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	Obsolete
Core Processor	dsPIC
Core Size	16-Bit
Speed	20 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/dspic30f6014t-20i-pf

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

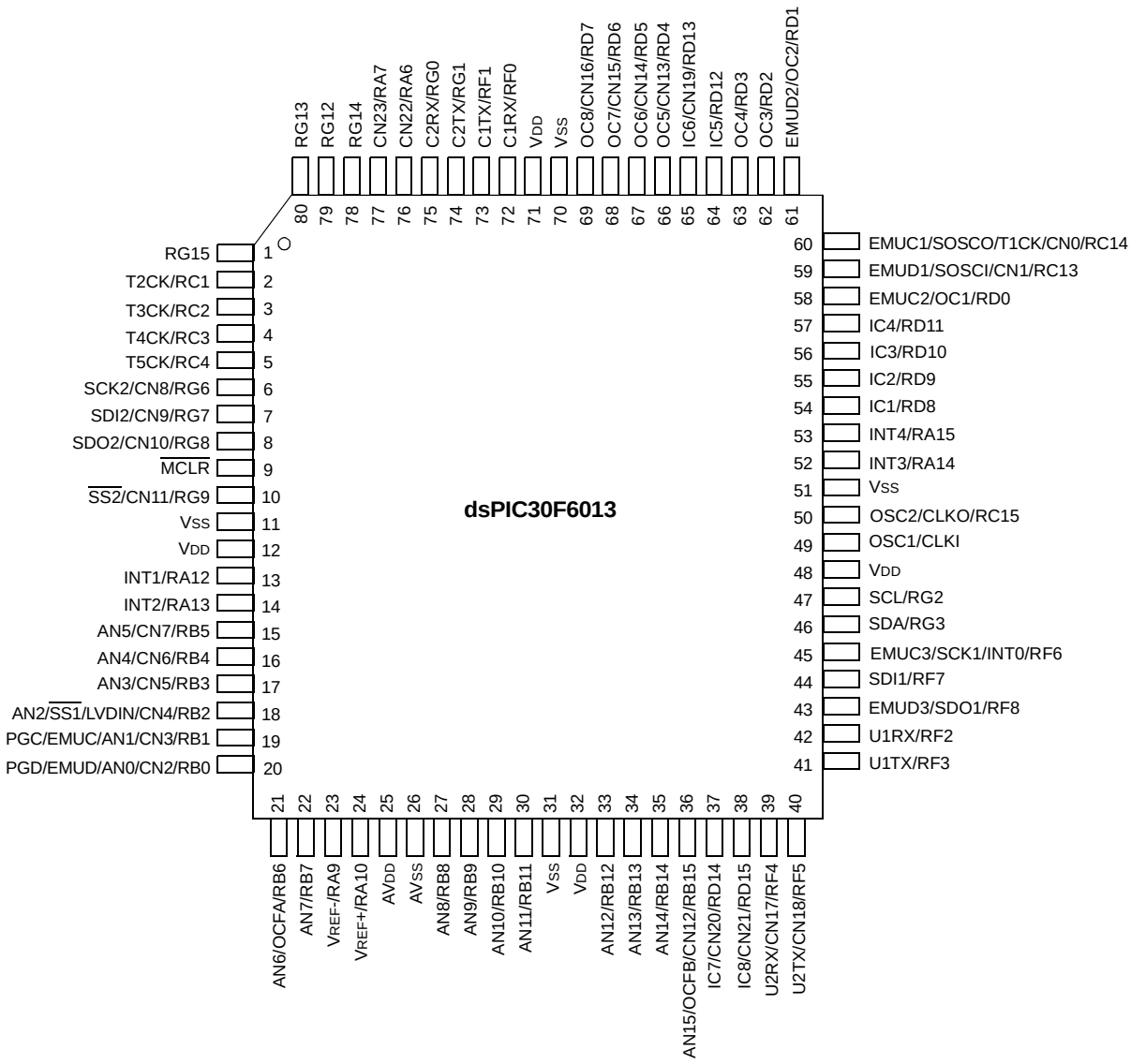
Trademarks

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
ISO/TS 16949:2002

Microchip received ISO/TS-16949:2002 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona, Gresham, Oregon and Mountain View, California. The Company's quality system processes and procedures are for its PIC® 8-bit MCUs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

Pin Diagrams (Continued)

80-Pin TQFP



*dsPIC30F6013A recommended for new designs

Note: For descriptions of individual pins, see **Section 1.0 "Device Overview"**.

Table 1-1 provides a brief description of device I/O pinouts and the functions that may be multiplexed to a port pin. Multiple functions may exist on one port pin. When multiplexing occurs, the peripheral module's functional requirements may force an override of the data direction of the port pin.

TABLE 1-1: PINOUT I/O DESCRIPTIONS

Pin Name	Pin Type	Buffer Type	Description
AN0-AN15	I	Analog	Analog input channels. AN0 and AN1 are also used for device programming data and clock inputs, respectively.
AV_{DD}	P	P	Positive supply for analog module.
AV_{SS}	P	P	Ground reference for analog module.
CLKI CLKO	I O	ST/CMOS —	External clock source input. Always associated with OSC1 pin function. Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode. Optionally functions as CLKO in RC and EC modes. Always associated with OSC2 pin function.
CN0-CN23	I	ST	Input change notification inputs. Can be software programmed for internal weak pull-ups on all inputs.
COFS CSCK CSDI CSDO	I/O I/O I O	ST ST ST —	Data Converter Interface frame synchronization pin. Data Converter Interface serial clock input/output pin. Data Converter Interface serial data input pin. Data Converter Interface serial data output pin.
C1RX C1TX C2RX C2TX	I O I O	ST — ST —	CAN1 bus receive pin. CAN1 bus transmit pin. CAN2 bus receive pin. CAN2 bus transmit pin
EMUD EMUC EMUD1 EMUC1 EMUD2 EMUC2 EMUD3 EMUC3	I/O I/O I/O I/O I/O I/O I/O I/O	ST ST ST ST ST ST ST ST	ICD Primary Communication Channel data input/output pin. ICD Primary Communication Channel clock input/output pin. ICD Secondary Communication Channel data input/output pin. ICD Secondary Communication Channel clock input/output pin. ICD Tertiary Communication Channel data input/output pin. ICD Tertiary Communication Channel clock input/output pin. ICD Quaternary Communication Channel data input/output pin. ICD Quaternary Communication Channel clock input/output pin.
IC1-IC8	I	ST	Capture inputs 1 through 8.
INT0 INT1 INT2 INT3 INT4	I I I I I	ST ST ST ST ST	External interrupt 0. External interrupt 1. External interrupt 2. External interrupt 3. External interrupt 4.
LVDIN	I	Analog	Low-Voltage Detect Reference Voltage input pin.
$\overline{MCLR}$	I/P	ST	Master Clear (Reset) input or programming voltage input. This pin is an active low Reset to the device.
OCFA OCFB OC1-OC8	I I O	ST ST —	Compare Fault A input (for Compare channels 1, 2, 3 and 4). Compare Fault B input (for Compare channels 5, 6, 7 and 8). Compare outputs 1 through 8.

