Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

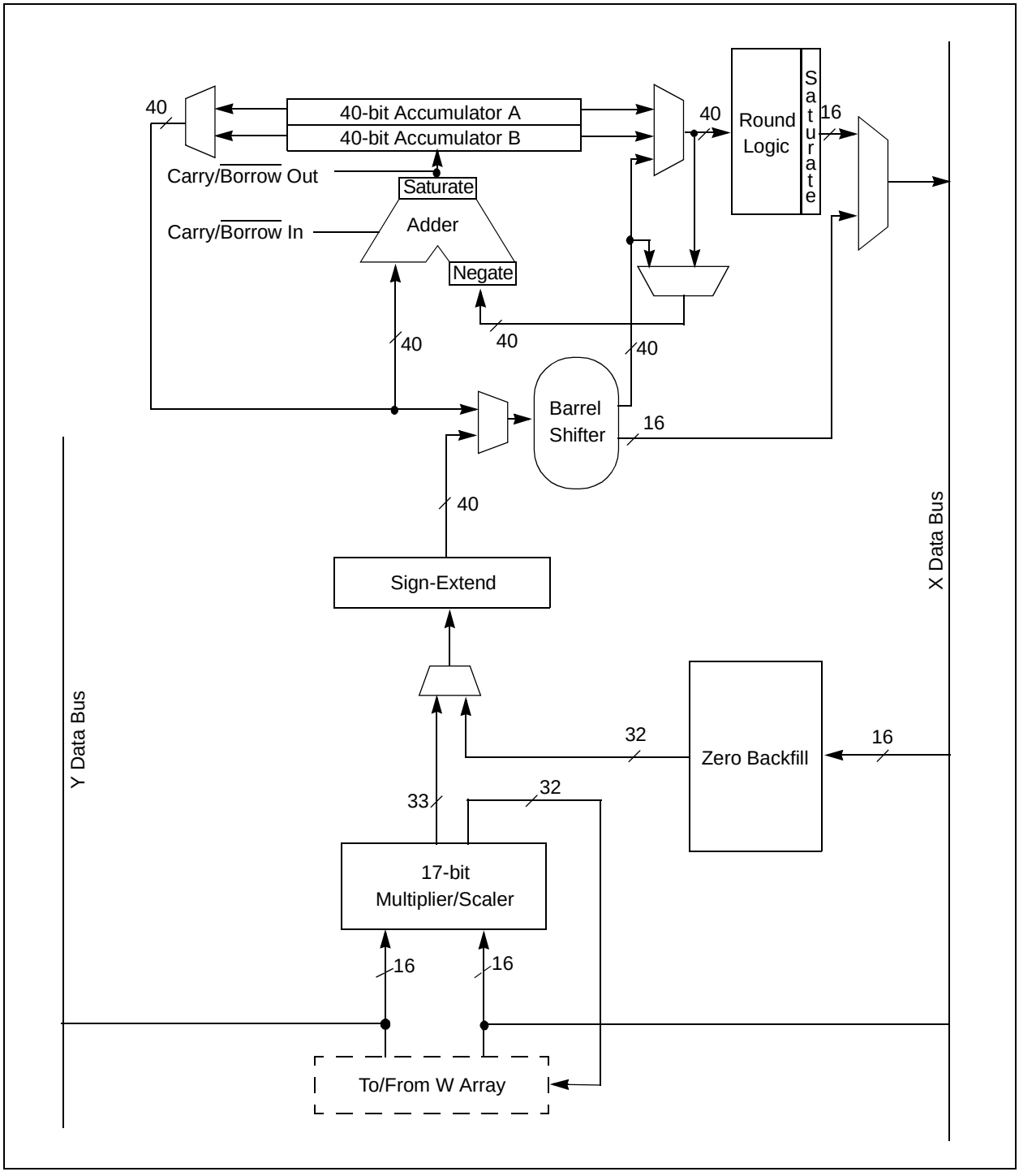
Details	
Product Status	Active
Core Processor	dsPIC
Core Size	16-Bit
Speed	30 MIPs
Connectivity	CANbus, I²C, SPI, UART/USART
Peripherals	AC'97, Brown-out Detect/Reset, I²S, LVD, POR, PWM, WDT
Number of I/O	68
Program Memory Size	144KB (48K x 24)
Program Memory Type	FLASH
EEPROM Size	4K x 8
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	2.5V ~ 5.5V
Data Converters	A/D 16x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	80-TQFP
Supplier Device Package	80-TQFP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/dspic30f6014-30i-pf

1.0 DEVICE 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 "*dsPIC30F/33F Programmer's Reference Manual*" (DS70157).

This document contains specific information for the dsPIC30F6011/6012/6013/6014 Digital Signal Controller (DSC) devices. The dsPIC30F devices contain extensive Digital Signal Processor (DSP) functionality within a high-performance 16-bit microcontroller (MCU) architecture. Figure 1-1 and Figure 1-2 show device block diagrams for dsPIC30F6011/6012 and dsPIC30F6013/6014 respectively.

FIGURE 2-2: DSP ENGINE BLOCK DIAGRAM



© 2006 Microchip Technology Inc.

3.2 Data Address Space

The core has two data spaces. The data spaces can be considered either separate (for some DSP instructions), or as one unified linear address range (for MCU instructions). The data spaces are accessed using two Address Generation Units (AGUs) and separate data paths.

3.2.1 DATA SPACE MEMORY MAP

The data space memory is split into two blocks, X and Y data space. A key element of this architecture is that Y space is a subset of X space, and is fully contained within X space. In order to provide an apparent linear addressing space, X and Y spaces have contiguous addresses.

When executing any instruction other than one of the MAC class of instructions, the X block consists of the 64-Kbyte data address space (including all Y addresses). When executing one of the MAC class of instructions, the X block consists of the 64-Kbyte data address space excluding the Y address block (for data reads only). In other words, all other instructions regard the entire data memory as one composite address space. The MAC class instructions extract the Y address space from data space and address it using EAs sourced from W10 and W11. The remaining X data space is addressed using W8 and W9. Both address spaces are concurrently accessed only with the MAC class instructions.

The data space memory maps are shown in Figure 3-8 and Figure 3-9.

3.2.2 DATA SPACES

The X data space is used by all instructions and supports all addressing modes. There are separate read and write data buses. The X read data bus is the return data path for all instructions that view data space as combined X and Y address space. It is also the X address space data path for the dual operand read instructions (MAC class). The X write data bus is the only write path to data space for all instructions.

The X data space also supports Modulo Addressing for all instructions, subject to addressing mode restrictions. Bit-Reversed Addressing is only supported for writes to X data space.

