# Microchip Technology - DSPIC30F6013A-20E/PF Datasheet

**Welcome to E-XFL.COM**

## What is "Embedded - Microcontrollers"?

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

## Applications of "Embedded - Microcontrollers"

| Details | |
|---|---|
| Product Status | Active |
| Core Processor | dsPIC |
| Core Size | 16-Bit |
| Speed | 20 MIPS |
| Connectivity | CANbus, I²C, SPI, UART/USART |
| Peripherals | Brown-out Detect/Reset, LVD, POR, PWM, WDT |
| Number of I/O | 68 |
| Program Memory Size | 132KB (44K x 24) |
| Program Memory Type | FLASH |
| EEPROM Size | 2K x 8 |
| RAM Size | 6K x 8 |
| Voltage - Supply (Vcc/Vdd) | 2.5V ~ 5.5V |
| Data Converters | A/D 16x12b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 125°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/dspic30f6013a-20e-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**



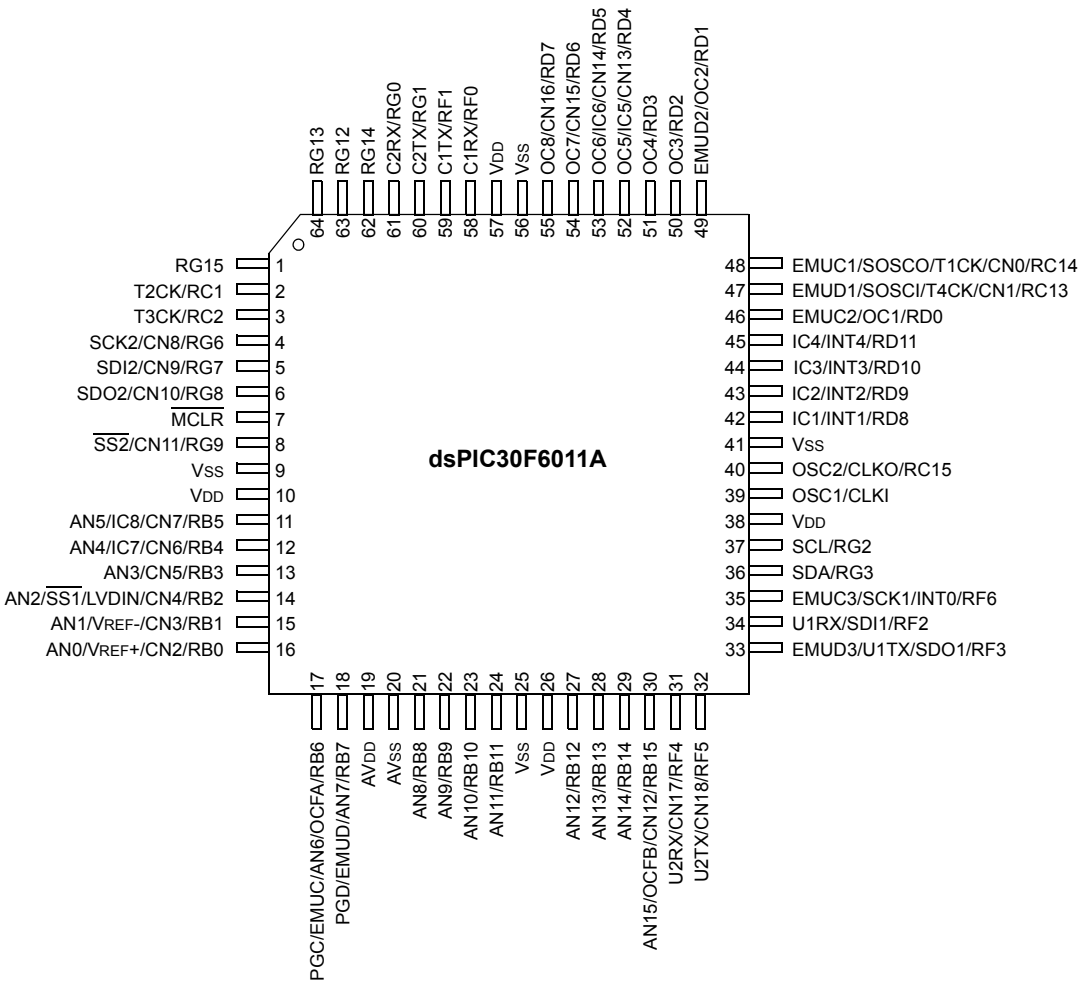
ISBN: 978-1-60932-883-2

*Microchip received ISO/TS-16949:2002 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, 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.*

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

## Pin Diagrams

**64-Pin TQFP**



## Table of Contents



## TO OUR VALUED CUSTOMERS

It is our intention to provide our valued customers with the best documentation possible to ensure successful use of your Microchip products. To this end, we will continue to improve our publications to better suit your needs. Our publications will be refined and enhanced as new volumes and updates are introduced.

If you have any questions or comments regarding this publication, please contact the Marketing Communications Department via E-mail at **docerrors@microchip.com** or fax the **Reader Response Form** in the back of this data sheet to (480) 792-4150. We welcome your feedback.

### Most Current Data Sheet

To obtain the most up-to-date version of this data sheet, please register at our Worldwide Web site at:

**http://www.microchip.com**

You can determine the version of a data sheet by examining its literature number found on the bottom outside corner of any page. The last character of the literature number is the version number, (e.g., DS30000A is version A of document DS30000).

### Errata

An errata sheet, describing minor operational differences from the data sheet and recommended workarounds, may exist for current devices. As device/documentation issues become known to us, we will publish an errata sheet. The errata will specify the revision of silicon and revision of document to which it applies.