Legend: CMOS = CMOS compatible input or output; Analog = Analog input
ST = Schmitt Trigger input with CMOS levels; O = Output
I = Input; P = Power

FIGURE 3-9: DATA SPACE FOR MCU AND DSP (MAC CLASS) INSTRUCTIONS EXAMPLE

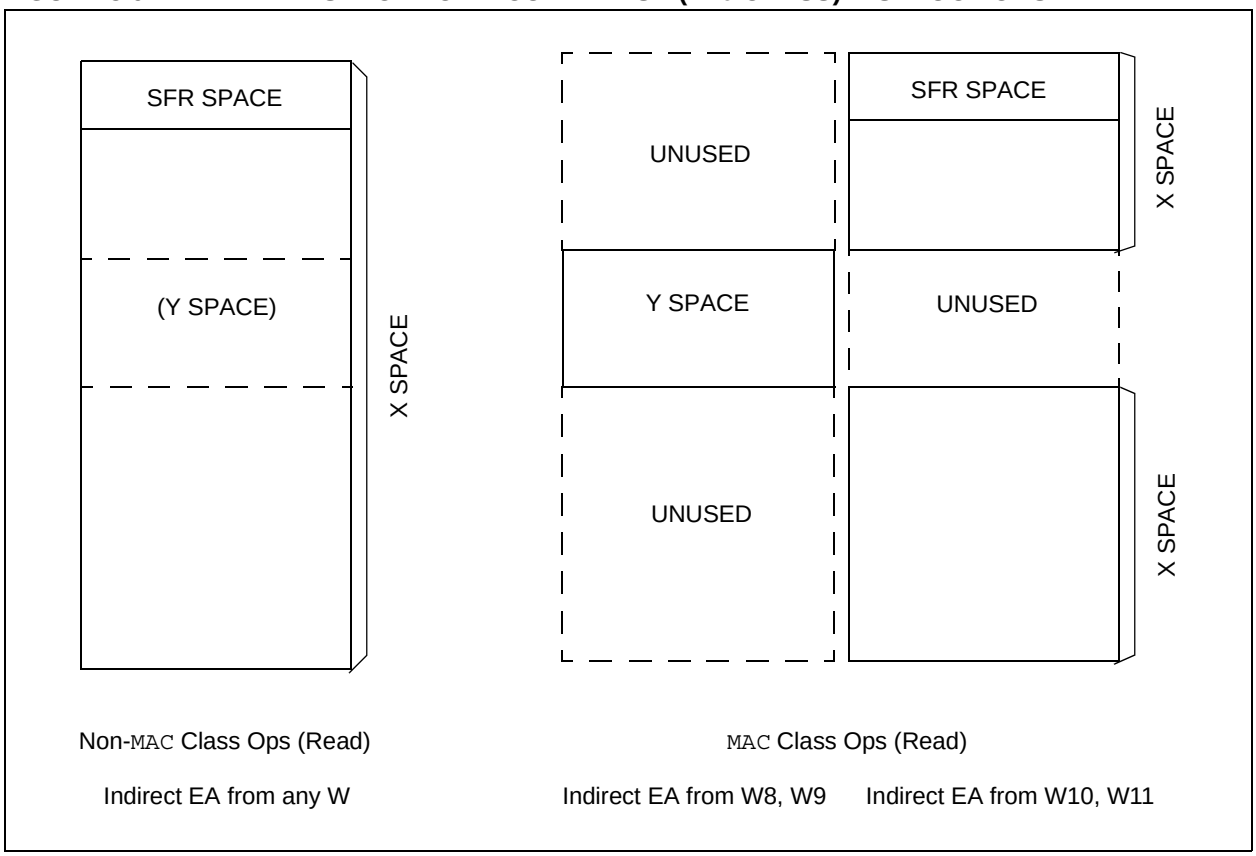


TABLE 3-2: EFFECT OF INVALID MEMORY ACCESSES

Attempted Operation	Data Returned
EA = an unimplemented address	0x0000
W8 or W9 used to access Y data space in a MAC instruction	0x0000
W10 or W11 used to access X data space in a MAC instruction	0x0000

All effective addresses are 16 bits wide and point to bytes within the data space. Therefore, the data space address range is 64 Kbytes or 32K words.

3.2.3 DATA SPACE WIDTH

The core data width is 16 bits. All internal registers are organized as 16-bit wide words. Data space memory is organized in byte addressable, 16-bit wide blocks.

3.2.4 DATA ALIGNMENT

To help maintain backward compatibility with PIC® MCU devices and improve data space memory usage efficiency, the dsPIC30F instruction set supports both word and byte operations. Data is aligned in data memory and registers as words, but all data space EAs resolve to bytes. Data byte reads will read the complete word which contains the byte, using the LSb of any EA to determine which byte to select. The selected byte is placed onto the LSB of the X data path (no byte accesses are possible from the Y data path as the MAC class of instruction can only fetch words). That is, data memory and registers are organized as two parallel byte wide entities with shared (word) address decode but separate write lines. Data byte writes only write to the corresponding side of the array or register which matches the byte address.