The Y data space is used in concert with the X data space by the MAC class of instructions (CLR, ED, EDAC, MAC, MOVSAC, MPY, MPY.N and MSC) to provide two concurrent data read paths. No writes occur across the Y bus. This class of instructions dedicates two W register pointers, W10 and W11, to always address Y data space, independent of X data space, whereas W8 and W9 always address X data space. Note that during accumulator write back, the data address space is considered a combination of X and Y data spaces, so the write occurs across the X bus. Consequently, the write can be to any address in the entire data space.

The Y data space can only be used for the data prefetch operation associated with the MAC class of instructions. It also supports Modulo Addressing for automated circular buffers. Of course, all other instructions can access the Y data address space through the X data path as part of the composite linear space.

The boundary between the X and Y data spaces is defined as shown in Figure 3-8 and Figure 3-8 and is not user programmable. Should an EA point to data outside its own assigned address space, or to a location outside physical memory, an all zero word/byte will be returned. For example, although Y address space is visible by all non-MAC instructions using any addressing mode, an attempt by a MAC instruction to fetch data from that space using W8 or W9 (X space pointers) will return 0x0000.

4.1.3 MOVE AND ACCUMULATOR INSTRUCTIONS

Move instructions and the DSP accumulator class of instructions provide a greater degree of addressing flexibility than other instructions. In addition to the addressing modes supported by most MCU instructions, move and accumulator instructions also support Register Indirect with Register Offset Addressing mode, also referred to as Register Indexed mode.

Note:	For the MOV instructions, the addressing mode specified in the instruction can differ for the source and destination EA. However, the 4-bit Wb (register offset) field is shared between both source and destination (but typically only used by one).

In summary, the following addressing modes are supported by move and accumulator instructions:

- Register Direct
- Register Indirect
- Register Indirect Post-modified
- Register Indirect Pre-modified
- Register Indirect with Register Offset (Indexed)
- Register Indirect with Literal Offset
- 8-bit Literal
- 16-bit Literal

Note:	Not all instructions support all the addressing modes given above. Individual instructions may support different subsets of these addressing modes.

4.1.4 MAC INSTRUCTIONS

The dual source operand DSP instructions (CLR, ED, EDAC, MAC, MPY, MPY.N, MOVSAC and MSC), also referred to as MAC instructions, utilize a simplified set of addressing modes to allow the user to effectively manipulate the data pointers through register indirect tables.

The 2 source operand prefetch registers must be a member of the set {W8, W9, W10, W11}. For data reads, W8 and W9 will always be directed to the X RAGU and W10 and W11 will always be directed to the Y AGU. The effective addresses generated (before and after modification) must, therefore, be valid addresses within X data space for W8 and W9 and Y data space for W10 and W11.

Note:	Register indirect with register offset addressing is only available for W9 (in X space) and W11 (in Y space).

In summary, the following addressing modes are supported by the MAC class of instructions:

- Register Indirect
- Register Indirect Post-modified by 2
- Register Indirect Post-modified by 4
- Register Indirect Post-modified by 6
- Register Indirect with Register Offset (Indexed)

4.1.5 OTHER INSTRUCTIONS

Besides the various addressing modes outlined above, some instructions use literal constants of various sizes. For example, BRA (branch) instructions use 16-bit signed literals to specify the branch destination directly, whereas the DISI instruction uses a 14-bit unsigned literal field. In some instructions, such as ADD Acc, the source of an operand or result is implied by the opcode itself. Certain operations, such as NOP, do not have any operands.

4.2 Modulo Addressing

Modulo Addressing is a method of providing an automated means to support circular data buffers using hardware. The objective is to remove the need for software to perform data address boundary checks when executing tightly looped code, as is typical in many DSP algorithms.

Modulo Addressing can operate in either data or program space (since the data pointer mechanism is essentially the same for both). One circular buffer can be supported in each of the X (which also provides the pointers into program space) and Y data spaces. Modulo Addressing can operate on any W register pointer. However, it is not advisable to use W14 or W15 for Modulo Addressing since these two registers are used as the Stack Frame Pointer and Stack Pointer, respectively.

In general, any particular circular buffer can only be configured to operate in one direction, as there are certain restrictions on the buffer start address (for incrementing buffers), or end address (for decrementing buffers) based upon the direction of the buffer.

The only exception to the usage restrictions is for buffers which have a power-of-2 length. As these buffers satisfy the start and end address criteria, they may operate in a Bidirectional mode (i.e., address boundary checks will be performed on both the lower and upper address boundaries).

© 2006 Microchip Technology Inc.

6.6.3 LOADING WRITE LATCHES

Example 6-2 shows a sequence of instructions that can be used to load the 96 bytes of write latches. 32 TBLWTL and 32 TBLWTH instructions are needed to load the write latches selected by the table pointer.

EXAMPLE 6-2: LOADING WRITE LATCHES

```
; Set up a pointer to the first program memory location to be written
; program memory selected, and writes enabled
        MOV     #0x0000,W0                  ;
        MOV     W0,TBLPAG                   ; Initialize PM Page Boundary SFR
        MOV     #0x6000,W0                  ; An example program memory address
; Perform the TBLWT instructions to write the latches
; 0th_program_word
        MOV     #LOW_WORD_0,W2              ;
        MOV     #HIGH_BYTE_0,W3             ;
        TBLWTL  W2,[W0]                     ; Write PM low word into program latch
        TBLWTH  W3,[W0++]                   ; Write PM high byte into program latch
; 1st_program_word
        MOV     #LOW_WORD_1,W2              ;
        MOV     #HIGH_BYTE_1,W3             ;
        TBLWTL  W2,[W0]                     ; Write PM low word into program latch
        TBLWTH  W3,[W0++]                   ; Write PM high byte into program latch
;  2nd_program_word
        MOV     #LOW_WORD_2,W2              ;
        MOV     #HIGH_BYTE_2,W3             ;
        TBLWTL  W2, [W0]                    ; Write PM low word into program latch
        TBLWTH  W3, [W0++]                  ; Write PM high byte into program latch
        •
        •
        •
; 31st_program_word
        MOV     #LOW_WORD_31,W2             ;
        MOV     #HIGH_BYTE_31,W3            ;
        TBLWTL  W2, [W0]                    ; Write PM low word into program latch
        TBLWTH  W3, [W0++]                  ; Write PM high byte into program latch
```

Note: In Example 6-2, the contents of the upper byte of W3 has no effect.