To determine if an errata sheet exists for a particular device, please check with one of the following:

- Microchip's Worldwide Web site; **http://www.microchip.com**
- Your local Microchip sales office (see last page)

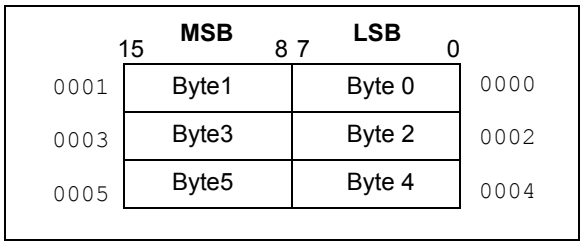
When contacting a sales office, please specify which device, revision of silicon and data sheet (include literature number) you are using.

### Customer Notification System

Register on our web site at **www.microchip.com** to receive the most current information on all of our products.

All word accesses must be aligned to an even address. Misaligned word data fetches are not supported so care must be taken when mixing byte and word operations, or translating from 8-bit MCU code. Should a misaligned read or write be attempted, an address error trap will be generated. If the error occurred on a read, the instruction underway is completed, whereas if it occurred on a write, the instruction will be executed but the write will not occur. In either case, a trap will then be executed, allowing the system and/or user to examine the machine state prior to execution of the address fault.

**FIGURE 3-10: DATA ALIGNMENT**




All byte loads into any W register are loaded into the LSB. The MSB is not modified.

A sign-extend (SE) instruction is provided to allow users to translate 8-bit signed data to 16-bit signed values. Alternatively, for 16-bit unsigned data, users can clear the MSB of any W register by executing a zero-extend (ZE) instruction on the appropriate address.

Although most instructions are capable of operating on word or byte data sizes, it should be noted that some instructions, including the DSP instructions, operate only on words.

### 3.2.5 NEAR DATA SPACE

An 8 Kbyte 'near' data space is reserved in X address memory space between 0x0000 and 0x1FFF, which is directly addressable via a 13-bit absolute address field within all memory direct instructions. The remaining X address space and all of the Y address space is addressable indirectly. Additionally, the whole of X data space is addressable using MOV instructions, which support memory direct addressing with a 16-bit address field.

### 3.2.6 SOFTWARE STACK

The dsPIC DSC devices contain a software stack. W15 is used as the Stack Pointer.

The Stack Pointer always points to the first available free word and grows from lower addresses towards higher addresses. It pre-decrements for stack pops and post-increments for stack pushes as shown in Figure 3-11. Note that for a PC push during any CALL instruction, the MSB of the PC is zero-extended before the push, ensuring that the MSB is always clear.

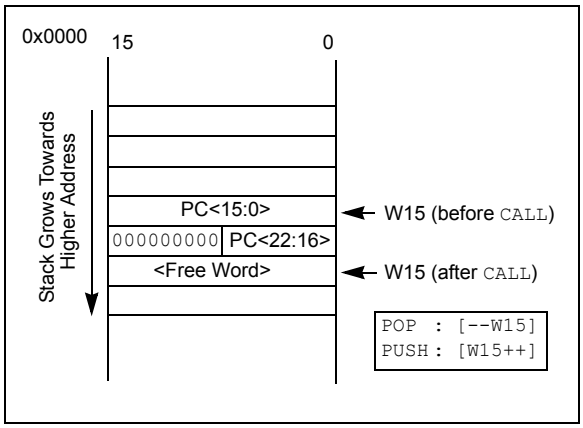
| **Note:** | A PC push during exception processing will concatenate the SRL register to the MSB of the PC prior to the push. |
|---|---|

There is a Stack Pointer Limit register (SPLIM) associated with the Stack Pointer. SPLIM is uninitialized at Reset. As is the case for the Stack Pointer, SPLIM<0> is forced to '0' because all stack operations must be word aligned. Whenever an Effective Address (EA) is generated using W15 as a source or destination pointer, the address thus generated is compared with the value in SPLIM. If the contents of the Stack Pointer (W15) and the SPLIM register are equal and a push operation is performed, a Stack Error Trap will not occur. The Stack Error Trap will occur on a subsequent push operation. Thus, for example, if it is desirable to cause a Stack Error Trap when the stack grows beyond address 0x2000 in RAM, initialize the SPLIM with the value, 0x1FFE.

Similarly, a Stack Pointer underflow (stack error) trap is generated when the Stack Pointer address is found to be less than 0x0800, thus preventing the stack from interfering with the Special Function Register (SFR) space.

A write to the SPLIM register should not be immediately followed by an indirect read operation using W15.

**FIGURE 3-11: CALL STACK FRAME**



### 3.2.7 DATA RAM PROTECTION FEATURE

The dsPIC30F6011A/6012A/6013A/6014A devices support data RAM protection features which enable segments of RAM to be protected when used in conjunction with Boot and Secure Code Segment Security. BSRAM (Secure RAM segment for BS) is accessible only from the Boot Segment Flash code when enabled. SSRAM (Secure RAM segment for RAM) is accessible only from the Secure Segment Flash code when enabled.

See Table 3-3 for an overview of the BSRAM and SSRAM SFRs.

**NOTES:**

## 7.0 DATA EEPROM MEMORY

**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 Data EEPROM Memory is readable and writable during normal operation over the entire VDD range. The data EEPROM memory is directly mapped in the program memory address space.

The four SFRs used to read and write the program Flash memory are used to access data EEPROM memory, as well. As described in **Section 6.5 "Control Registers"**, these registers are:

- NVMCON
- NVMADR
- NVMADRU
- NVMKEY

The EEPROM data memory allows read and write of single words and 16-word blocks. When interfacing to data memory, NVMADR in conjunction with the NVMADRU register are used to address the EEPROM location being accessed. `TBLRDL` and `TBLWTL` instructions are used to read and write data EEPROM. The dsPIC30F devices have up to 8 Kbytes (4K words) of data EEPROM with an address range from 0x7FF000 to 0x7FFFFE.

A word write operation should be preceded by an erase of the corresponding memory location(s). The write typically requires 2 ms to complete but the write time will vary with voltage and temperature.

A program or erase operation on the data EEPROM does not stop the instruction flow. The user is responsible for waiting for the appropriate duration of time before initiating another data EEPROM write/erase operation. Attempting to read the data EEPROM while a programming or erase operation is in progress results in unspecified data.

