reset

A device Reset forces all registers to their Reset state. This forces the A/D module to be turned off, and any conversion and acquisition sequence is aborted. The values that are in the ADCBUF registers are not modified. The A/D result register will contain unknown data after a Power-on Reset.

18.12 Output Formats

The A/D result is 10 bits wide. The data buffer RAM is also 10 bits wide. The 10-bit data can be read in one of four different formats. The FORM<1:0> bits select the format. Each of the output formats translates to a 16-bit result on the data bus.

Write data will always be in right justified (integer) format.

						d09	d08	d07	d06	d05	d04	d03	d02	d01	d00
d09	d08	d07	d06	d05	d04	d03	d02	d01	d00	0	0	0	0	0	0
d09	d08	d07	d06	d05	d04	d03	d02	d01	d00	0	0	0	0	0	0
			r												
d09	d09	d09	d09	d09	d09	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00
0	0	0	0	0	0	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00
	0 0 0 0	d09 d08 d09 d08 d09 d08 d09 d09 d09 d09	d09 d08 d07 d09 d08 d07 d09 d08 d07 d09 d09 d09 d09 d09 d09	d09 d08 d07 d06 d09 d08 d07 d06 d09 d08 d07 d06 d09 d09 d09 d09 d09 d09 d09 d09 d09 d09 d09 d09	d09 d08 d07 d06 d05 d09 d08 d07 d06 d05 d09 d08 d07 d06 d05 d09 d09 d09 d09 d09 d09 d09 d09 d09 d09 d09 d09 d09 d09 d09 0 0 0 0 0	d09 d08 d07 d06 d05 d04 d09 d08 d07 d06 d05 d04 d09 d08 d07 d06 d05 d04 d09 d09 d09 d09 d09 d09 d09 d09 d09 d09 d09 d09 d09 d09 d09 d09 d09 d09 0 0 0 0 0 0	d09 d09 d08 d07 d06 d05 d04 d03 d09 d08 d07 d06 d05 d04 d03 d09 d08 d07 d06 d05 d04 d03 d09 d09 d09 d09 d09 d09 d09 d09 d09 d09 d09 d09 d09 d09 d09 d09 0 0 0 0 0 0 0 d09	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	$\begin{array}{c c c c c c c c c c c c c c c c c c c $	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	$\begin{array}{c c c c c c c c c c c c c c c c c c c $	$\begin{array}{c c c c c c c c c c c c c c c c c c c $	$\begin{array}{c c c c c c c c c c c c c c c c c c c $	$\begin{array}{c c c c c c c c c c c c c c c c c c c $

FIGURE 18-4: A/D OUTPUT DATA FORMATS

19.2.3 LP OSCILLATOR CONTROL

Enabling the LP oscillator is controlled with two elements:

- The current oscillator group bits COSC<1:0>
- The LPOSCEN bit (OSCCON register)

The LP oscillator is ON (even during Sleep mode) if LPOSCEN = 1. The LP oscillator is the device clock if:

- COSC<1:0> = 00 (LP selected as main osc.) and
- LPOSCEN = 1

Keeping the LP oscillator ON at all times allows for a fast switch to the 32 kHz system clock for lower power operation. Returning to the faster main oscillator will still require a start-up time

19.2.4 PHASE-LOCKED LOOP (PLL)

The PLL multiplies the clock that is generated by the primary oscillator. The PLL is selectable to have either gains of x4, x8, and x16. Input and output frequency ranges are summarized in Table 19-3.

TABLE 19-3: PLL FREQUENCY RANGE

Fin	PLL Multiplier	Fout
4 MHz-10 MHz	x4	16 MHz-40 MHz
4 MHz-10 MHz	x8	32 MHz-80 MHz
4 MHz-7.5 MHz	x16	64 MHz-120 MHz

The PLL features a lock output which is asserted when the PLL enters a phase locked state. Should the loop fall out of lock (e.g., due to noise), the lock signal will be rescinded. The state of this signal is reflected in the read-only LOCK bit in the OSCCON register.

19.2.5 FAST RC OSCILLATOR (FRC)

The FRC oscillator is a fast (7.37 MHz \pm 2% nominal) internal RC oscillator. This oscillator is intended to provide reasonable device operating speeds without the use of an external crystal, ceramic resonator or RC network.

The dsPIC30F operates from the FRC oscillator when the current oscillator selection control bits in the OSCCON register (OSCCON<13:12>) are set to '01'.