6.6.4 INITIATING THE PROGRAMMING SEQUENCE

For protection, the write initiate sequence for NVMKEY must be used to allow any erase or program operation to proceed. After the programming command has been executed, the user must wait for the programming time until programming is complete. The two instructions following the start of the programming sequence should be NOPs.

EXAMPLE 6-3: INITIATING A PROGRAMMING SEQUENCE

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


© 2006 Microchip Technology Inc.

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 NVMADR.
   c) Enable NVM interrupt (optional).
   d) Write '55' to NVMKEY.
   e) Write 'AA' 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 '55' to NVMKEY.
   d) Write 'AA' 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.

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 Non-Volatile 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.

EXAMPLE 7-4: DATA EEPROM WORD WRITE

```
; Point to data memory
    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 captures last table access address
; Select data EEPROM for 1 word op
    MOV         #0x4004,W0
    MOV         W0,NVMCON

; Operate key to allow write operation
    DISI        #5                               ; Block all interrupts with priority <7 for
                                                 ; next 5 instructions
    MOV         #0x55,W0
    MOV         W0,NVMKEY                        ; Write the 0x55 key
    MOV         #0xAA,W1
    MOV         W1,NVMKEY                        ; Write the 0xAA key
    BSET        NVMCON,#WR                       ; Initiate program sequence
    NOP
    NOP
; Write cycle will complete in 2mS. CPU is not stalled for the Data Write Cycle
; User can poll WR bit, use NVMIF or Timer IRQ to determine write complete
```

TABLE 8-1: PORTA REGISTER MAP FOR dsPIC30F6013/6014

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
TRISA	02C0	TRISA15	TRISA14	TRISA13	TRISA12	—	TRISA10	TRISA9	—	TRISA7	TRISA6	—	—	—	—	—	—	1111 0110 1100 0000
PORTA	02C2	RA15	RA14	RA13	RA12	—	RA10	RA9	—	RA7	RA6	—	—	—	—	—	—	0000 0000 0000 0000
LATA	02C4	LATA15	LATA14	LATA13	LATA12	—	LATA10	LATA9	—	LATA7	LATA6	—	—	—	—	—	—	0000 0000 0000 0000

Note 1: PORTA is not implemented in the dsPIC30F6011/6012 devices.

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

TABLE 8-2: PORTB REGISTER MAP FOR dsPIC30F6011/6012/6013/6014

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
TRISB	02C6	TRISB15	TRISB14	TRISB13	TRISB12	TRISB11	TRISB10	TRISB9	TRISB8	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111 1111 1111
PORTB	02C8	RB15	RB14	RB13	RB12	RB11	RB10	RB9	RB8	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	0000 0000 0000 0000
LATB	02CB	LATB15	LATB14	LATB13	LATB12	LATB11	LATB10	LATB9	LATB8	LATB7	LATB6	LATB5	LATB4	LATB3	LATB2	LATB1	LATB0	0000 0000 0000 0000

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

TABLE 8-3: PORTC REGISTER MAP FOR dsPIC30F6011/6012

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
TRISC	02CC	TRISC15	TRISC14	TRISC13	—	—	—	—	—	—	—	—	—	—	TRISC2	TRISC1	—	1110 0000 0000 0110
PORTC	02CE	RC15	RC14	RC13	—	—	—	—	—	—	—	—	—	—	RC2	RC1	—	0000 0000 0000 0000
LATC	02D0	LATC15	LATC14	LATC13	—	—	—	—	—	—	—	—	—	—	LATC2	LATC1	—	0000 0000 0000 0000

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

TABLE 8-4: PORTC REGISTER MAP FOR dsPIC30F6013/6014

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
TRISC	02CC	TRISC15	TRISC14	TRISC13	—	—	—	—	—	—	—	—	TRISC4	TRISC3	TRISC2	TRISC1	—	1110 0000 0001 1110
PORTC	02CE	RC15	RC14	RC13	—	—	—	—	—	—	—	—	RC4	RC3	RC2	RC1	—	0000 0000 0000 0000
LATC	02D0	LATC15	LATC14	LATC13	—	—	—	—	—	—	—	—	LATC4	LATC3	LATC2	LATC1	—	0000 0000 0000 0000

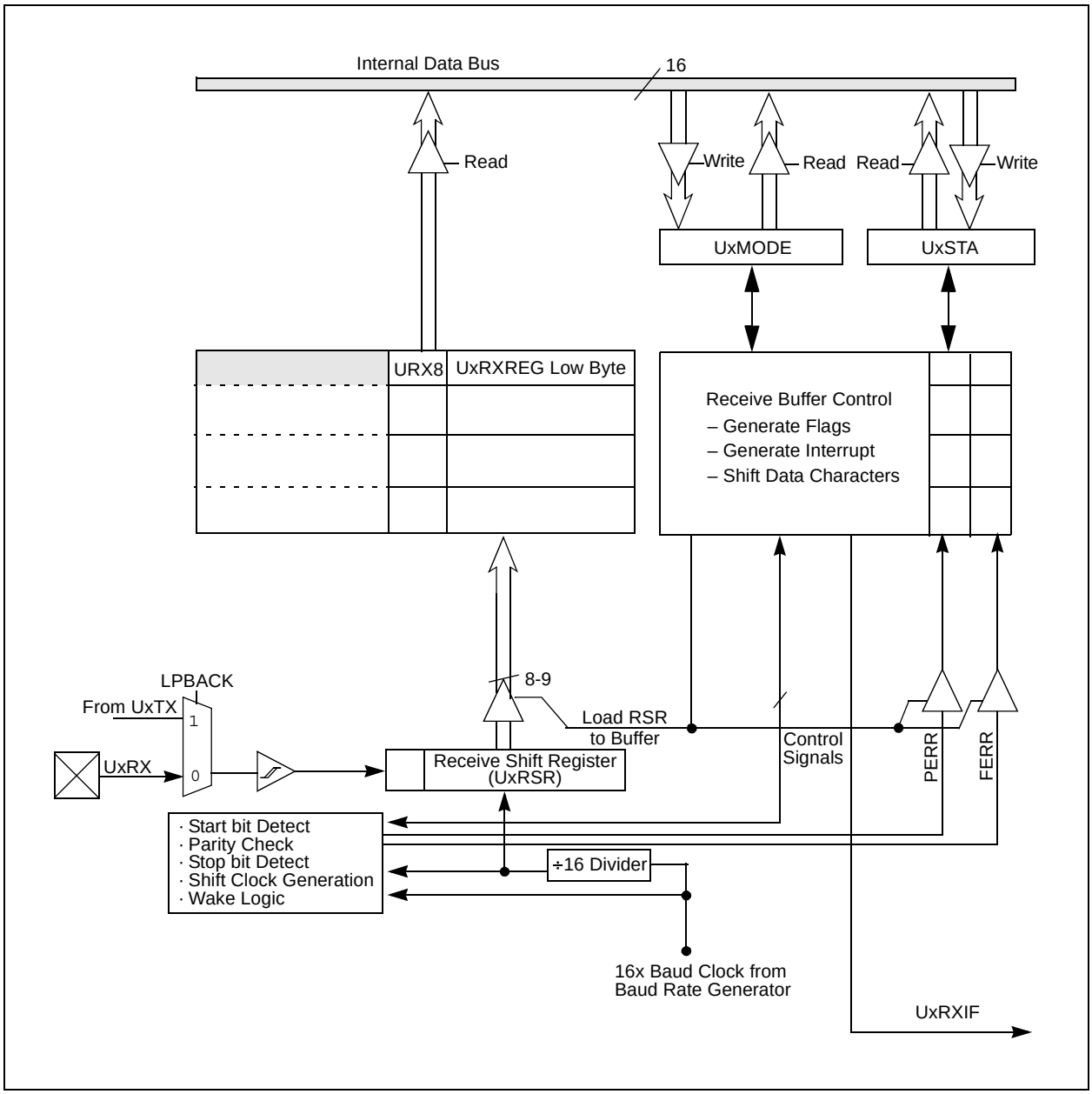
Note: Refer to "*dsPIC30F Family Reference Manual*" (DS70046) for descriptions of register bit fields.