As a consequence of this byte accessibility, all effective address calculations (including those generated by the DSP operations which are restricted to word sized data) are internally scaled to step through word aligned memory. For example, the core would recognize that Post-Modified Register Indirect Addressing mode [Ws++] will result in a value of Ws + 1 for byte operations and Ws + 2 for word operations.

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
```

NOTES:

7.2 Erasing Data EEPROM

7.2.1 ERASING A BLOCK OF DATA EEPROM

In order to erase a block of data EEPROM, the NVMADRU and NVMADR registers must initially point to the block of memory to be erased. Configure NVMCON for erasing a block of data EEPROM, and set the ERASE and WREN bits in the NVMCON register. Setting the WR bit initiates the erase as shown in Example 7-2.

EXAMPLE 7-2: DATA EEPROM BLOCK ERASE

```
; Select data EEPROM block, ERASE, WREN bits
    MOV     #0x4045,W0
    MOV     W0,NVMCON                 ; Initialize NVMCON SFR

; Start erase cycle by setting WR after writing key 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                ; Initiate erase sequence
    NOP
    NOP
; Erase cycle will complete in 2mS. CPU is not stalled for the Data Erase Cycle
; User can poll WR bit, use NVMIF or Timer IRQ to determine erasure complete
```

7.2.2 ERASING A WORD OF DATA EEPROM

The NVMADRU and NVMADR registers must point to the block. Select erase a block of data Flash, and set the ERASE and WREN bits in the NVMCON register. Setting the WR bit initiates the erase as shown in Example 7-3.

EXAMPLE 7-3: DATA EEPROM WORD ERASE

```
; Select data EEPROM word, ERASE, WREN bits
    MOV     #0x4044,W0
    MOV     W0,NVMCON