The four bit field specified by TUN<3:0> (OSCCON <15:14> and OSCCON<11:10>) allows the user to tune the internal fast RC oscillator (nominal 7.37MHz). The user can tune the FRC oscillator within a range of -12% (or -960 kHz) to +10.5% (or +840 kHz) in steps of 1.50% around the factory calibrated setting, see Table 19-4.

Note:	OSCTUN functionality has been provided
	to help customers compensate for
	temperature effects on the FRC frequency
	over a wide range of temperatures. The
	tuning step size is an approximation and is
	neither characterized nor tested.

TUN<3:0> Bits	FRC Frequency
0111	+ 10.5%
0110	+ 9.0%
0101	+ 7.5%
0100	+ 6.0%
0011	+ 4.5%
0010	+ 3.0%
0001	+ 1.5%
0000	Center Frequency (oscillator is running at calibrated frequency)
1111	- 1.5%
1110	- 3.0%
1101	- 4.5%
1100	- 6.0%
1011	- 7.5%
1010	- 9.0%
1001	- 10.5%
1000	- 12.0%

19.2.6 LOW-POWER RC OSCILLATOR (LPRC)

The LPRC oscillator is a component of the Watchdog Timer (WDT) and oscillates at a nominal frequency of 512 kHz. The LPRC oscillator is the clock source for the Power-up Timer (PWRT) circuit, WDT, and clock monitor circuits. It may also be used to provide a low frequency clock source option for applications where power consumption is critical and timing accuracy is not required

The LPRC oscillator is always enabled at a Power-on Reset because it is the clock source for the PWRT. After the PWRT expires, the LPRC oscillator will remain on if one of the following is true:

- The Fail-Safe Clock Monitor is enabled
- The WDT is enabled
- The LPRC oscillator is selected as the system clock via the COSC<1:0> control bits in the OSCCON register

If one of the above conditions is not true, the LPRC will shut-off after the PWRT expires.

Note 1: OSC2 pin function is determined by the Primary Oscillator mode selection (FPR<3:0>).

 OSC1 pin cannot be used as an I/O pin even if the secondary oscillator or an internal clock source is selected at all times. Most single-word instructions are executed in a single instruction cycle, unless a conditional test is true or the program counter is changed as a result of the instruction. In these cases, the execution takes two instruction cycles with the additional instruction cycle(s) executed as a NOP. Notable exceptions are the BRA (unconditional/computed branch), indirect CALL/GOTO, all table reads and writes and RETURN/RETFIE instructions, which are single-word instructions, but take two or three cycles. Certain instructions that involve skipping over the subsequent instruction, require either two or three cycles if the skip is performed, depending on whether the instruction being skipped is a single-word or two-word instruction. Moreover, double word moves require two cycles. The double word instructions execute in two instruction cycles.

Note:	For more det	ails on the insti	ruction set,
	refer to the "	16-bit MCU and	DSC Pro-
	grammer's	Reference	Manual"
	(DS70157).		

Field Description					
#text	Means literal defined by "text"				
(text)	Means "content of text"				
[text]	Means "the location addressed by text"				
{ }	Optional field or operation				
<n:m></n:m>	Register bit field				
.b	Byte mode selection				
.d	Double word mode selection				
.S	Shadow register select				
.w	Word mode selection (default)				
Acc	One of two accumulators {A, B}				
AWB	Accumulator write-back destination address register \in {W13, [W13] + = 2}				
bit4	4-bit bit selection field (used in word addressed instructions) $\in \{015\}$				
C, DC, N, OV, Z	MCU status bits: Carry, Digit Carry, Negative, Overflow, Zero				
Expr	Absolute address, label or expression (resolved by the linker)				
f	File register address ∈ {0x00000x1FFF}				
lit1	1-bit unsigned literal ∈ {0,1}				
lit4	4-bit unsigned literal ∈ {015}				
lit5	5-bit unsigned literal ∈ {031}				
lit8	8-bit unsigned literal ∈ {0255}				
lit10	10-bit unsigned literal $\in \{0255\}$ for Byte mode, $\{0:1023\}$ for Word mode				
lit14	14-bit unsigned literal ∈ {016384}				
lit16	16-bit unsigned literal ∈ {065535}				
lit23	23-bit unsigned literal $\in \{08388608\}$; LSB must be 0				
None	Field does not require an entry, may be blank				
OA, OB, SA, SB	DSP status bits: ACCA Overflow, ACCB Overflow, ACCA Saturate, ACCB Saturate				
PC	Program Counter				
Slit10	10-bit signed literal ∈ {-512511}				
Slit16	16-bit signed literal ∈ {-3276832767}				
Slit6	6-bit signed literal ∈ {-1616}				