© 2006 Microchip Technology Inc.

NOTES:

© 2006 Microchip Technology Inc.

FIGURE 16-2: UART RECEIVER BLOCK DIAGRAM



© 2006 Microchip Technology Inc.

TABLE 17-2: CAN2 REGISTER MAP (CONTINUED)

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
C2TX1EID	0412	Transmit Buffer 1 Extended Identifier <17:14>				—	—	—	—	Transmit Buffer 1 Extended Identifier <13:6>								uuuu 0000 uuuu uuuu
C2TX1DLC	0414	Transmit Buffer 1 Extended Identifier <5:0>						TXRTR	TXRB1	TXRB0	DLC<3:0>				—	—	—	uuuu uuuu uuuu u000
C2TX1B1	0416	Transmit Buffer 1 Byte 1								Transmit Buffer 1 Byte 0								uuuu uuuu uuuu uuuu
C2TX1B2	0418	Transmit Buffer 1 Byte 3								Transmit Buffer 1 Byte 2								uuuu uuuu uuuu uuuu
C2TX1B3	041A	Transmit Buffer 1 Byte 5								Transmit Buffer 1 Byte 4								uuuu uuuu uuuu uuuu
C2TX1B4	041C	Transmit Buffer 1 Byte 7								Transmit Buffer 1 Byte 6								uuuu uuuu uuuu uuuu
C2TX1CON	041E	—	—	—	—	—	—	—	—	—	TXABT	TXLARB	TXERR	TXREQ	—	TXPRI<1:0>		0000 0000 0000 0000
C2TX0SID	0420	Transmit Buffer 0 Standard Identifier <10:6>				—	—	—		Transmit Buffer 0 Standard Identifier <5:0>						SRR	TXIDE	uuuu u000 uuuu uuuu
C2TX0EID	0422	Transmit Buffer 0 Extended Identifier <17:14>				—	—	—	—	Transmit Buffer 0 Extended Identifier <13:6>								uuuu 0000 uuuu uuuu
C2TX0DLC	0424	Transmit Buffer 0 Extended Identifier <5:0>						TXRTR	TXRB1	TXRB0	DLC<3:0>				—	—	—	uuuu uuuu uuuu u000
C2TX0B1	0426	Transmit Buffer 0 Byte 1								Transmit Buffer 0 Byte 0								uuuu uuuu uuuu uuuu
C2TX0B2	0428	Transmit Buffer 0 Byte 3								Transmit Buffer 0 Byte 2								uuuu uuuu uuuu uuuu
C2TX0B3	042A	Transmit Buffer 0 Byte 5								Transmit Buffer 0 Byte 4								uuuu uuuu uuuu uuuu
C2TX0B4	042C	Transmit Buffer 0 Byte 7								Transmit Buffer 0 Byte 6								uuuu uuuu uuuu uuuu
C2TX0CON	042E	—	—	—	—	—	—	—	—	—	TXABT	TXLARB	TXERR	TXREQ	—	TXPRI<1:0>		0000 0000 0000 0000
C2RX1SID	0430	—	—	—	Receive Buffer 1 Standard Identifier <10:0>											SRR	RXIDE	000u uuuu uuuu uuuu
C2RX1EID	0432	—	—	—	—	Receive Buffer 1 Extended Identifier <17:6>												0000 uuuu uuuu uuuu
C2RX1DLC	0434	Receive Buffer 1 Extended Identifier <5:0>						RXRTR	RXRB1	—	—	—	RXRB0	DLC<3:0>				uuuu uuuu 000u uuuu
C2RX1B1	0436	Receive Buffer 1 Byte 1								Receive Buffer 1 Byte 0								uuuu uuuu uuuu uuuu
C2RX1B2	0438	Receive Buffer 1 Byte 3								Receive Buffer 1 Byte 2								uuuu uuuu uuuu uuuu
C2RX1B3	043A	Receive Buffer 1 Byte 5								Receive Buffer 1 Byte 4								uuuu uuuu uuuu uuuu
C2RX1B4	043C	Receive Buffer 1 Byte 7								Receive Buffer 1 Byte 6								uuuu uuuu uuuu uuuu
C2RX1CON	043E	—	—	—	—	—	—	—	—	RXFUL	—	—	—	RXRTRRO	FILHIT<2:0>			0000 0000 0000 0000
C2RX0SID	0440	—	—	—	Receive Buffer 0 Standard Identifier <10:0>											SRR	RXIDE	000u uuuu uuuu uuuu
C2RX0EID	0442	—	—	—	—	Receive Buffer 0 Extended Identifier <17:6>												0000 uuuu uuuu uuuu
C2RX0DLC	0444	Receive Buffer 0 Extended Identifier <5:0>						RXRTR	RXRB1	—	—	—	RXRB0	DLC<3:0>				uuuu uuuu 000u uuuu
C2RX0B1	0446	Receive Buffer 0 Byte 1								Receive Buffer 0 Byte 0								uuuu uuuu uuuu uuuu
C2RX0B2	0448	Receive Buffer 0 Byte 3								Receive Buffer 0 Byte 2								uuuu uuuu uuuu uuuu
C2RX0B3	044A	Receive Buffer 0 Byte 5								Receive Buffer 0 Byte 4								uuuu uuuu uuuu uuuu
C2RX0B4	044C	Receive Buffer 0 Byte 7								Receive Buffer 0 Byte 6								uuuu uuuu uuuu uuuu
C2RX0CON	044E	—	—	—	—	—	—	—	—	RXFUL	—	—	—	RXRTRRO	DBEN	JTOFF	FILHIT0	0000 0000 0000 0000
C2CTRL	0450	CANCAP	—	CSIDLE	ABAT	CANCKS	REQOP<2:0>			OPMODE<2:0>			—	ICODE<2:0>			—	0000 0100 1000 0000
C2CFG1	0452	—	—	—	—	—	—	—	—	SJW<1:0>		BRP<5:0>						0000 0000 0000 0000
C2CFG2	0454	—	WAKFIL	—	—	—	SEG2PH<2:0>			SEG2PHTS	SAM	SEG1PH<2:0>			PRSEG<2:0>			0u00 0uuu uuuu uuuu