Control bit WR initiates write operations similar to program Flash writes. This bit cannot be cleared, only set, in software. They are cleared in hardware at the completion of the write operation. The inability to clear the WR bit in software prevents the accidental or premature termination of a write operation.

The WREN bit, when set, will allow a write operation. On power-up, the WREN bit is clear. The WRERR bit is set when a write operation is interrupted by a $\overline{\text{MCLR}}$ Reset or a WDT Time-out Reset during normal operation. In these situations, following Reset, the user can check the WRERR bit and rewrite the location. The address register NVMADR remains unchanged.

**Note:** Interrupt flag bit NVMIF in the IFS0 register is set when write is complete. It must be cleared in software.

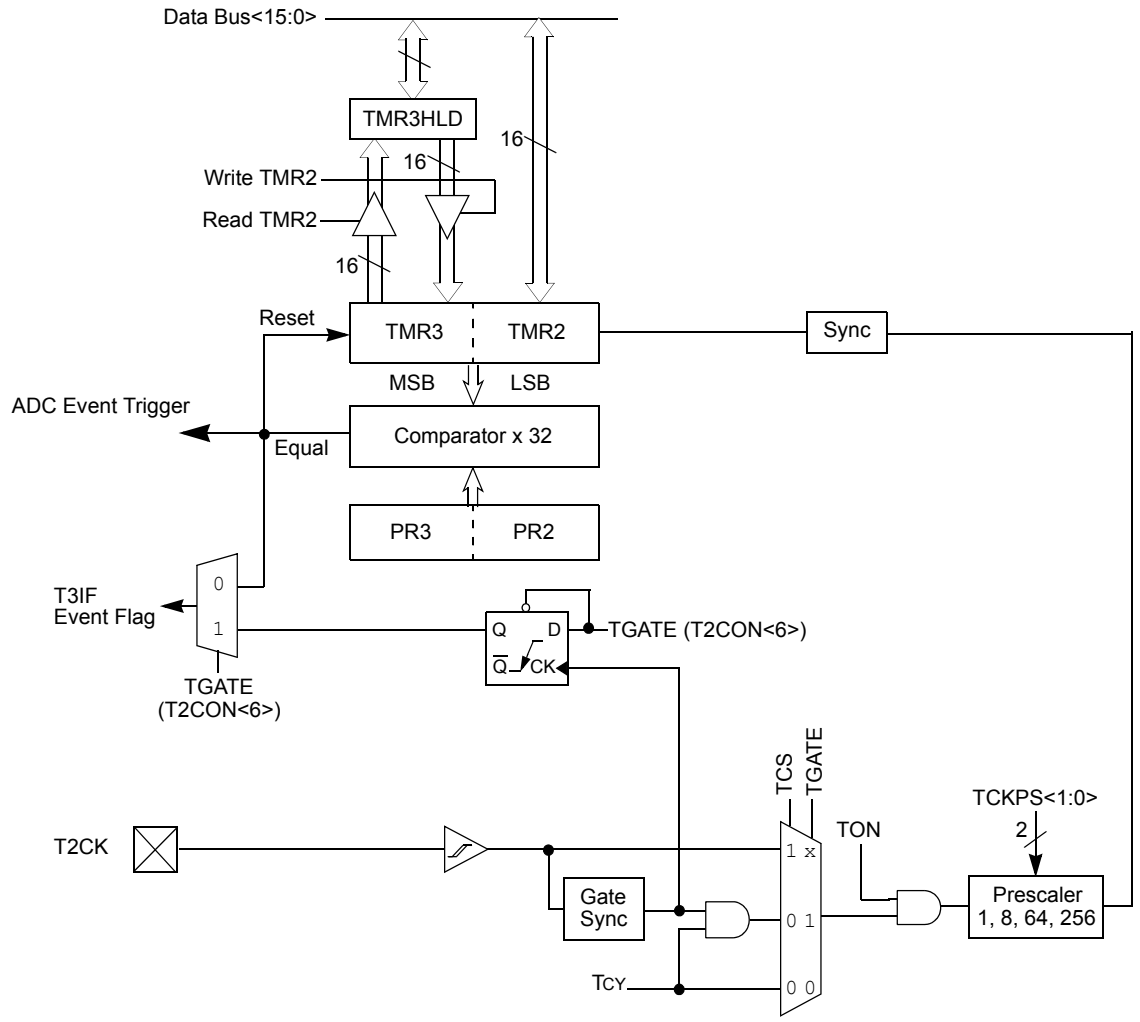
### 7.1 Reading the Data EEPROM

A `TBLRD` instruction reads a word at the current program word address. This example uses W0 as a pointer to data EEPROM. The result is placed in register W4 as shown in Example 7-1.

**EXAMPLE 7-1: DATA EEPROM READ**

```
MOV     #LOW_ADDR_WORD,W0   ; Init Pointer
MOV     #HIGH_ADDR_WORD,W1
MOV     W1,TBLPAG
TBLRDL  [ W0 ], W4          ; read data EEPROM
```

**FIGURE 10-1: 32-BIT TIMER2/3 BLOCK DIAGRAM**




**Note:** Timer configuration bit T32 (T2CON<3>) must be set to '1' for a 32-bit timer/counter operation. All control bits are respective to the T2CON register.

## 10.1 Timer Gate Operation

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

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

## 10.2 ADC Event Trigger

When a match occurs between the 32-bit timer (TMR3/TMR2) and the 32-bit combined period register (PR3/PR2), or between the 16-bit timer TMR3 and the 16-bit period register PR3, a special ADC trigger event signal is generated by Timer3.

## 10.3 Timer Prescaler

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

- A write to the TMR2/TMR3 register
- A write to the T2CON/T3CON register
- A device Reset, such as a POR and BOR

However, if the timer is disabled (TON = 0), then the Timer 2 prescaler cannot be reset since the prescaler clock is halted.

