

Welcome to E-XFL.COM

#### What is "Embedded - Microcontrollers"?

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

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

#### Details

E·XF

Product Status	Active
Core Processor	dsPIC
Core Size	16-Bit
Speed	20 MIPS
Connectivity	I <sup>2</sup> C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	12
Program Memory Size	12KB (4K x 24)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	2.5V ~ 5.5V
Data Converters	A/D 8x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	18-SOIC (0.295", 7.50mm Width)
Supplier Device Package	18-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/dspic30f2011-20i-so

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

# **Table of Contents**

1.0	Device Overview	11
2.0	CPU Architecture Overview	19
3.0	Memory Organization	29
4.0	Address Generator Units	43
5.0	Flash Program Memory	49
6.0	Data EEPROM Memory	55
7.0	I/O Ports	59
8.0	Interrupts	65
9.0	Timer1 Module	73
10.0	Timer2/3 Module	77
11.0	Input Capture Module	83
12.0	Output Compare Module	87
13.0	SPI™ Module	93
14.0	I2C™ Module	97
15.0	Universal Asynchronous Receiver Transmitter (UART) Module	. 105
16.0	12-bit Analog-to-Digital Converter (ADC) Module	. 113
17.0	System Integration	. 123
18.0	Instruction Set Summary	. 137
19.0	Development Support	. 145
20.0	Electrical Characteristics	. 149
21.0	Packaging Information	. 187
Index	۲	. 201
The I	Aicrochip Web Site	. 207
Custo	omer Change Notification Service	. 207
Custo	omer Support	. 207
Read	er Response	. 208
Prod	uct Identification System	. 209

# 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.

# FIGURE 1-3: dsPIC30F3012 BLOCK DIAGRAM





# 2.3 Divide Support

The dsPIC DSC devices feature a 16/16-bit signed fractional divide operation, as well as 32/16-bit and 16/16-bit signed and unsigned integer divide operations, in the form of single instruction iterative divides. The following instructions and data sizes are supported:

- 1. DIVF 16/16 signed fractional divide
- 2. DIV.sd 32/16 signed divide
- 3. DIV.ud 32/16 unsigned divide
- 4. DIV.s 16/16 signed divide
- 5. DIV.u 16/16 unsigned divide

The 16/16 divides are similar to the 32/16 (same number of iterations), but the dividend is either zero-extended or sign-extended during the first iteration.

The divide instructions must be executed within a REPEAT loop. Any other form of execution (e.g., a series of discrete divide instructions) will not function correctly because the instruction flow depends on RCOUNT. The divide instruction does not automatically set up the RCOUNT value and it must, therefore, be explicitly and correctly specified in the REPEAT instruction, as shown in Table 2-1 (REPEAT executes the target instruction {operand value+1} times). The REPEAT loop count must be setup for 18 iterations of the DIV/DIVF instruction. Thus, a complete divide operation requires 19 cycles.

**Note:** The divide flow is interruptible; however, the user needs to save the context as appropriate.

#### TABLE 2-1: DIVIDE INSTRUCTIONS

Instruction	Function
DIVF	Signed fractional divide: Wm/Wn $\rightarrow$ W0; Rem $\rightarrow$ W1
DIV.sd	Signed divide: (Wm+1:Wm)/Wn $\rightarrow$ W0; Rem $\rightarrow$ W1
DIV.s	Signed divide: Wm/Wn $\rightarrow$ W0; Rem $\rightarrow$ W1
DIV.ud	Unsigned divide: (Wm+1:Wm)/Wn $\rightarrow$ W0; Rem $\rightarrow$ W1
DIV.u	Unsigned divide: Wm/Wn $\rightarrow$ W0; Rem $\rightarrow$ W1



# 3.0 MEMORY ORGANIZATION

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).

# 3.1 Program Address Space

The program address space is 4M instruction words. The program space memory maps for the dsPIC30F2011/2012/3012/3013 devices is shown in Figure 3-1.

Program memory is addressable by a 24-bit value from either the 23-bit PC, table instruction Effective Address (EA), or data space EA, when program space is mapped into data space as defined by Table 3-1. Note that the program space address is incremented by two between successive program words in order to provide compatibility with data space addressing.