Legend: u = uninitialized bit

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

© 2006 Microchip Technology Inc.

19.4 Programming the Start of Conversion Trigger

The conversion trigger will terminate acquisition and start the requested conversions.

The SSRC<2:0> bits select the source of the conversion trigger. The SSRC bits provide for up to 4 alternate sources of conversion trigger.

When SSRC<2:0> = 000, the conversion trigger is under software control. Clearing the SAMP bit will cause the conversion trigger.

When SSRC<2:0> = 111 (Auto-Start mode), the conversion trigger is under A/D clock control. The SAMC bits select the number of A/D clocks between the start of acquisition and the start of conversion. This provides the fastest conversion rates on multiple channels. SAMC must always be at least 1 clock cycle.

Other trigger sources can come from timer modules or external interrupts.

19.5 Aborting a Conversion

Clearing the ADON bit during a conversion will abort the current conversion and stop the sampling sequencing until the next sampling trigger. The ADCBUF will not be updated with the partially completed A/D conversion sample. That is, the ADCBUF will continue to contain the value of the last completed conversion (or the last value written to the ADCBUF register).

If the clearing of the ADON bit coincides with an auto-start, the clearing has a higher priority and a new conversion will not start.

19.6 Selecting the ADC Conversion Clock

The ADC conversion requires 14 TAD. The source of the ADC conversion clock is software selected, using a six-bit counter. There are 64 possible options for TAD.

EQUATION 19-1: ADC CONVERSION CLOCK

$$T_{AD} = T_{CY} * (0.5 * (ADCS<5:0> + 1))$$

The internal RC oscillator is selected by setting the ADRC bit.

For correct ADC conversions, the ADC conversion clock (TAD) must be selected to ensure a minimum TAD time of 334 nsec (for VDD = 5V). Refer to the Electrical Specifications section for minimum TAD under other operating conditions.

Example 19-1 shows a sample calculation for the ADCS<5:0> bits, assuming a device operating speed of 30 MIPS.

EXAMPLE 19-1: ADC CONVERSION CLOCK AND SAMPLING RATE CALCULATION

Minimum T_{AD} = 334 nsec

T_{CY} = 33 .33 nsec (30 MIPS)

$$ADCS<5:0> = 2\ \frac{T_{AD}}{T_{CY}} - 1$$

$$= 2 \bullet \frac{334\ nsec}{33.33\ nsec} - 1$$

$$= 19.04$$

Therefore,

Set ADCS<5:0> = 19

$$\text{Actual } T_{AD} = \frac{T_{CY}}{2}\ (ADCS<5:0> + 1)$$

$$= \frac{33.33\ nsec}{2}\ (19 + 1)$$

$$= 334\ nsec$$

If SSRC<2:0> = '111' and SAMC<4:0> = '00001'

Since,

Sampling Time = Acquisition Time + Conversion Time

= 1 TAD + 14 TAD

= 15 x 334 nsec

Therefore,

$$\text{Sampling Rate} = \frac{1}{(15 \times 334\ nsec)}$$

= ~200 kHz

20.2 Oscillator Configurations

20.2.1 INITIAL CLOCK SOURCE SELECTION

While coming out of Power-on Reset or Brown-out Reset, the device selects its clock source based on:

a) FOS<1:0> Configuration bits that select one of four oscillator groups,

b) and FPR<3:0> Configuration bits that select one of 13 oscillator choices within the primary group.

The selection is as shown in Table 20-2.

TABLE 20-2: CONFIGURATION BIT VALUES FOR CLOCK SELECTION

Oscillator Mode	Oscillator Source	FOS1	FOS0	FPR3	FPR2	FPR1	FPR0	OSC2 Function
EC	Primary	1	1	1	0	1	1	CLKO
ECIO	Primary	1	1	1	1	0	0	I/O
EC w/ PLL 4x	Primary	1	1	1	1	0	1	I/O
EC w/ PLL 8x	Primary	1	1	1	1	1	0	I/O
EC w/ PLL 16x	Primary	1	1	1	1	1	1	I/O
ERC	Primary	1	1	1	0	0	1	CLKO
ERCIO	Primary	1	1	1	0	0	0	I/O
XT	Primary	1	1	0	1	0	0	OSC2
XT w/ PLL 4x	Primary	1	1	0	1	0	1	OSC2
XT w/ PLL 8x	Primary	1	1	0	1	1	0	OSC2
XT w/ PLL 16x	Primary	1	1	0	1	1	1	OSC2
XTL	Primary	1	1	0	0	0	X	OSC2
HS	Primary	1	1	0	0	1	X	OSC2
LP	Secondary	0	0	—	—	—	—	**(Notes 1, 2)**
FRC	Internal FRC	0	1	—	—	—	—	**(Notes 1, 2)**
LPRC	Internal LPRC	1	0	—	—	—	—	**(Notes 1, 2)**

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

2: 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.

20.2.2 OSCILLATOR START-UP TIMER (OST)

In order to ensure that a crystal oscillator (or ceramic resonator) has started and stabilized, an Oscillator Start-up Timer is included. It is a simple 10-bit counter that counts 1024 TOSC cycles before releasing the oscillator clock to the rest of the system. The time-out period is designated as TOST. The TOST time is involved every time the oscillator has to restart (i.e., on POR, BOR and wake-up from Sleep). The Oscillator Start-up Timer is applied to the LP oscillator, XT, XTL, and HS modes (upon wake-up from Sleep, POR and BOR) for the primary oscillator.

20.2.3 LP OSCILLATOR CONTROL

Enabling the LP oscillator is controlled with two elements:

1. The current oscillator group bits COSC<1:0>.
2. The LPOSCEN bit (OSCON 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 oscillator) 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.