; Start erase cycle by setting WR after writing key 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                  ; Initiate erase sequence
    NOP
    NOP
; Erase cycle will complete in 2mS. CPU is not stalled for the Data Erase Cycle
; User can poll WR bit, use NVMIF or Timer IRQ to determine erasure complete
```

TABLE 17-1: CAN1 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
C1RXF0SID	0300	—	—	—	Receive Acceptance Filter 0 Standard Identifier <10:0>											—	EXIDE	000u uuuu uuuu uu0u
C1RXF0EIDH	0302	—	—	—	—	Receive Acceptance Filter 0 Extended Identifier <17:6>												0000 uuuu uuuu uuuu
C1RXF0EIDL	0304	Receive Acceptance Filter 0 Extended Identifier <5:0>						—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXF1SID	0308	—	—	—	Receive Acceptance Filter 1 Standard Identifier <10:0>											—	EXIDE	000u uuuu uuuu uu0u
C1RXF1EIDH	030A	—	—	—	—	Receive Acceptance Filter 1 Extended Identifier <17:6>												0000 uuuu uuuu uuuu
C1RXF1EIDL	030C	Receive Acceptance Filter 1 Extended Identifier <5:0>						—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXF2SID	0310	—	—	—	Receive Acceptance Filter 2 Standard Identifier <10:0>											—	EXIDE	000u uuuu uuuu uu0u
C1RXF2EIDH	0312	—	—	—	—	Receive Acceptance Filter 2 Extended Identifier <17:6>												0000 uuuu uuuu uuuu
C1RXF2EIDL	0314	Receive Acceptance Filter 2 Extended Identifier <5:0>						—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXF3SID	0318	—	—	—	Receive Acceptance Filter 3 Standard Identifier <10:0>											—	EXIDE	000u uuuu uuuu uu0u
C1RXF3EIDH	031A	—	—	—	—	Receive Acceptance Filter 3 Extended Identifier <17:6>												0000 uuuu uuuu uuuu
C1RXF3EIDL	031C	Receive Acceptance Filter 3 Extended Identifier <5:0>						—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXF4SID	0320	—	—	—	Receive Acceptance Filter 4 Standard Identifier <10:0>											—	EXIDE	000u uuuu uuuu uu0u
C1RXF4EIDH	0322	—	—	—	—	Receive Acceptance Filter 4 Extended Identifier <17:6>												0000 uuuu uuuu uuuu
C1RXF4EIDL	0324	Receive Acceptance Filter 4 Extended Identifier <5:0>						—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXF5SID	0328	—	—	—	Receive Acceptance Filter 5 Standard Identifier <10:0>											—	EXIDE	000u uuuu uuuu uu0u
C1RXF5EIDH	032A	—	—	—	—	Receive Acceptance Filter 5 Extended Identifier <17:6>												0000 uuuu uuuu uuuu
C1RXF5EIDL	032C	Receive Acceptance Filter 5 Extended Identifier <5:0>						—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXM0SID	0330	—	—	—	Receive Acceptance Mask 0 Standard Identifier <10:0>											—	MIDE	000u uuuu uuuu uu0u
C1RXM0EIDH	0332	—	—	—	—	Receive Acceptance Mask 0 Extended Identifier <17:6>												0000 uuuu uuuu uuuu
C1RXM0EIDL	0334	Receive Acceptance Mask 0 Extended Identifier <5:0>						—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXM1SID	0338	—	—	—	Receive Acceptance Mask 1 Standard Identifier <10:0>											—	MIDE	000u uuuu uuuu uu0u
C1RXM1EIDH	033A	—	—	—	—	Receive Acceptance Mask 1 Extended Identifier <17:6>												0000 uuuu uuuu uuuu
C1RXM1EIDL	033C	Receive Acceptance Mask 1 Extended Identifier <5:0>						—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1TX2SID	0340	Transmit Buffer 2 Standard Identifier <10:6>					—	—	—	Transmit Buffer 2 Standard Identifier <5:0>						SRR	TXIDE	uuuu u000 uuuu uuuu
C1TX2EID	0342	Transmit Buffer 2 Extended Identifier <17:14>				—	—	—	—	Transmit Buffer 2 Extended Identifier <13:6>								uuuu 0000 uuuu uuuu
C1TX2DLC	0344	Transmit Buffer 2 Extended Identifier <5:0>						TXRTR	TXRB1	TXRB0	DLC<3:0>				—	—	—	uuuu uuuu uuuu u000
C1TX2B1	0346	Transmit Buffer 2 Byte 1								Transmit Buffer 2 Byte 0								uuuu uuuu uuuu uuuu
C1TX2B2	0348	Transmit Buffer 2 Byte 3								Transmit Buffer 2 Byte 2								uuuu uuuu uuuu uuuu
C1TX2B3	034A	Transmit Buffer 2 Byte 5								Transmit Buffer 2 Byte 4								uuuu uuuu uuuu uuuu
C1TX2B4	034C	Transmit Buffer 2 Byte 7								Transmit Buffer 2 Byte 6								uuuu uuuu uuuu uuuu
C1TX2CON	034E	—	—	—	—	—	—	—	—	—	TXABT	TXLARB	TXERR	TXREQ	—	TXPRI<1:0>		0000 0000 0000 0000