TABLE 20-1: SYMBOLS USED IN OPCODE DESCRIPTIONS

Field	Description						
Wb	Base W register ∈ {W0W15}						
Wd	Destination W register ∈ {Wd, [Wd], [Wd++], [Wd], [++Wd], [Wd]}						
Wdo	Destination W register ∈ {Wnd, [Wnd], [Wnd++], [Wnd], [++Wnd], [Wnd], [Wnd+Wb]}						
Wm,Wn	Dividend, Divisor working register pair (direct addressing)						
Wm*Wm	Multiplicand and Multiplier working register pair for Square instructions \in {W4 * W4,W5 * W5,W6 * W6,W7 * W7}						
Wm*Wn	Multiplicand and Multiplier working register pair for DSP instructions ∈ {W4 * W5,W4 * W6,W4 * W7,W5 * W6,W5 * W7,W6 * W7}						
Wn	One of 16 working registers ∈ {W0W15}						
Wnd	One of 16 destination working registers ∈ {W0W15}						
Wns	One of 16 source working registers ∈ {W0W15}						
WREG	W0 (working register used in file register instructions)						
Ws	Source W register ∈ {Ws, [Ws], [Ws++], [Ws], [++Ws], [Ws]}						
Wso	Source W register ∈ {Wns, [Wns], [Wns++], [Wns], [++Wns], [Wns], [Wns+Wb]}						
Wx	X data space prefetch address register for DSP instructions ∈ {[W8] + = 6, [W8] + = 4, [W8] + = 2, [W8], [W8] - = 6, [W8] - = 4, [W8] - = 2, [W9] + = 6, [W9] + = 4, [W9] + = 2, [W9], [W9] - = 6, [W9] - = 4, [W9] - = 2, [W9 + W12],none}						
Wxd	X data space prefetch destination register for DSP instructions \in {W4W7}						
Wy	Y data space prefetch address register for DSP instructions ∈ {[W10] + = 6, [W10] + = 4, [W10] + = 2, [W10], [W10] - = 6, [W10] - = 4, [W10] - = 2, [W11] + = 6, [W11] + = 4, [W11] + = 2, [W11], [W11] - = 6, [W11] - = 4, [W11] - = 2, [W11 + W12], none}						
Wyd	Y data space prefetch destination register for DSP instructions \in {W4W7}						

TABLE 20-1: SYMBOLS USED IN OPCODE DESCRIPTIONS (CONTINUED)

TABLE 22-23: TIMER2 EXTERNAL CLOCK TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 2.5V to 5.5V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended						
Param No.	Symbol	Characte	eristic		Min	Тур	Мах	Units	Conditions
TB10	TtxH	TxCK High Time	Synchronous, no prescaler Synchronous, with prescaler		0.5 TCY + 20		—	ns	Must also meet parameter TB15
					10	_	—	ns	
TB11	TtxL	TxCK Low Time	Synchronous, no prescaler Synchronous, with prescaler		0.5 TCY + 20	_	—	ns	Must also meet parameter TB15
					10		—	ns	
TB15	TtxP	TxCK Input Period	J Synchronous, no prescaler		Tcy + 10	—	—	ns	N = prescale value
			Synchro with pre	nous, scaler	Greater of: 20 ns or (TCY + 40)/N				(1, 8, 64, 256)
TB20	TCKEXTMRL	Delay from Externa Edge to Timer Incre	al TxCK Clock		0.5 TCY	_	1.5 TCY	_	_

TABLE 22-24: TIMER3 EXTERNAL CLOCK TIMING REQUIREMENTS

AC CHARACTERISTICS

Standard Operating Conditions: 2.5V to 5.5V(unless otherwise stated)Operating temperature-40°C ≤TA ≤+85°C for Industrial