© 2006 Microchip Technology Inc.

20.3 Reset

The dsPIC30F differentiates between various kinds of Reset:

a) Power-on Reset (POR)
b) $\overline{\text{MCLR}}$ Reset during normal operation
c) $\overline{\text{MCLR}}$ Reset during Sleep
d) Watchdog Timer (WDT) Reset (during normal operation)
e) Programmable Brown-out Reset (BOR)
f) RESET Instruction
g) Reset caused by trap lockup (TRAPR)
h) Reset caused by illegal opcode or by using an uninitialized W register as an address pointer (IOPUWR)

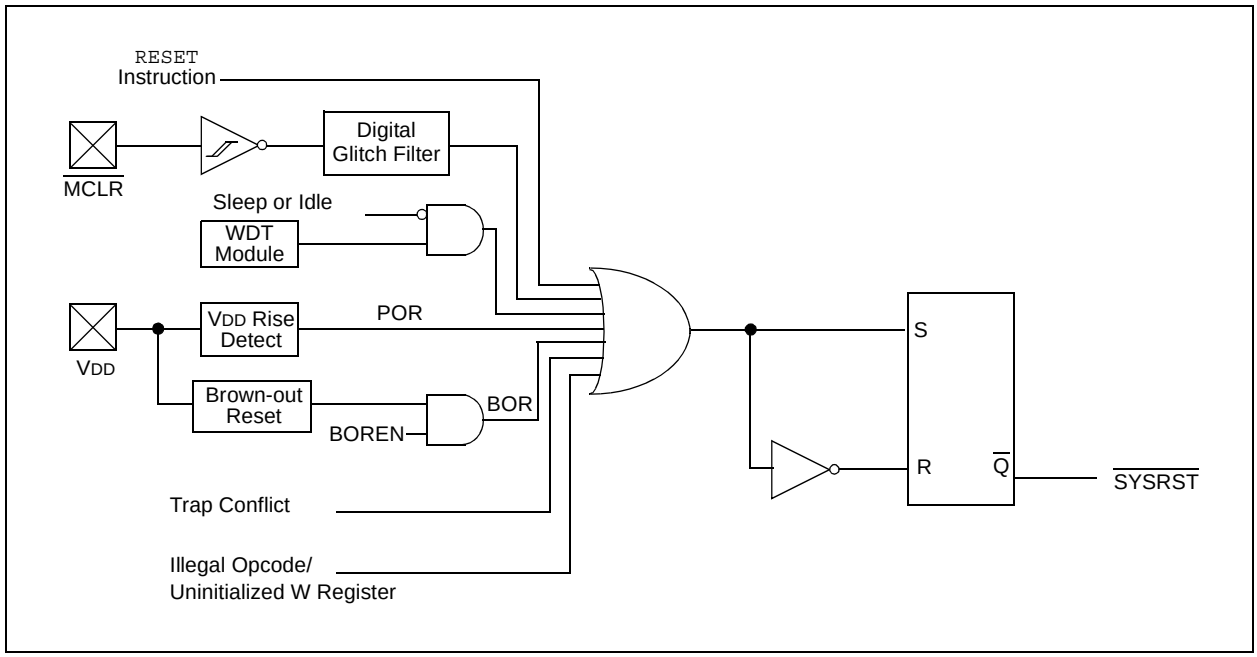
Different registers are affected in different ways by various Reset conditions. Most registers are not affected by a WDT wake-up since this is viewed as the resumption of normal operation. Status bits from the RCON register are set or cleared differently in different Reset situations, as indicated in Table 20-5. These bits are used in software to determine the nature of the Reset.

A block diagram of the On-Chip Reset Circuit is shown in Figure 20-2.

A $\overline{\text{MCLR}}$ noise filter is provided in the $\overline{\text{MCLR}}$ Reset path. The filter detects and ignores small pulses.

Internally generated Resets do not drive $\overline{\text{MCLR}}$ pin low.

FIGURE 20-2: RESET SYSTEM BLOCK DIAGRAM



20.3.1 POR: POWER-ON RESET

A power-on event will generate an internal POR pulse when a VDD rise is detected. The Reset pulse will occur at the POR circuit threshold voltage (VPOR) which is nominally 1.85V. The device supply voltage characteristics must meet specified starting voltage and rise rate requirements. The POR pulse will reset a POR timer and place the device in the Reset state. The POR also selects the device clock source identified by the oscillator configuration fuses.

The POR circuit inserts a small delay, TPOR, which is nominally 10 µs and ensures that the device bias circuits are stable. Furthermore, a user selected power-up time-out (TPWRT) is applied. The TPWRT parameter is based on device Configuration bits and can be 0 ms (no delay), 4 ms, 16 ms, or 64 ms. The total delay is at device power-up, TPOR + TPWRT. When these delays have expired, $\overline{\text{SYSRST}}$ will be negated on the next leading edge of the Q1 clock and the PC will jump to the Reset vector.

The timing for the $\overline{\text{SYSRST}}$ signal is shown in Figure 20-3 through Figure 20-5.

TABLE 23-11: ELECTRICAL CHARACTERISTICS: BOR

DC 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**	**Min**	**Typ(1)**	**Max**	**Units**	**Conditions**
BO10	VBOR	BOR Voltage(2) on VDD transition high to low	BORV = 11(3)	—	—	—	V	Not in operating range
			BORV = 10	2.6	—	2.71	V	
			BORV = 01	4.1	—	4.4	V	
			BORV = 00	4.58	—	4.73	V	
BO15	VBHYS		—	5	—	mV	

Note 1: Data in "Typ" column is at 5V, 25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

2: These parameters are characterized but not tested in manufacturing.

3: '11' values not in usable operating range.

TABLE 23-12: DC CHARACTERISTICS: PROGRAM AND EEPROM

DC 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**	**Min**	**Typ(1)**	**Max**	**Units**	**Conditions**
		Data EEPROM Memory(2)					
D120	ED	Byte Endurance	100K	1M	—	E/W	-40°C ≤ TA ≤ +85°C
D121	VDRW	VDD for Read/Write	VMIN	—	5.5	V	Using EECON to read/write VMIN = Minimum operating voltage
D122	TDEW	Erase/Write Cycle Time	—	2	—	ms	
D123	TRETD	Characteristic Retention	40	100	—	Year	Provided no other specifications are violated
D124	IDEW	IDD During Programming	—	10	30	mA	Row Erase
		Program FLASH Memory(2)					
D130	EP	Cell Endurance	10K	100K	—	E/W	-40°C ≤ TA ≤ +85°C
D131	VPR	VDD for Read	VMIN	—	5.5	V	VMIN = Minimum operating voltage
D132	VEB	VDD for Bulk Erase	4.5	—	5.5	V	
D133	VPEW	VDD for Erase/Write	3.0	—	5.5	V	
D134	TPEW	Erase/Write Cycle Time	—	2	—	ms	
D135	TRETD	Characteristic Retention	40	100	—	Year	Provided no other specifications are violated
D136	TEB	ICSP Block Erase Time	—	4	—	ms	
D137	IPEW	IDD During Programming	—	10	30	mA	Row Erase
D138	IEB	IDD During Programming	—	10	30	mA	Bulk Erase

Note 1: Data in "Typ" column is at 5V, 25°C unless otherwise stated.