C1TX1SID	0350	Transmit Buffer 1 Standard Identifier <10:6>					—	—	—	Transmit Buffer 1 Standard Identifier <5:0>						SRR	TXIDE	uuuu u000 uuuu uuuu

Legend: u = uninitialized bit

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

TABLE 17-1: CAN1 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
C1INTE	0398	—	—	—	—	—	—	—	—	IVRIE	WAKIE	ERRIE	TX2IE	TX1IE	TX0IE	RX1E	RX0IE	0000 0000 0000 0000
C1EC	039A	Transmit Error Count Register								Receive Error Count Register								0000 0000 0000 0000

Legend: u = uninitialized bit

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

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.

18.0 DATA CONVERTER INTERFACE (DCI) 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).

18.1 Module Introduction

The dsPIC30F Data Converter Interface (DCI) module allows simple interfacing of devices, such as audio coder/decoders (Codecs), A/D converters and D/A converters. The following interfaces are supported:

- Framed Synchronous Serial Transfer (Single or Multi-Channel)
- Inter-IC Sound (I^2S) Interface
- AC-Link Compliant mode

The DCI module provides the following general features:

- Programmable word size up to 16 bits
- Support for up to 16 time slots, for a maximum frame size of 256 bits
- Data buffering for up to 4 samples without CPU overhead

18.2 Module I/O Pins

There are four I/O pins associated with the module. When enabled, the module controls the data direction of each of the four pins.

18.2.1 CSCK PIN

The CSCK pin provides the serial clock for the DCI module. The CSCK pin may be configured as an input or output using the CSCKD control bit in the DCICON1 SFR. When configured as an output, the serial clock is provided by the dsPIC30F. When configured as an input, the serial clock must be provided by an external device.

18.2.2 CSDO PIN

The serial data output (CSDO) pin is configured as an output only pin when the module is enabled. The CSDO pin drives the serial bus whenever data is to be transmitted. The CSDO pin is tri-stated or driven to '0' during CSCK periods when data is not transmitted, depending on the state of the CSDOM control bit. This allows other devices to place data on the serial bus during transmission periods not used by the DCI module.

18.2.3 CSDI PIN

The serial data input (CSDI) pin is configured as an input only pin when the module is enabled.

18.2.3.1 COFS PIN

The Codec frame synchronization (COFS) pin is used to synchronize data transfers that occur on the CSDO and CSDI pins. The COFS pin may be configured as an input or an output. The data direction for the COFS pin is determined by the COFSD control bit in the DCICON1 register.

The DCI module accesses the shadow registers while the CPU is in the process of accessing the memory mapped buffer registers.

18.2.4 BUFFER DATA ALIGNMENT

Data values are always stored left justified in the buffers since most Codec data is represented as a signed 2's complement fractional number. If the received word length is less than 16 bits, the unused LSbs in the receive buffer registers are set to '0' by the module. If the transmitted word length is less than 16 bits, the unused LSbs in the transmit buffer register are ignored by the module. The word length setup is described in subsequent sections of this document.

18.2.5 TRANSMIT/RECEIVE SHIFT REGISTER

The DCI module has a 16-bit shift register for shifting serial data in and out of the module. Data is shifted in/out of the shift register MSb first, since audio PCM data is transmitted in signed 2's complement format.

18.2.6 DCI BUFFER CONTROL

The DCI module contains a buffer control unit for transferring data between the shadow buffer memory and the serial shift register. The buffer control unit is a simple 2-bit address counter that points to word locations in the shadow buffer memory. For the receive memory space (high address portion of DCI buffer memory), the address counter is concatenated with a '0' in the MSb location to form a 3-bit address. For the transmit memory space (high portion of DCI buffer memory), the address counter is concatenated with a '1' in the MSb location.