				·	•	-40°0	C ≤TA ≤+1	25°C fo	r Extended
Param No.	Symbol	Characteristic			Min	Тур	Max	Units	Conditions
TC10	TtxH	TxCK High Time	Synchror	nous	0.5 Tcy + 20		_	ns	Must also meet parameter TC15
TC11	TtxL	TxCK Low Time	Synchror	nous	0.5 TCY + 20		_	ns	Must also meet parameter TC15
TC15	TtxP	TxCK Input Period	Synchror no presc	nous, aler	Tcy + 10		_	ns	N = prescale value
			Synchror with pres	nous, scaler	Greater of: 20 ns or (TCY + 40)/N				(1, 8, 64, 256)
TC20	TCKEXTMRL	Delay from External TxCK Clock Edge to Timer Increment		0.5 TCY	_	1.5 TCY	_	—	

dsPIC30F2010



TABLE 22-30: QUADRATURE DECODER TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 2.5V to 5.5V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended				
Param No.	Symbol	Characteristic ⁽¹⁾		Тур ⁽²⁾	Мах	Units	Conditions
TQ30	TQUL	Quadrature Input Low Time		6 Тсү	—	ns	—
TQ31	ΤαυΗ	Quadrature Input High Time		6 Tcy	—	ns	—
TQ35	TQUIN	Quadrature Input Period		12 Tcy	—	ns	—
TQ36	ΤουΡ	Quadrature Phase Period		3 TCY	—	ns	—
TQ40	TQUFL	Filter Time to Recognize Low with Digital Filter	Ι,	3 * N * Tcy	—	ns	N = 1, 2, 4, 16, 32, 64, 128 and 256 (Note 2)
TQ41	TQUFH	Filter Time to Recognize Hig with Digital Filter	h,	3 * N * Tcy		ns	N = 1, 2, 4, 16, 32, 64, 128 and 256 (Note 2)

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

2: N = Index Channel Digital Filter Clock Divide Select Bits. Refer to Section 16. "Quadrature Encoder Interface (QEI)" (DS70063) in the "dsPIC30F Family Reference Manual" (DS70046).

TABLE 22-37: I²C[™] BUS DATA TIMING REQUIREMENTS (SLAVE MODE)

AC CHARACTERISTICS			Standard Operating Conditions: 2.5V to 5.5V(unless otherwise stated)Operating temperature $-40^{\circ}C \leq TA \leq +85^{\circ}C$ for Industrial $-40^{\circ}C \leq TA \leq +125^{\circ}C$ for Extended						
Param No.	Symbol	Charact	eristic 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	SDA and SCL	100 kHz mode	—	300	ns	CB is specified to be from		
		Fall Time	400 kHz mode	20 + 0.1 Св	300	ns	10 to 400 pF		
			1 MHz mode ⁽¹⁾	—	100	ns			
IS21	TR:SCL	SDA and SCL Rise Time	100 kHz mode	—	1000	ns	CB is specified to be from		
			400 kHz mode	20 + 0.1 Св	300	ns	10 to 400 pF		
			1 MHz mode ⁽¹⁾	—	300	ns			
IS25	TSU:DAT	Data Input Setup Time	100 kHz mode	250		ns			
			400 kHz mode	100	_	ns	1 —		
			1 MHz mode ⁽¹⁾	100	_	ns]		
IS26	THD:DAT	Data Input Hold Time	100 kHz mode	0		ns			
			400 kHz mode	0	0.9	μs	1 —		
			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		
			400 kHz mode	0.6		μs	Start condition		
			1 MHz mode ⁽¹⁾	0.25	_	μs			
IS31	THD:STA	Start Condition	100 kHz mode	4.0	_	μs	After this period the first		
		Hold Time	400 kHz mode	0.6		μs	clock pulse is generated		
			1 MHz mode ⁽¹⁾	0.25	_	μs			
IS33	TSU:STO	Stop Condition	100 kHz mode	4.7		μs			
		Setup Time	400 kHz mode	0.6		μs	—		
			1 MHz mode ⁽¹⁾	0.6	_	μs]		
IS34	THD:STO	Stop Condition	100 kHz mode	4000		ns			
		Hold Time	400 kHz mode	600	_	ns	1 —		
			1 MHz mode ⁽¹⁾	250		ns]		
IS40	TAA:SCL	Output Valid From	100 kHz mode	0	3500	ns			
		Clock	400 kHz mode	0	1000	ns	1 –		
			1 MHz mode ⁽¹⁾	0	350	ns]		
IS45	TBF:SDA	Bus Free Time	100 kHz mode	4.7	—	μs	Time the bus must be free		
			400 kHz mode	1.3	—	μs	before a new transmission		
			1 MHz mode ⁽¹⁾	0.5	—	μs	can start		
IS50	Св	Bus Capacitive Loading		—	400	pF	-		

Note 1: Maximum pin capacitance = 10 pF for all I^2C^{TM} pins (for 1 MHz mode only).

PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.