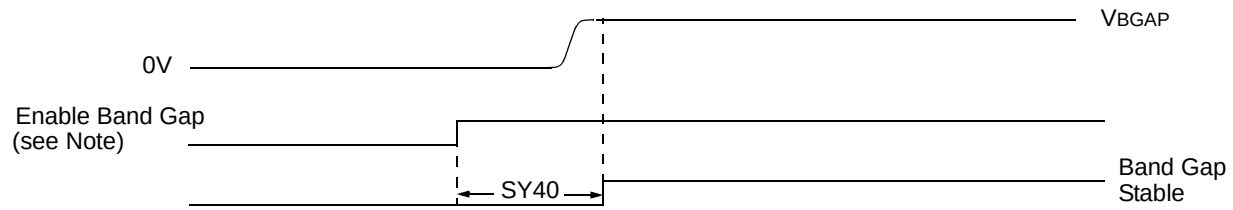
2: These parameters are characterized but not tested in manufacturing.

TABLE 23-21: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER, POWER-UP TIMER AND BROWN-OUT RESET 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)**	**Min**	**Typ(2)**	**Max**	**Units**	**Conditions**
SY25	TBOR	Brown-out Reset Pulse Width(3)	100	—	—	µs	VDD ≤ VBOR (D034)
SY30	TOST	Oscillation Start-up Timer Period	—	1024 TOSC	—	—	TOSC = OSC1 period
SY35	TFSCM	Fail-Safe Clock Monitor Delay	—	500	900	µs	-40°C to +85°C

Note 1: These parameters are characterized but not tested in manufacturing.
2: Data in "Typ" column is at 5V, 25°C unless otherwise stated.
3: Refer to Figure 23-2 and Table 23-11 for BOR.

FIGURE 23-7: BAND GAP START-UP TIME CHARACTERISTICS



Note: Set LVDEN bit (RCON<12>) or the BOREN bit (FBORPOR<7>).

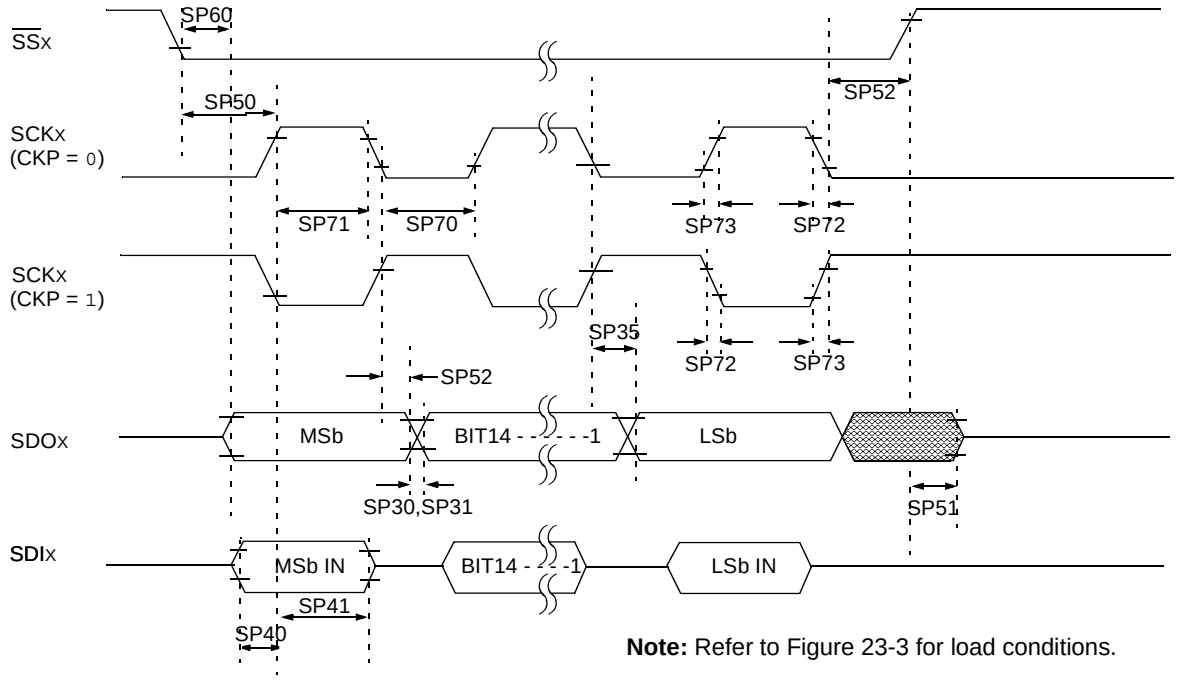
TABLE 23-22: BAND GAP START-UP TIME REQUIREMENTS

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)**	**Min**	**Typ(2)**	**Max**	**Units**	**Conditions**
SY40	TBGAP	Band Gap Start-up Time	—	40	65	µs	Defined as the time between the instant that the band gap is enabled and the moment that the band gap reference voltage is stable. RCON<13>Status bit

Note 1: These parameters are characterized but not tested in manufacturing.
2: Data in "Typ" column is at 5V, 25°C unless otherwise stated.

© 2006 Microchip Technology Inc.

FIGURE 23-17: SPI MODULE SLAVE MODE (CKE = 1) TIMING CHARACTERISTICS



Note: Refer to Figure 23-3 for load conditions.

TABLE 23-34: SPI MODULE SLAVE MODE (CKE = 1) 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)**	**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 D032
SP31	TdoR	SDOx Data Output Rise Time(3)	—	—	—	ns	See parameter D031
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	—

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 SCK is 100 ns. Therefore, the clock generated in master mode must not violate this specification.

4: Assumes 50 pF load on all SPI pins.

© 2006 Microchip Technology Inc.