User program space access is restricted to the lower 4M instruction word address range (0x000000 to 0x7FFFFE) for all accesses other than TBLRD/TBLWT, which uses TBLPAG<7> to determine user or configuration space access. In Table 3-1, Program Space Address Construction, bit 23 allows access to the Device ID, the User ID and the Configuration bits. Otherwise, bit 23 is always clear.

#### 3.1.1 DATA ACCESS FROM PROGRAM MEMORY USING TABLE INSTRUCTIONS

This architecture fetches 24-bit wide program memory. Consequently, instructions are always aligned. However, as the architecture is modified Harvard, data can also be present in program space.

There are two methods by which program space can be accessed: via special table instructions, or through the remapping of a 16K word program space page into the upper half of data space (see Section 3.1.2 "Data Access from Program Memory Using Program Space Visibility"). The TBLRDL and TBLWTL instructions offer a direct method of reading or writing the lsw of any address within program space, without going through data space. The TBLRDH and TBLWTH instructions are the only method whereby the upper 8 bits of a program space word can be accessed as data.

The PC is incremented by two for each successive 24-bit program word. This allows program memory addresses to directly map to data space addresses. Program memory can thus be regarded as two 16-bit word wide address spaces, residing side by side, each with the same address range. TBLRDL and TBLWTL access the space which contains the lsw, and TBLRDH and TBLWTH access the space which contains the MSB.

Figure 3-2 shows how the EA is created for table operations and data space accesses (PSV = 1). Here, P<23:0> refers to a program space word, whereas D<15:0> refers to a data space word.

A set of table instructions are provided to move byte or word-sized data to and from program space. See Figure 3-4 and Figure 3-5.

 TBLRDL: Table Read Low Word: Read the LS Word of the program address; P<15:0> maps to D<15:0>. Byte: Read one of the LSB of the program

address; P < 7:0 > maps to the destination byte when byte select = 0;

P<15:8> maps to the destination byte when byte select = 1.

- TBLWTL: Table Write Low (refer to Section 5.0 "Flash Program Memory" for details on Flash Programming)
- TBLRDH: Table Read High Word: Read the MS Word of the program address; P<23:16> maps to D<7:0>; D<15:8> will always be = 0.

*Byte:* Read one of the MSB of the program address;

P<23:16> maps to the destination byte when byte select = 0;

The destination byte will always be = 0 when byte select = 1.

 TBLWTH: Table Write High (refer to Section 5.0 "Flash Program Memory" for details on Flash Programming)

# FIGURE 3-3: PROGRAM DATA TABLE ACCESS (Isw)



# 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 3-7 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 is 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) returns 0x0000.

#### 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 read the complete word that 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++] results in a value of Ws + 1 for byte operations and Ws + 2 for word operations.

All word accesses must be aligned to an even address. Misaligned word data fetches are not supported, so care should 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 is generated. If the error occurred on a read, the instruction underway is completed, whereas if it occurred on a write, the instruction is executed, but the write does not occur. In either case, a trap is then executed, allowing the system and/or user to examine the machine state prior to execution of the address fault.

#### FIGURE 3-9: DATA ALIGNMENT

	15 <b>MSB</b>	8 7	LSB	0	
0001	Byte 1		Byte 0		0000
0003	Byte 3		Byte 2		0002
0005	Byte 5		Byte 4		0004

# TABLE 3-3: CORE REGISTER MAP (CONTINUED)

	•••••••••••••••••••••••••••••••••••••••	••••			(		/											
SFR Name	Address (Home)	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset State
CORCON	0044	—	—	—	US	EDT	DL2	DL1	DL0	SATA	SATB	SATDW	ACCSAT	IPL3	PSV	RND	IF	0000 0000 0010 0000
MODCON	0046	XMODEN	YMODEN	—	—		BWM	l<3:0>			YW	M<3:0>			XWM<	:3:0>		0000 0000 0000 0000
XMODSRT	0048							X	S<15:1>								0	uuuu uuuu uuuu uuu0
XMODEND	004A							XI	E<15:1>								1	uuuu uuuu uuul
YMODSRT	004C							YS	S<15:1>								0	uuuu uuuu uuuu uuu0
YMODEND	004E		YE<15:1> 1 u										uuuu uuuu uuul					
XBREV	0050	BREN	BREN XB<14:0> u										uuuu uuuu uuuu					
DISICNT	0052	_	_							DISICN	Г<13:0>							0000 0000 0000 0000

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

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

# 11.0 INPUT CAPTURE MODULE

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

This section describes the input capture module and associated operational modes. The features provided by this module are useful in applications requiring frequency (period) and pulse measurement.