Note: The DCI buffer control unit always accesses the same relative location in the transmit and receive buffers, so only one address counter is provided.

19.0 12-BIT ANALOG-TO-DIGITAL CONVERTER (A/D) 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).

The 12-bit Analog-to-Digital Converter (ADC) allows conversion of an analog input signal to a 12-bit digital number. This module is based on a Successive Approximation Register (SAR) architecture and provides a maximum sampling rate of 200 ksps. The ADC module has up to 16 analog inputs which are multiplexed into a sample and hold amplifier. The output of the sample and hold is the input into the converter which generates the result. The analog reference voltage is software selectable to either the device supply voltage (AVDD/AVSS) or the voltage level on the (VREF+/VREF-) pin. The ADC has a unique feature of being able to operate while the device is in Sleep mode with RC oscillator selection.

The ADC module has six 16-bit registers:

- ADC Control Register 1 (ADCON1)
- ADC Control Register 2 (ADCON2)
- ADC Control Register 3 (ADCON3)
- ADC Input Select Register (ADCHS)
- ADC Port Configuration Register (ADPCFG)
- ADC Input Scan Selection Register (ADCSSL)

The ADCON1, ADCON2 and ADCON3 registers control the operation of the A/D module. The ADCHS register selects the input channels to be converted. The ADPCFG register configures the port pins as analog inputs or as digital I/O. The ADCSSL register selects inputs for scanning.

Note: The SSRC<2:0>, ASAM, SMPI<3:0>, BUFM and ALTS bits, as well as the ADCON3 and ADCSSL registers, must not be written to while ADON = 1. This would lead to indeterminate results.

The block diagram of the 12-bit ADC module is shown in Figure 19-1.

FIGURE 19-1: 12-BIT ADC FUNCTIONAL BLOCK DIAGRAM

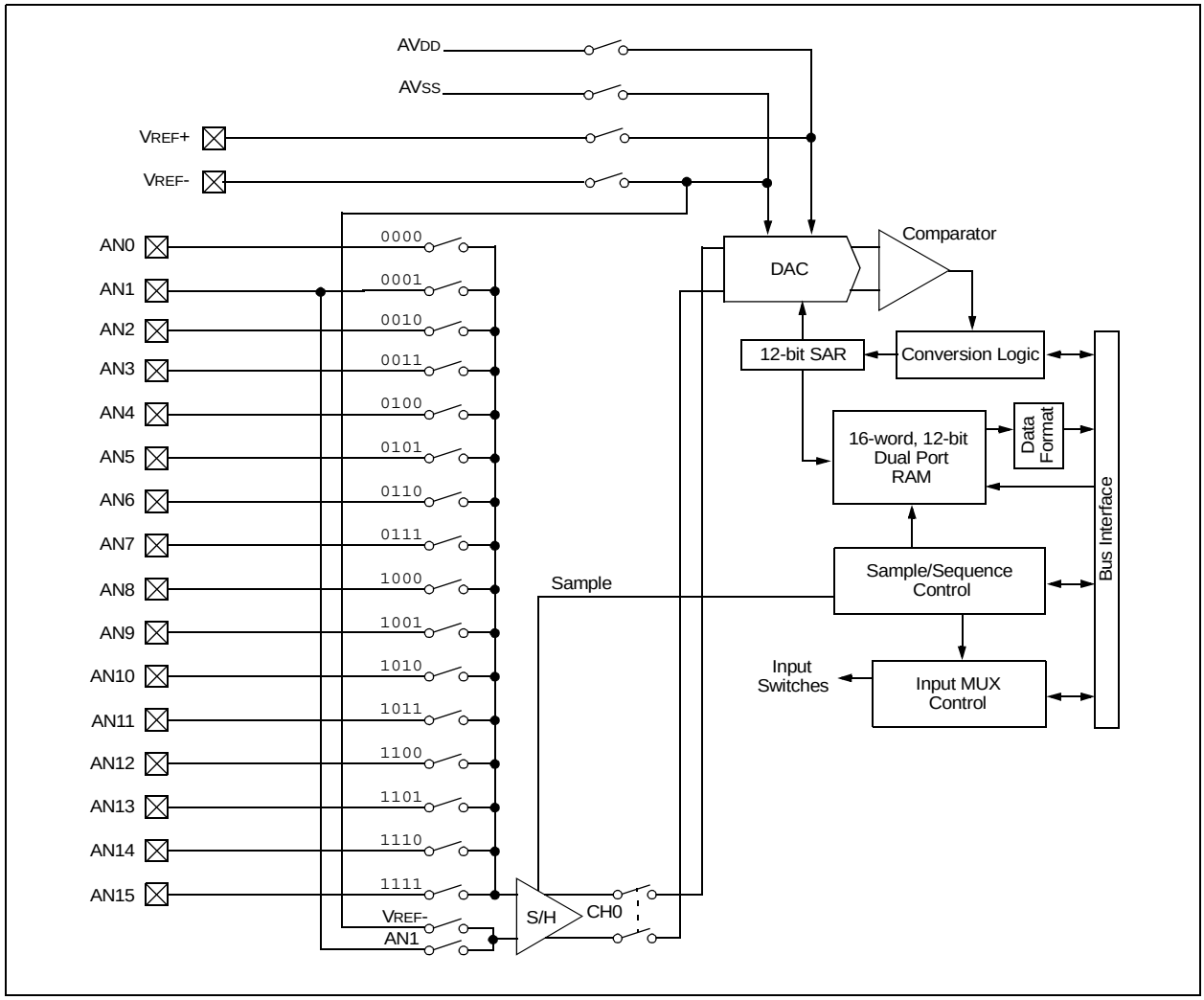


TABLE 21-1: SYMBOLS USED IN OPCODE DESCRIPTIONS (CONTINUED)

Field	Description
Wb	Base W register ∈ {W0..W15}
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 ∈ {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 ∈ {W0..W15}
Wnd	One of 16 destination working registers ∈ {W0..W15}
Wns	One of 16 source working registers ∈ {W0..W15}
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 ∈ {W4..W7}
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 ∈ {W4..W7}

TABLE 21-2: INSTRUCTION SET OVERVIEW (CONTINUED)

Base Instr #	Assembly Mnemonic	Assembly Syntax	Description	# of Words	# of Cycles	Status Flags Affected
48	MPY	MPY Wm*Wn,Acc,Wx,Wxd,Wy,Wyd	Multiply Wm by Wn to Accumulator	1	1	OA,OB,OAB, SA,SB,SAB
		MPY Wm*Wm,Acc,Wx,Wxd,Wy,Wyd	Square Wm to Accumulator	1	1	OA,OB,OAB, SA,SB,SAB
49	MPY.N	MPY.N Wm*Wn,Acc,Wx,Wxd,Wy,Wyd	-(Multiply Wm by Wn) to Accumulator	1	1	None
50	MSC	MSC Wm*Wm,Acc,Wx,Wxd,Wy,Wyd, AWB	Multiply and Subtract from Accumulator	1	1	OA,OB,OAB, SA,SB,SAB
51	MUL	MUL.SS Wb,Ws,Wnd	{Wnd+1, Wnd} = signed(Wb) * signed(Ws)	1	1	None
		MUL.SU Wb,Ws,Wnd	{Wnd+1, Wnd} = signed(Wb) * unsigned(Ws)	1	1	None
		MUL.US Wb,Ws,Wnd	{Wnd+1, Wnd} = unsigned(Wb) * signed(Ws)	1	1	None
		MUL.UU Wb,Ws,Wnd	{Wnd+1, Wnd} = unsigned(Wb) * unsigned(Ws)	1	1	None
		MUL.SU Wb,#lit5,Wnd	{Wnd+1, Wnd} = signed(Wb) * unsigned(lit5)	1	1	None
		MUL.UU Wb,#lit5,Wnd	{Wnd+1, Wnd} = unsigned(Wb) * unsigned(lit5)	1	1	None
		MUL f	W3:W2 = f * WREG	1	1	None
52	NEG	NEG Acc	Negate Accumulator	1	1	OA,OB,OAB, SA,SB,SAB
		NEG f	f = $\bar{f}$ + 1	1	1	C,DC,N,OV,Z
		NEG f,WREG	WREG = $\bar{f}$ + 1	1	1	C,DC,N,OV,Z
		NEG Ws,Wd	Wd = $\overline{Ws}$ + 1	1	1	C,DC,N,OV,Z
53	NOP	NOP	No Operation	1	1	None
		NOPR	No Operation	1	1	None
54	POP	POP f	Pop f from top-of-stack (TOS)	1	1	None
		POP Wdo	Pop from top-of-stack (TOS) to Wdo	1	1	None
		POP.D Wnd	Pop from top-of-stack (TOS) to W(nd):W(nd+1)	1	2	None
		POP.S	Pop Shadow Registers	1	1	All
55	PUSH	PUSH f	Push f to top-of-stack (TOS)	1	1	None
		PUSH Wso	Push Wso to top-of-stack (TOS)	1	1	None
		PUSH.D Wns	Push W(ns):W(ns+1) to top-of-stack (TOS)	1	2	None