TABLE 23-35: I²C™ BUS DATA TIMING REQUIREMENTS (MASTER MODE)

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**		**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 Fall Time	100 kHz mode	—	300	ns	CB is specified to be from 10 to 400 pF
			400 kHz mode	20 + 0.1 CB	300	ns	
			1 MHz mode(2)	—	100	ns	
IM21	TR:SCL	SDA and SCL Rise Time	100 kHz mode	—	1000	ns	CB is specified to be from 10 to 400 pF
			400 kHz mode	20 + 0.1 CB	300	ns	
			1 MHz mode(2)	—	300	ns	
IM25	TSU:DAT	Data Input Setup Time	100 kHz mode	250	—	ns	—
			400 kHz mode	100	—	ns	
			1 MHz mode(2)	TBD	—	ns	
IM26	THD:DAT	Data Input Hold Time	100 kHz mode	0	—	ns	—
			400 kHz mode	0	0.9	µs	
			1 MHz mode(2)	TBD	—	ns	
IM30	TSU:STA	Start Condition Setup Time	100 kHz mode	TCY/2 (BRG + 1)	—	µs	Only relevant for repeated Start condition
			400 kHz mode	TCY/2 (BRG + 1)	—	µs	
			1 MHz mode(2)	TCY/2 (BRG + 1)	—	µs	
IM31	THD:STA	Start Condition Hold Time	100 kHz mode	TCY/2 (BRG + 1)	—	µs	After this period the first clock pulse is generated
			400 kHz mode	TCY/2 (BRG + 1)	—	µs	
			1 MHz mode(2)	TCY/2 (BRG + 1)	—	µs	
IM33	TSU:STO	Stop Condition Setup 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	
IM34	THD:STO	Stop Condition Hold Time	100 kHz mode	TCY/2 (BRG + 1)	—	ns	—
			400 kHz mode	TCY/2 (BRG + 1)	—	ns	
			1 MHz mode(2)	TCY/2 (BRG + 1)	—	ns	
IM40	TAA:SCL	Output Valid From Clock	100 kHz mode	—	3500	ns	—
			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 free before a new transmission can start
			400 kHz mode	1.3	—	µs	
			1 MHz mode(2)	TBD	—	µs	
IM50	CB	Bus Capacitive Loading		—	400	pF	

Note 1: BRG is the value of the I²C Baud Rate Generator. Refer to the **"Inter-Integrated Circuit™ (I²C)"** section in the "*dsPIC30F Family Reference Manual*" (DS70046).

2: Maximum pin capacitance = 10 pF for all I²C pins (for 1 MHz mode only).

© 2006 Microchip Technology Inc.

APPENDIX A: REVISION HISTORY

Revision F (November 2006)

Previous versions of this data sheet contained Advance or Preliminary Information. They were distributed with incomplete characterization data.

Revision F of this document reflects the following updates:

- Supported I^2C Slave Addresses (see Table 15-1)
- ADC Conversion Clock selection to allow 200 kHz sampling rate (see **Section 19.0 "12-bit Analog-to-Digital Converter (A/D) Module"**)
- Operating Current (Idd) Specifications (see Table 23-5)
- BOR voltage limits (see Table 23-11)
- I/O pin Input Specifications (see Table 23-8)
- Watchdog Timer time-out limits (see Table 23-21)

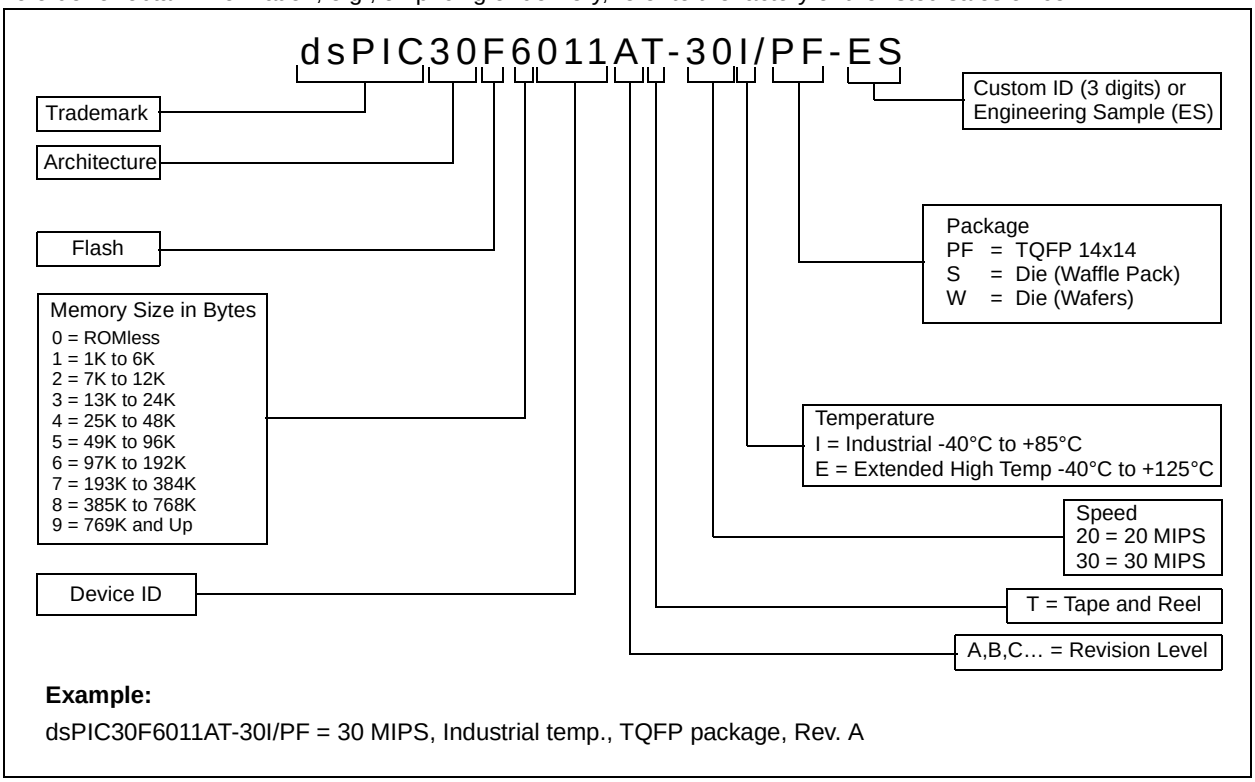
D

E

© 2006 Microchip Technology Inc.

PRODUCT IDENTIFICATION SYSTEM

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



Example:

dsPIC30F6011AT-30I/PF = 30 MIPS, Industrial temp., TQFP package, Rev. A