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

## 10.4 Timer Operation During Sleep Mode

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

## 10.5 Timer Interrupt

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

Enabling an interrupt is accomplished via the respective timer interrupt enable bit, T3IE (IEC0<7>).

#### 12.1.2 CAPTURE BUFFER OPERATION

Each capture channel has an associated FIFO buffer which is four 16-bit words deep. There are two status flags which provide status on the FIFO buffer:

- ICBFNE – Input Capture Buffer Not Empty
- ICOV – Input Capture Overflow

The ICBFNE will be set on the first input capture event and remain set until all capture events have been read from the FIFO. As each word is read from the FIFO, the remaining words are advanced by one position within the buffer.

In the event that the FIFO is full with four capture events and a fifth capture event occurs prior to a read of the FIFO, an overflow condition will occur and the ICOV bit will be set to a logic '1'. The fifth capture event is lost and is not stored in the FIFO. No additional events will be captured until all four events have been read from the buffer.

If a FIFO read is performed after the last read and no new capture event has been received, the read will yield indeterminate results.

#### 12.1.3 TIMER2 AND TIMER3 SELECTION MODE

The input capture module consists of up to 8 input capture channels. Each channel can select between one of two timers for the time base, Timer2 or Timer3.

Selection of the timer resource is accomplished through SFR bit, ICTMR (ICxCON<7>). Timer3 is the default timer resource available for the input capture module.

#### 12.1.4 HALL SENSOR MODE

When the input capture module is set for capture on every edge, rising and falling, ICM<2:0> = 001, the following operations are performed by the input capture logic:

- The input capture interrupt flag is set on every edge, rising and falling.
- The interrupt on Capture mode setting bits, ICI<1:0>, is ignored since every capture generates an interrupt.
- A capture overflow condition is not generated in this mode.

### 12.2 Input Capture Operation During Sleep and Idle Modes

An input capture event will generate a device wake-up or interrupt, if enabled, if the device is in CPU Idle or Sleep mode.

Independent of the timer being enabled, the input capture module will wake-up from the CPU Sleep or Idle mode when a capture event occurs if ICM<2:0> = 111 and the interrupt enable bit is asserted. The same wake-up can generate an interrupt if the conditions for processing the interrupt have been satisfied. The wake-up feature is useful as a method of adding extra external pin interrupts.

#### 12.2.1 INPUT CAPTURE IN CPU SLEEP MODE

CPU Sleep mode allows input capture module operation with reduced functionality. In the CPU Sleep mode, the ICI<1:0> bits are not applicable and the input capture module can only function as an external interrupt source.

The capture module must be configured for interrupt only on rising edge (ICM<2:0> = 111) in order for the input capture module to be used while the device is in Sleep mode. The prescale settings of 4:1 or 16:1 are not applicable in this mode.

#### 12.2.2 INPUT CAPTURE IN CPU IDLE MODE

CPU Idle mode allows input capture module operation with full functionality. In the CPU Idle mode, the Interrupt mode selected by the ICI<1:0> bits is applicable, as well as the 4:1 and 16:1 capture prescale settings which are defined by control bits ICM<2:0>. This mode requires the selected timer to be enabled. Moreover, the ICSIDL bit must be asserted to a logic '0'.

If the input capture module is defined as ICM<2:0> = 111 in CPU Idle mode, the input capture pin will serve only as an external interrupt pin.

### 12.3 Input Capture Interrupts

The input capture channels have the ability to generate an interrupt based upon the selected number of capture events. The selection number is set by control bits ICI<1:0> (ICxCON<6:5>).

Each channel provides an interrupt flag (ICxIF) bit. The respective capture channel interrupt flag is located in the corresponding IFSx status register.

Enabling an interrupt is accomplished via the respective capture channel interrupt enable (ICxIE) bit. The capture interrupt enable bit is located in the corresponding IEC control register.

**TABLE 17-1: CAN1 REGISTER MAP(1)**

| 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 |
| C1TX1EID | 0352 | Transmit Buffer 1 Extended Identifier <17:14> | | | | — | — | — | — | Transmit Buffer 1 Extended Identifier <13:6> | | | | | | | | uuuu 0000 uuuu uuuu |
| C1TX1DLC | 0354 | Transmit Buffer 1 Extended Identifier <5:0> | | | | | | TXRTR | TXRB1 | TXRB0 | DLC<3:0> | | | | — | — | — | uuuu uuuu uuuu u000 |

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

**Note 1:** Refer to the *"dsPIC30F Family Reference Manual"* (DS70046) for descriptions of register bit fields.

**TABLE 17-2: CAN2 REGISTER MAP(1) (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 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 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 |
| C2INTF | 0456 | RX0OVR | RX1OVR | TXBO | TXEP | RXEP | TXWAR | RXWAR | EWARN | IVRIF | WAKIF | ERRIF | TX2IF | TX1IF | TX0IF | RX1IF | RX0IF | 0000 0000 0000 0000 |
| C2INTE | 0458 | — | — | — | — | — | — | — | — | IVRIE | WAKIE | ERRIE | TX2IE | TX1IE | TX0IE | RX1E | RX0IE | 0000 0000 0000 0000 |
| C2EC | 045A | Transmit Error Count Register | | | | | | | | Receive Error Count Register | | | | | | | | 0000 0000 0000 0000 |

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

**Note 1:** Refer to the *"dsPIC30F Family Reference Manual"* (DS70046) for descriptions of register bit fields.

**TABLE 20-7: SYSTEM INTEGRATION REGISTER MAP FOR dsPIC30F601XA(1)**

| 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 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RCON | 0740 | TRAPR | IOPUWR | BGST | LVDEN | LVDL<3:0> | | | | EXTR | SWR | SWDTEN | WDTO | SLEEP | IDLE | BOR | POR | Depends on type of Reset. |
| OSCCON | 0742 | — | COSC<2:0> | | | — | NOSC<2:0> | | | POST<1:0> | | LOCK | — | CF | — | LPOSCEN | OSWEN | Depends on Configuration bits. |
| OSCTUN | 0744 | — | — | — | — | — | — | — | — | — | — | — | — | TUN<3:0> | | | | 0000 0000 0000 0000 |
| PMD1 | 0770 | T5MD | T4MD | T3MD | T2MD | T1MD | — | — | DCIMD | I2CMD | U2MD | U1MD | SPI2MD | SPI1MD | C2MD | C1MD | ADCMD | 0000 0000 0000 0000 |
| PMD2 | 0772 | IC8MD | IC7MD | IC6MD | IC5MD | IC4MD | IC3MD | IC2MD | IC1MD | OC8MD | OC7MD | OC6MD | OC5MD | OC4MD | OC3MD | OC2MD | OC1MD | 0000 0000 0000 0000 |

**Legend:** — = unimplemented bit, read as '0'
**Note 1:** Refer to the *"dsPIC30F Family Reference Manual"* (DS70046) for descriptions of register bit fields.