		PUSH.S	Push Shadow Registers	1	1	None
56	PWRSAV	PWRSAV #lit1	Go into Sleep or Idle mode	1	1	WDTO,Sleep
57	RCALL	RCALL Expr	Relative Call	1	2	None
		RCALL Wn	Computed Call	1	2	None
58	REPEAT	REPEAT #lit14	Repeat Next Instruction lit14+1 times	1	1	None
		REPEAT Wn	Repeat Next Instruction (Wn)+1 times	1	1	None
59	RESET	RESET	Software device Reset	1	1	None
60	RETFIE	RETFIE	Return from interrupt	1	3 (2)	None
61	RETLW	RETLW #lit10,Wn	Return with literal in Wn	1	3 (2)	None
62	RETURN	RETURN	Return from Subroutine	1	3 (2)	None
63	RLC	RLC f	f = Rotate Left through Carry f	1	1	C,N,Z
		RLC f,WREG	WREG = Rotate Left through Carry f	1	1	C,N,Z
		RLC Ws,Wd	Wd = Rotate Left through Carry Ws	1	1	C,N,Z
64	RLNC	RLNC f	f = Rotate Left (No Carry) f	1	1	N,Z
		RLNC f,WREG	WREG = Rotate Left (No Carry) f	1	1	N,Z
		RLNC Ws,Wd	Wd = Rotate Left (No Carry) Ws	1	1	N,Z
65	RRC	RRC f	f = Rotate Right through Carry f	1	1	C,N,Z
		RRC f,WREG	WREG = Rotate Right through Carry f	1	1	C,N,Z
		RRC Ws,Wd	Wd = Rotate Right through Carry Ws	1	1	C,N,Z

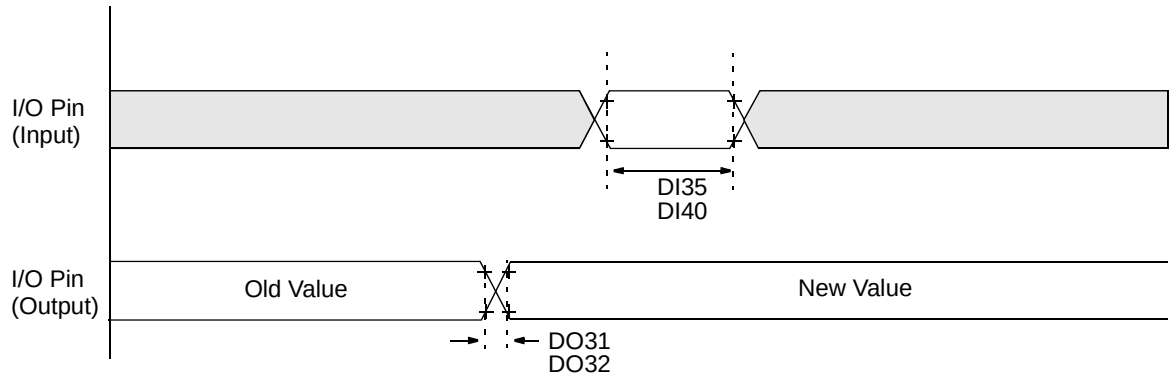
TABLE 23-5: DC CHARACTERISTICS: OPERATING CURRENT (I_{DD})

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			
Parameter No.	**Typical(1)**	**Max**	**Units**	**Conditions**		
Operating Current (I_{DD})(2)						
DC31a	6.8	10	mA	25°C	3.3V	0.128 MIPS LPRC (512 kHz)
DC31b	6.3	10	mA	85°C		
DC31c	6.1	10	mA	125°C		
DC31e	16	22	mA	25°C	5V	
DC31f	15	22	mA	85°C		
DC31g	15	22	mA	125°C		
DC30a	13	19	mA	25°C	3.3V	(1.8 MIPS) FRC (7.37 MHz)
DC30b	13	19	mA	85°C		
DC30c	13	19	mA	125°C		
DC30e	27	39	mA	25°C	5V	
DC30f	26	39	mA	85°C		
DC30g	25	39	mA	125°C		
DC23a	27	41	mA	25°C	3.3V	4 MIPS
DC23b	27	41	mA	85°C		
DC23c	27	41	mA	125°C		
DC23e	41	60	mA	25°C	5V	
DC23f	40	60	mA	85°C		
DC23g	40	60	mA	125°C		
DC24a	46	71	mA	25°C	3.3V	10 MIPS
DC24b	46	71	mA	85°C		
DC24c	47	71	mA	125°C		
DC24e	79	120	mA	25°C	5V	
DC24f	78	120	mA	85°C		
DC24g	78	120	mA	125°C		
DC27a	83	120	mA	25°C	3.3V	20 MIPS
DC27b	83	120	mA	85°C		
DC27d	138	190	mA	25°C	5V	
DC27e	137	190	mA	85°C		
DC27f	136	190	mA	125°C		
DC29a	194	255	mA	25°C	5V	30 MIPS
DC29b	192	255	mA	85°C		

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

2: The supply current is mainly a function of the operating voltage and frequency. Other factors such as I/O pin loading and switching rate, oscillator type, internal code execution pattern and temperature also have an impact on the current consumption. The test conditions for all I_{DD} measurements are as follows: OSC1 driven with external square wave from rail to rail. All I/O pins are configured as Inputs and pulled to V_{DD}. $\overline{MCLR}$ = V_{DD}, WDT, FSCM, LVD and BOR are disabled. CPU, SRAM, Program Memory and Data Memory are operational. No peripheral modules are operating.

FIGURE 23-5: CLKO AND I/O TIMING CHARACTERISTICS



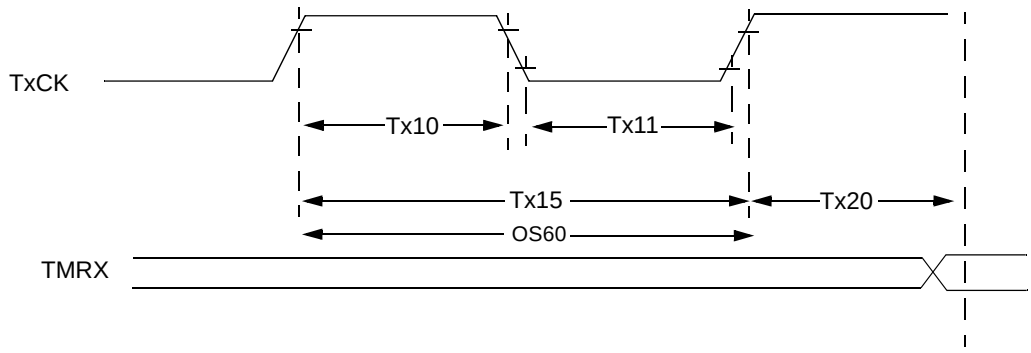
Note: Refer to Figure 23-3 for load conditions.

TABLE 23-20: CLKO AND I/O 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)(2)(3)**	**Min**	**Typ(4)**	**Max**	**Units**	**Conditions**
DO31	TIOR	Port output rise time	—	7	20	ns	—
DO32	TIOF	Port output fall time	—	7	20	ns	—
DI35	TINP	INTx pin high or low time (output)	20	—	—	ns	—
DI40	TRBP	CNx high or low time (input)	2 TCY	—	—	ns	—

Note 1: These parameters are asynchronous events not related to any internal clock edges
2: Measurements are taken in RC mode and EC mode where CLKO output is 4 x TOSC.
3: These parameters are characterized but not tested in manufacturing.
4: Data in “Typ” column is at 5V, 25°C unless otherwise stated.

FIGURE 23-8: TYPE A, B AND C TIMER EXTERNAL CLOCK TIMING CHARACTERISTICS



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

TABLE 23-23: TYPE A TIMER (TIMER1) EXTERNAL CLOCK TIMING REQUIREMENTS(1)

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**	**Typ**	**Max**	**Units**	**Conditions**
TA10	TTXH	TxCK High Time	Synchronous, no prescaler	0.5 TCY + 20	—	—	ns	Must also meet parameter TA15
			Synchronous, with prescaler	10	—	—	ns	
			Asynchronous	10	—	—	ns	