Figure 11-1 depicts a block diagram of the input capture module. Input capture is useful for such modes as:

- Frequency/Period/Pulse Measurements
- · Additional Sources of External Interrupts

Important operational features of the input capture module are:

- Simple Capture Event mode
- Timer2 and Timer3 mode selection
- · Interrupt on input capture event

These operating modes are determined by setting the appropriate bits in the IC1CON and IC2CON registers. The dsPIC30F2011/2012/3012/3013 devices have two capture channels.

# 11.1 Simple Capture Event Mode

The simple capture events in the dsPIC30F product family are:

- · Capture every falling edge
- Capture every rising edge
- Capture every 4th rising edge
- · Capture every 16th rising edge
- · Capture every rising and falling edge

These simple Input Capture modes are configured by setting the appropriate bits, ICM<2:0> (ICxCON<2:0>).

#### 11.1.1 CAPTURE PRESCALER

There are four input capture prescaler settings specified by bits ICM<2:0> (ICxCON<2:0>). Whenever the capture channel is turned off, the prescaler counter is cleared. In addition, any Reset clears the prescaler counter.

## FIGURE 11-1: INPUT CAPTURE MODE BLOCK DIAGRAM<sup>(1)</sup>



#### TABLE 13-1: SPI1 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
SPI1STAT	0220	SPIEN	—	SPISIDL	-	_	—	_	_	—	SPIROV	—	—	—	—	SPITBF	SPIRBF	0000 0000 0000 0000
SPI1CON	0222	_	FRMEN	SPIFSD	-	DISSDO	MODE16	SMP	CKE	SSEN	CKP	MSTEN	SPRE2	SPRE1	SPRE0	PPRE1	PPRE0	0000 0000 0000 0000
SPI1BUF	0224							Tra	ansmit an	d Receive	Buffer							0000 0000 0000 0000

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

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

NOTES:

# FIGURE 15-2: UART RECEIVER BLOCK DIAGRAM



### 15.5.2 FRAMING ERROR (FERR)

The FERR bit (UxSTA<2>) is set if a '0' is detected instead of a Stop bit. If two Stop bits are selected, both Stop bits must be '1', otherwise FERR will be set. The read-only FERR bit is buffered along with the received data. It is cleared on any Reset.

#### 15.5.3 PARITY ERROR (PERR)

The PERR bit (UxSTA<3>) is set if the parity of the received word is incorrect. This error bit is applicable only if a Parity mode (odd or even) is selected. The read-only PERR bit is buffered along with the received data bytes. It is cleared on any Reset.

#### 15.5.4 IDLE STATUS

When the receiver is active (i.e., between the initial detection of the Start bit and the completion of the Stop bit), the RIDLE bit (UxSTA<4>) is '0'. Between the completion of the Stop bit and detection of the next Start bit, the RIDLE bit is '1', indicating that the UART is Idle.

#### 15.5.5 RECEIVE BREAK

The receiver will count and expect a certain number of bit times based on the values programmed in the PDSEL (UxMODE<2:1>) and STSEL (UxMODE<0>) bits.

If the break is longer than 13 bit times, the reception is considered complete after the number of bit times specified by PDSEL and STSEL. The URXDA bit is set, FERR is set, zeros are loaded into the receive FIFO, interrupts are generated if appropriate and the RIDLE bit is set.

When the module receives a long break signal and the receiver has detected the Start bit, the data bits and the invalid Stop bit (which sets the FERR), the receiver must wait for a valid Stop bit before looking for the next Start bit. It cannot assume that the break condition on the line is the next Start bit.