**TABLE 20-8: DEVICE CONFIGURATION REGISTER MAP(1)**

| Name | Address | 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 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| FOSC | F80000 | FCKSM<1:0> | | — | — | — | FOS<2:0> | | | — | — | — | FPR<4:0> | | | | |
| FWDT | F80002 | FWDTEN | — | — | — | — | — | — | — | — | — | FWPSA<1:0> | | FWPSB<3:0> | | | |
| FBORPOR | F80004 | MCLREN | — | — | — | — | PWMPIN(2) | HPOL(2) | LPOL(2) | BOREN | — | BORV<1:0> | | — | — | FPWRT<1:0> | |
| FBS | F80006 | — | — | RBS<1:0> | | — | — | — | EBS | — | — | — | — | BSS<2:0> | | | BWRP |
| FSS | F80008 | — | — | RSS<1:0> | | — | — | ESS<1:0> | | — | — | — | — | SSS<2:0> | | | SWRP |
| FGS | F8000A | — | — | — | — | — | — | — | — | — | — | — | — | — | GSS<1:0> | | GWRP |
| FICD | F8000C | BKBUG | COE | — | — | — | — | — | — | — | — | — | — | — | — | ICS<1:0> | |

**Legend:** — = unimplemented bit, read as '0'
**Note 1:** Refer to the *"dsPIC30F Family Reference Manual"* (DS70046) for descriptions of register bit fields.
**2:** These bits are reserved (read as '1' and must be programmed as '1').

**TABLE 21-2: INSTRUCTION SET OVERVIEW (CONTINUED)**

| Base Instr # | Assembly Mnemonic | Assembly Syntax | Description | # of Words | # of Cycles | Status Flags Affected |
|---|---|---|---|---|---|---|
| 9 | BTG | BTG f,#bit4 | Bit Toggle f | 1 | 1 | None |
| | | BTG Ws,#bit4 | Bit Toggle Ws | 1 | 1 | None |
| 10 | BTSC | BTSC f,#bit4 | Bit Test f, Skip if Clear | 1 | 1 (2 or 3) | None |
| | | BTSC Ws,#bit4 | Bit Test Ws, Skip if Clear | 1 | 1 (2 or 3) | None |
| 11 | BTSS | BTSS f,#bit4 | Bit Test f, Skip if Set | 1 | 1 (2 or 3) | None |
| | | BTSS Ws,#bit4 | Bit Test Ws, Skip if Set | 1 | 1 (2 or 3) | None |
| 12 | BTST | BTST f,#bit4 | Bit Test f | 1 | 1 | Z |
| | | BTST.C Ws,#bit4 | Bit Test Ws to C | 1 | 1 | C |
| | | BTST.Z Ws,#bit4 | Bit Test Ws to Z | 1 | 1 | Z |
| | | BTST.C Ws,Wb | Bit Test Ws<Wb> to C | 1 | 1 | C |
| | | BTST.Z Ws,Wb | Bit Test Ws<Wb> to Z | 1 | 1 | Z |
| 13 | BTSTS | BTSTS f,#bit4 | Bit Test then Set f | 1 | 1 | Z |
| | | BTSTS.C Ws,#bit4 | Bit Test Ws to C, then Set | 1 | 1 | C |
| | | BTSTS.Z Ws,#bit4 | Bit Test Ws to Z, then Set | 1 | 1 | Z |
| 14 | CALL | CALL lit23 | Call subroutine | 2 | 2 | None |
| | | CALL Wn | Call indirect subroutine | 1 | 2 | None |
| 15 | CLR | CLR f | f = 0x0000 | 1 | 1 | None |
| | | CLR WREG | WREG = 0x0000 | 1 | 1 | None |
| | | CLR Ws | Ws = 0x0000 | 1 | 1 | None |
| | | CLR Acc,Wx,Wxd,Wy,Wyd,AWB | Clear Accumulator | 1 | 1 | OA,OB,SA,SB |
| 16 | CLRWDT | CLRWDT | Clear Watchdog Timer | 1 | 1 | WDTO,Sleep |
| 17 | COM | COM f | $f = \overline{f}$ | 1 | 1 | N,Z |
| | | COM f,WREG | $WREG = \overline{f}$ | 1 | 1 | N,Z |
| | | COM Ws,Wd | $Wd = \overline{Ws}$ | 1 | 1 | N,Z |
| 18 | CP | CP f | Compare f with WREG | 1 | 1 | C,DC,N,OV,Z |
| | | CP Wb,#lit5 | Compare Wb with lit5 | 1 | 1 | C,DC,N,OV,Z |
| | | CP Wb,Ws | Compare Wb with Ws (Wb - Ws) | 1 | 1 | C,DC,N,OV,Z |
| 19 | CP0 | CP0 f | Compare f with 0x0000 | 1 | 1 | C,DC,N,OV,Z |
| | | CP0 Ws | Compare Ws with 0x0000 | 1 | 1 | C,DC,N,OV,Z |
| 20 | CPB | CPB f | Compare f with WREG, with Borrow | 1 | 1 | C,DC,N,OV,Z |
| | | CPB Wb,#lit5 | Compare Wb with lit5, with Borrow | 1 | 1 | C,DC,N,OV,Z |
| | | CPB Wb,Ws | Compare Wb with Ws, with Borrow (Wb - Ws - $\overline{C}$) | 1 | 1 | C,DC,N,OV,Z |
| 21 | CPSEQ | CPSEQ Wb, Wn | Compare Wb with Wn, skip if = | 1 | 1 (2 or 3) | None |
| 22 | CPSGT | CPSGT Wb, Wn | Compare Wb with Wn, skip if > | 1 | 1 (2 or 3) | None |
| 23 | CPSLT | CPSLT Wb, Wn | Compare Wb with Wn, skip if < | 1 | 1 (2 or 3) | None |
| 24 | CPSNE | CPSNE Wb, Wn | Compare Wb with Wn, skip if ≠ | 1 | 1 (2 or 3) | None |
| 25 | DAW | DAW Wn | Wn = decimal adjust Wn | 1 | 1 | C |
| 26 | DEC | DEC f | f = f -1 | 1 | 1 | C,DC,N,OV,Z |
| | | DEC f,WREG | WREG = f -1 | 1 | 1 | C,DC,N,OV,Z |
| | | DEC Ws,Wd | Wd = Ws - 1 | 1 | 1 | C,DC,N,OV,Z |
| 27 | DEC2 | DEC2 f | f = f -2 | 1 | 1 | C,DC,N,OV,Z |
| | | DEC2 f,WREG | WREG = f -2 | 1 | 1 | C,DC,N,OV,Z |
| | | DEC2 Ws,Wd | Wd = Ws - 2 | 1 | 1 | C,DC,N,OV,Z |
| 28 | DISI | DISI #lit14 | Disable Interrupts for k instruction cycles | 1 | 1 | None |