TA11	TTXL	TxCK Low Time	Synchronous, no prescaler	0.5 TCY + 20	—	—	ns	Must also meet parameter TA15
			Synchronous, with prescaler	10	—	—	ns	
			Asynchronous	10	—	—	ns	
TA15	TTXP	TxCK Input Period	Synchronous, no prescaler	TCY + 10	—	—	ns	
			Synchronous, with prescaler	Greater of: 20 ns or (TCY + 40)/N	—	—	—	N = prescale value (1, 8, 64, 256)
			Asynchronous	20	—	—	ns	
OS60	Ft1	SOSC1/T1CK oscillator input frequency range (oscillator enabled by setting bit TCS (T1CON, bit 1))		DC	—	50	kHz	
TA20	TCKEXTMRL	Delay from External TxCK Clock Edge to Timer Increment		0.5 TCY		1.5 TCY	—	

Note 1: Timer1 is a Type A.

TABLE 23-29: DCI MODULE (MULTICHANNEL, I^2S MODES) 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**
CS10	TCSCKL	CSCK Input Low Time (CSCK pin is an input)	TCY/2 + 20	—	—	ns	—
		CSCK Output Low Time(3) (CSCK pin is an output)	30	—	—	ns	—
CS11	TCSCKH	CSCK Input High Time (CSCK pin is an input)	TCY/2 + 20	—	—	ns	—
		CSCK Output High Time(3) (CSCK pin is an output)	30	—	—	ns	—
CS20	TCSCKF	CSCK Output Fall Time(4) (CSCK pin is an output)	—	10	25	ns	—
CS21	TCSCKR	CSCK Output Rise Time(4) (CSCK pin is an output)	—	10	25	ns	—
CS30	TCSDOF	CSDO Data Output Fall Time(4)	—	10	25	ns	—
CS31	TCSDOR	CSDO Data Output Rise Time(4)	—	10	25	ns	—
CS35	TDV	Clock edge to CSDO data valid	—	—	10	ns	—
CS36	TDIV	Clock edge to CSDO tri-stated	10	—	20	ns	—
CS40	TCSDI	Setup time of CSDI data input to CSCK edge (CSCK pin is input or output)	20	—	—	ns	—
CS41	THCSDI	Hold time of CSDI data input to CSCK edge (CSCK pin is input or output)	20	—	—	ns	—
CS50	TCOFSF(1)	COFS Fall Time (COFS pin is output)	—	10	25	ns	—
CS51	TCOFSR(1)	COFS Rise Time (COFS pin is output)	—	10	25	ns	—
CS55	TSCOFS	Setup time of COFS data input to CSCK edge (COFS pin is input)	20	—	—	ns	—
CS56	THCOFS	Hold time of COFS data input to CSCK edge (COFS pin is input)	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.

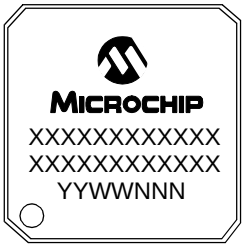
3: The minimum clock period for CSCK is 100 ns. Therefore, the clock generated in Master mode must not violate this specification.

4: Assumes 50 pF load on all DCI pins.

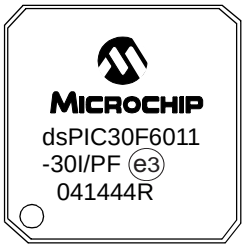
24.0 PACKAGING INFORMATION

24.1 Package Marking Information

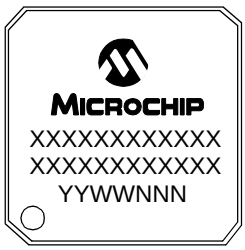
64-Lead TQFP (14x14x1mm)



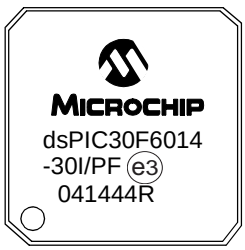
Example



80-Lead TQFP (14x14x1mm)



Example

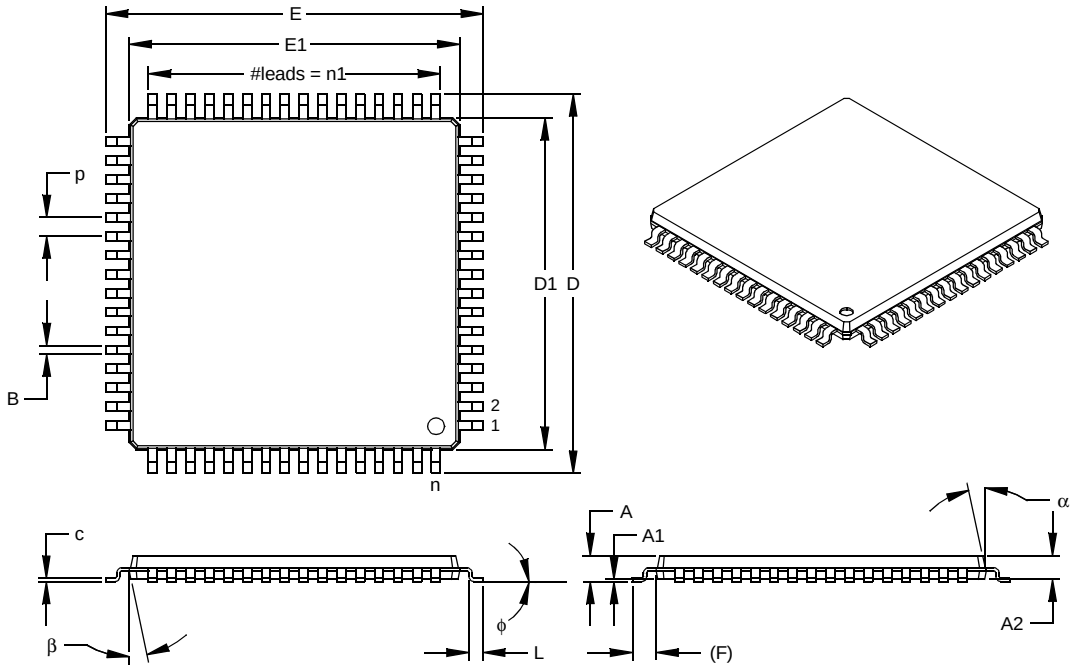


Legend:	XX...X	Customer-specific information
	Y	Year code (last digit of calendar year)
	YY	Year code (last 2 digits of calendar year)
	WW	Week code (week of January 1 is week ‘01’)
	NNN	Alphanumeric traceability code
	(e3)	Pb-free JEDEC designator for Matte Tin (Sn)
	*	This package is Pb-free. The Pb-free JEDEC designator ((e3)) can be found on the outer packaging for this package.

Note: In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line, thus limiting the number of available characters for customer-specific information.

64-Lead Plastic Thin Quad Flatpack (PF) 14x14x1 mm Body, 1.0/0.10 mm Lead Form (TQFP)

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



Units		INCHES			MILLIMETERS*		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		64			64	
Pitch	p		.031 BSC			0.80 BSC	
Pins per Side	n1		16			16	
Overall Height	A			.047			1.20
Molded Package Thickness	A2	.037	.039	.041	0.95	1.00	1.05
Standoff	A1	.002		.006	0.05		0.15
Foot Length	L	.018	.024	.030	0.45	0.60	0.75
Footprint	(F)		.039 REF			1.00 REF	
Foot Angle	ϕ	0	3.5	7	0	3.5	7
Overall Width	E		.630 BSC			16.00 BSC	
Overall Length	D		.630 BSC			16.00 BSC	
Molded Package Width	E1		.551 BSC			14.00 BSC	
Molded Package Length	D1		.551 BSC			14.00 BSC	
Lead Thickness	c	.004		.008	0.09		0.20
Lead Width	B	.012	.015	.018	0.30	0.37	0.45
Mold Draft Angle Top	α	11	12	13	11	12	13
Mold Draft Angle Bottom	β	11	12	13	11	12	13

***** Controlling Parameter

Notes:

Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" (0.254mm) per side.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.
See ASME Y14.5M

JEDEC Equivalent: MS-026

Drawing No. C04-066

Revised 7-20-06