Break is regarded as a character containing all '0's with the FERR bit set. The Break character is loaded into the buffer. No further reception can occur until a Stop bit is received. Note that RIDLE goes high when the Stop bit has not yet been received.

# 15.6 Address Detect Mode

Setting the ADDEN bit (UxSTA<5>) enables this special mode in which a 9th bit (URX8) value of '1' identifies the received word as an address, rather than data. This mode is only applicable for 9-bit data communication. The URXISEL control bit does not have any impact on interrupt generation in this mode since an interrupt (if enabled) will be generated every time the received word has the 9th bit set.

## 15.7 Loopback Mode

Setting the LPBACK bit enables this special mode in which the UxTX pin is internally connected to the UxRX pin. When configured for the Loopback mode, the UxRX pin is disconnected from the internal UART receive logic. However, the UxTX pin still functions as in a normal operation.

To select this mode:

- a) Configure UART for desired mode of operation.
- b) Set LPBACK = 1 to enable Loopback mode.
- c) Enable transmission as defined in Section 15.3 "Transmitting Data".

## 15.8 Baud Rate Generator

The UART has a 16-bit Baud Rate Generator to allow maximum flexibility in baud rate generation. The Baud Rate Generator register (UxBRG) is readable and writable. The baud rate is computed as follows:

- BRG = 16-bit value held in UxBRG register (0 through 65535)
- FCY = Instruction Clock Rate (1/Tcy)

The baud rate is given by Equation 15-1.

#### EQUATION 15-1: BAUD RATE

Baud Rate = FCY / (16\*(BRG+1))

Therefore, the maximum baud rate possible is:

FCY /16 (if BRG = 0),

and the minimum baud rate possible is:

Fcy / (16\* 65536).

With a full 16-bit Baud Rate Generator at 30 MIPS operation, the minimum baud rate achievable is 28.5 bps.

# 15.9 Auto-Baud Support

To allow the system to determine baud rates of received characters, the input can be optionally linked to a selected capture input (IC1 for UART1 and IC2 for UART2). To enable this mode, you must program the input capture module to detect the falling and rising edges of the Start bit.

# 17.2.7 FAIL-SAFE CLOCK MONITOR

The Fail-Safe Clock Monitor (FSCM) allows the device to continue to operate even in the event of an oscillator failure. The FSCM function is enabled by appropriately programming the FCKSM Configuration bits (clock switch and monitor selection bits) in the FOSC Device Configuration register. If the FSCM function is enabled, the LPRC internal oscillator will run at all times (except during Sleep mode) and will not be subject to control by the SWDTEN bit.

In the event of an oscillator failure, the FSCM will generate a clock failure trap event and will switch the system clock over to the FRC oscillator. The user will then have the option to either attempt to restart the oscillator or execute a controlled shutdown. The user may decide to treat the trap as a warm Reset by simply loading the Reset address into the oscillator fail trap vector. In this event, the CF (Clock Fail) bit (OSCCON<3>) is also set whenever a clock failure is recognized.

In the event of a clock failure, the WDT is unaffected and continues to run on the LPRC clock.

If the oscillator has a very slow start-up time coming out of POR, BOR or Sleep, it is possible that the PWRT timer will expire before the oscillator has started. In such cases, the FSCM will be activated and the FSCM will initiate a clock failure trap, and the COSC<2:0> bits are loaded with FRC oscillator selection. This will effectively shut-off the original oscillator that was trying to start.

The user may detect this situation and restart the oscillator in the clock fail trap ISR.

Upon a clock failure detection, the FSCM module will initiate a clock switch to the FRC oscillator as follows:

- 1. The COSC bits (OSCCON<14:12>) are loaded with the FRC oscillator selection value.
- 2. CF bit is set (OSCCON<3>).
- 3. OSWEN control bit (OSCCON<0>) is cleared.

For the purpose of clock switching, the clock sources are sectioned into four groups:

- Primary (with or without PLL)
- Secondary
- Internal FRC
- Internal LPRC

The user can switch between these functional groups but cannot switch between options within a group. If the primary group is selected, then the choice within the group is always determined by the FPR<4:0> Configuration bits. The OSCCON register holds the Control and Status bits related to clock switching.