## 22.7 MPLAB SIM Software Simulator

The MPLAB SIM Software Simulator allows code development in a PC-hosted environment by simulating the PIC MCUs and dsPIC® DSCs on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a comprehensive stimulus controller. Registers can be logged to files for further run-time analysis. The trace buffer and logic analyzer display extend the power of the simulator to record and track program execution, actions on I/O, most peripherals and internal registers.

The MPLAB SIM Software Simulator fully supports symbolic debugging using the MPLAB C Compilers, and the MPASM and MPLAB Assemblers. The software simulator offers the flexibility to develop and debug code outside of the hardware laboratory environment, making it an excellent, economical software development tool.

## 22.8 MPLAB REAL ICE In-Circuit Emulator System

MPLAB REAL ICE In-Circuit Emulator System is Microchip's next generation high-speed emulator for Microchip Flash DSC and MCU devices. It debugs and programs PIC® Flash MCUs and dsPIC® Flash DSCs with the easy-to-use, powerful graphical user interface of the MPLAB Integrated Development Environment (IDE), included with each kit.

The emulator is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with either a connector compatible with in-circuit debugger systems (RJ11) or with the new high-speed, noise tolerant, Low-Voltage Differential Signal (LVDS) interconnection (CAT5).

The emulator is field upgradable through future firmware downloads in MPLAB IDE. In upcoming releases of MPLAB IDE, new devices will be supported, and new features will be added. MPLAB REAL ICE offers significant advantages over competitive emulators including low-cost, full-speed emulation, run-time variable watches, trace analysis, complex breakpoints, a ruggedized probe interface and long (up to three meters) interconnection cables.

## 22.9 MPLAB ICD 3 In-Circuit Debugger System

MPLAB ICD 3 In-Circuit Debugger System is Microchip's most cost effective high-speed hardware debugger/programmer for Microchip Flash Digital Signal Controller (DSC) and microcontroller (MCU) devices. It debugs and programs PIC® Flash microcontrollers and dsPIC® DSCs with the powerful, yet easy-to-use graphical user interface of MPLAB Integrated Development Environment (IDE).

The MPLAB ICD 3 In-Circuit Debugger probe is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with a connector compatible with the MPLAB ICD 2 or MPLAB REAL ICE systems (RJ-11). MPLAB ICD 3 supports all MPLAB ICD 2 headers.

## 22.10 PICkit 3 In-Circuit Debugger/ Programmer and PICkit 3 Debug Express

The MPLAB PICkit 3 allows debugging and programming of PIC® and dsPIC® Flash microcontrollers at a most affordable price point using the powerful graphical user interface of the MPLAB Integrated Development Environment (IDE). The MPLAB PICkit 3 is connected to the design engineer's PC using a full speed USB interface and can be connected to the target via an Microchip debug (RJ-11) connector (compatible with MPLAB ICD 3 and MPLAB REAL ICE). The connector uses two device I/O pins and the reset line to implement in-circuit debugging and In-Circuit Serial Programming™.

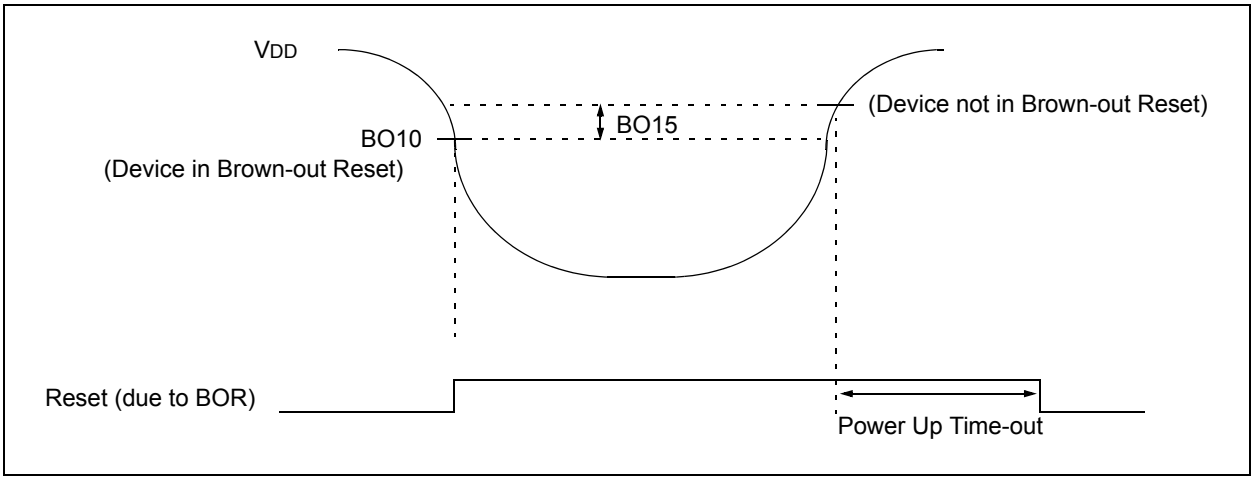
The PICkit 3 Debug Express include the PICkit 3, demo board and microcontroller, hookup cables and CDROM with user's guide, lessons, tutorial, compiler and MPLAB IDE software.

**TABLE 23-10: ELECTRICAL CHARACTERISTICS: LVDL**

| DC CHARACTERISTICS | | | Standard Operating Conditions: 2.5V to 5.5V (unless otherwise stated)<br>Operating temperature -40°C ≤TA ≤+85°C for Industrial<br>-40°C ≤TA ≤+125°C for Extended | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Param No.** | **Symbol** | **Characteristic(1)** | | **Min** | **Typ** | **Max** | **Units** | **Conditions** |
| LV10 | VPLVD | LVDL Voltage on VDD transition high to low | LVDL = 0000(2) | — | — | — | V | |
| | | | LVDL = 0001(2) | — | — | — | V | |
| | | | LVDL = 0010(2) | — | — | — | V | |
| | | | LVDL = 0011(2) | — | — | — | V | |
| | | | LVDL = 0100 | 2.50 | — | 2.65 | V | |
| | | | LVDL = 0101 | 2.70 | — | 2.86 | V | |
| | | | LVDL = 0110 | 2.80 | — | 2.97 | V | |
| | | | LVDL = 0111 | 3.00 | — | 3.18 | V | |
| | | | LVDL = 1000 | 3.30 | — | 3.50 | V | |
| | | | LVDL = 1001 | 3.50 | — | 3.71 | V | |
| | | | LVDL = 1010 | 3.60 | — | 3.82 | V | |
| | | | LVDL = 1011 | 3.80 | — | 4.03 | V | |
| | | | LVDL = 1100 | 4.00 | — | 4.24 | V | |
| | | | LVDL = 1101 | 4.20 | — | 4.45 | V | |
| | | | LVDL = 1110 | 4.50 | — | 4.77 | V | |
| LV15 | VLVDIN | External LVD input pin threshold voltage | LVDL = 1111 | — | — | — | V | |

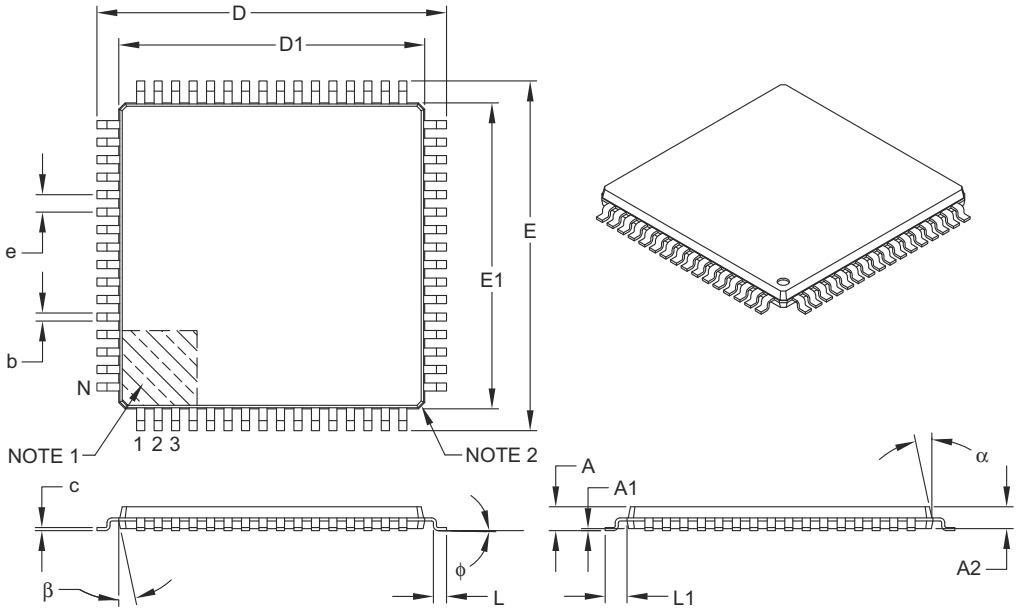
**Note 1:** These parameters are characterized but not tested in manufacturing.
**2:** These values not in usable operating range.

**FIGURE 23-2: BROWN-OUT RESET CHARACTERISTICS**



### 64-Lead Plastic Thin Quad Flatpack (PF) – 14x14x1 mm Body, 2.00 mm [TQFP]

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




| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Number of Leads | N | | 64 | |
| Lead Pitch | e | | 0.80 BSC | |
| Overall Height | A | – | – | 1.20 |
| Molded Package Thickness | A2 | 0.95 | 1.00 | 1.05 |
| Standoff | A1 | 0.05 | – | 0.15 |
| Foot Length | L | 0.45 | 0.60 | 0.75 |
| Footprint | L1 | | 1.00 REF | |
| Foot Angle | ϕ | 0° | 3.5° | 7° |
| Overall Width | E | | 16.00 BSC | |
| Overall Length | D | | 16.00 BSC | |
| Molded Package Width | E1 | | 14.00 BSC | |
| Molded Package Length | D1 | | 14.00 BSC | |
| Lead Thickness | c | 0.09 | – | 0.20 |
| Lead Width | b | 0.30 | 0.37 | 0.45 |
| Mold Draft Angle Top | α | 11° | 12° | 13° |
| Mold Draft Angle Bottom | β | 11° | 12° | 13° |

**Notes:**

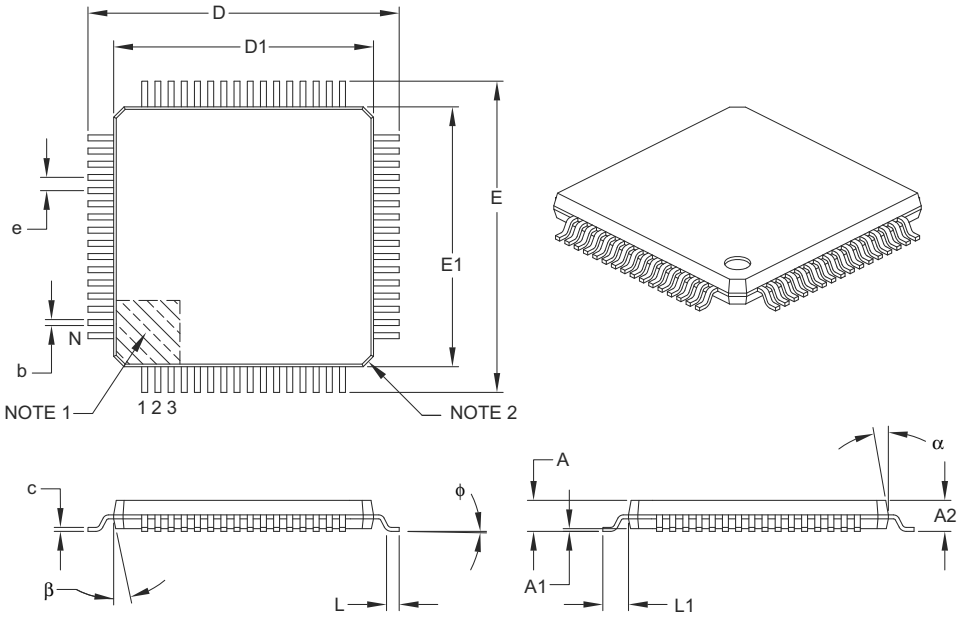
1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Chamfers at corners are optional; size may vary.
3. Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.25 mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-066B

### 64-Lead Plastic Thin Quad Flatpack (PT) – 10x10x1 mm Body, 2.00 mm [TQFP]

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




| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Number of Leads | N | | 64 | |
| Lead Pitch | e | | 0.50 BSC | |
| Overall Height | A | – | – | 1.20 |
| Molded Package Thickness | A2 | 0.95 | 1.00 | 1.05 |
| Standoff | A1 | 0.05 | – | 0.15 |
| Foot Length | L | 0.45 | 0.60 | 0.75 |
| Footprint | L1 | | 1.00 REF | |
| Foot Angle | φ | 0° | 3.5° | 7° |
| Overall Width | E | | 12.00 BSC | |
| Overall Length | D | | 12.00 BSC | |
| Molded Package Width | E1 | | 10.00 BSC | |
| Molded Package Length | D1 | | 10.00 BSC | |
| Lead Thickness | c | 0.09 | – | 0.20 |
| Lead Width | b | 0.17 | 0.22 | 0.27 |
| Mold Draft Angle Top | α | 11° | 12° | 13° |
| Mold Draft Angle Bottom | β | 11° | 12° | 13° |

**Notes:**

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Chamfers at corners are optional; size may vary.
3. Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.25 mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M.

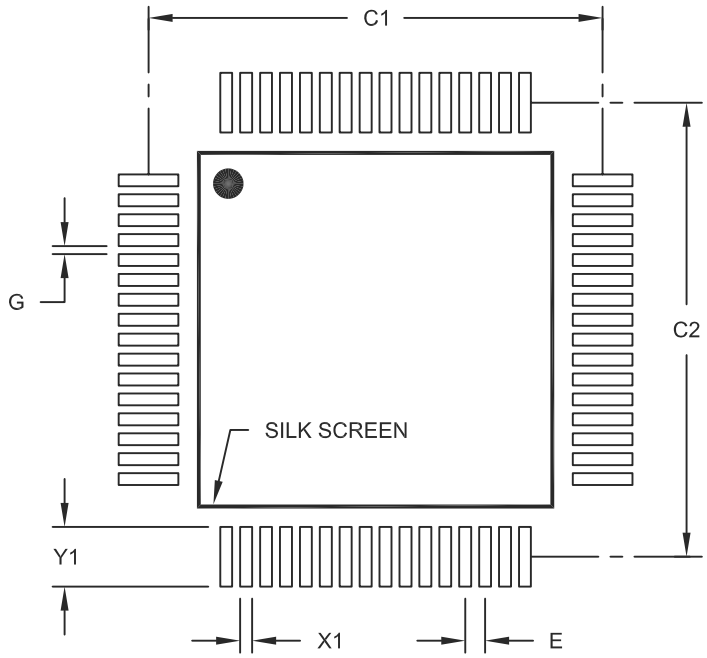
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-085B

### 64-Lead Plastic Thin Quad Flatpack (PT) – 10x10x1 mm Body, 2.00 mm [TQFP]

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




RECOMMENDED LAND PATTERN

| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Contact Pitch | E | | 0.50 BSC | |
| Contact Pad Spacing | C1 | | 11.40 | |
| Contact Pad Spacing | C2 | | 11.40 | |
| Contact Pad Width (X64) | X1 | | | 0.30 |
| Contact Pad Length (X64) | Y1 | | | 1.50 |
| Distance Between Pads | G | 0.20 | | |

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2085A

## THE MICROCHIP WEB SITE

Microchip provides online support via our WWW site at www.microchip.com. This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

## CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at www.microchip.com. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

## CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support
- Development Systems Information Line

Customers should contact their distributor, representative or field application engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

**Technical support is available through the web site at: http://support.microchip.com**