- COSC<2:0>: Read-only bits always reflect the current oscillator group in effect.
- NOSC<2:0>: Control bits which are written to indicate the new oscillator group of choice.
  - On POR and BOR, COSC<2:0> and NOSC<2:0> are both loaded with the Configuration bit values FOS<2:0>.
- LOCK: The LOCK bit indicates a PLL lock.
- CF: Read-only bit indicating if a clock fail detect has occurred.
- OSWEN: Control bit changes from a '0' to a '1' when a clock transition sequence is initiated. Clearing the OSWEN control bit will abort a clock transition in progress (used for hang-up situations).

If Configuration bits FCKSM<1:0> = 1x, then the clock switching and Fail-Safe Clock monitoring functions are disabled. This is the default Configuration bit setting.

If clock switching is disabled, then the FOS<2:0> and FPR<4:0> bits directly control the oscillator selection and the COSC<2:0> bits do not control the clock selection. However, these bits will reflect the clock source selection.

**Note:** The application should not attempt to switch to a clock of frequency lower than 100 kHz when the Fail-Safe Clock Monitor is enabled. If such clock switching is performed, the device may generate an oscillator fail trap and switch to the Fast RC oscillator.

### 17.2.8 PROTECTION AGAINST ACCIDENTAL WRITES TO OSCCON

A write to the OSCCON register is intentionally made difficult because it controls clock switching and clock scaling.

To write to the OSCCON low byte, the following code sequence must be executed without any other instructions in between:

Byte Write 0x46 to OSCCON low Byte Write 0x57 to OSCCON low

*Byte write is allowed for one instruction cycle.* Write the desired value or use bit manipulation instruction.

To write to the OSCCON high byte, the following instructions must be executed without any other instructions in between:

Byte Write 0x78 to OSCCON high Byte Write 0x9A to OSCCON high

*Byte write is allowed for one instruction cycle*. Write the desired value or use bit manipulation instruction.

# 17.3 Reset

The dsPIC30F2011/2012/3012/3013 devices differentiate between various kinds of Reset:

- a) Power-on Reset (POR)
- b) MCLR Reset during normal operation
- c) 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)
- 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 17-5. 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 17-2.

A MCLR noise filter is provided in the MCLR Reset path. The filter detects and ignores small pulses.

Internally generated Resets do not drive MCLR pin low.

# 17.3.1 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  $\mu$ s 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-<u>up</u>, <u>TPOR</u> + <u>TPWRT</u>. When these delays have expired, 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 SYSRST signal is shown in Figure 17-3 through Figure 17-5.



# FIGURE 17-2: RESET SYSTEM BLOCK DIAGRAM

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

#### TABLE 18-1: SYMBOLS USED IN OPCODE DESCRIPTIONS (CONTINUED)

# TABLE 18-2: INSTRUCTION SET OVERVIEW (CONTINUED)

Base Instr #	Assembly Mnemonic		Assembly Syntax	Description	# of Words	# of Cycle s	Status Flags Affected
66	RRNC	RRNC	f	f = Rotate Right (No Carry) f	1	1	N,Z
		RRNC	f,WREG	WREG = Rotate Right (No Carry) f	1	1	N,Z
		RRNC	Ws,Wd	Wd = Rotate Right (No Carry) Ws	1	1	N,Z
67	SAC	SAC	Acc,#Slit4,Wdo	Store Accumulator	1	1	None
		SAC.R	Acc,#Slit4,Wdo	Store Rounded Accumulator	1	1	None
68	SE	SE	Ws,Wnd	Wnd = sign-extended Ws	1	1	C,N,Z
69	SETM	SETM	f	f = 0xFFFF	1	1	None
		SETM	WREG	WREG = 0xFFFF	1	1	None
		SETM	Ws	Ws = 0xFFFF	1	1	None
70	SFTAC	SFTAC	Acc,Wn	Arithmetic Shift Accumulator by (Wn)	1	1	OA,OB,OAB, SA,SB,SAB
		SFTAC	Acc,#Slit6	Arithmetic Shift Accumulator by Slit6	1	1	OA,OB,OAB, SA,SB,SAB
71	SL	SL	f	f = Left Shift f	1	1	C,N,OV,Z
		SL	f,WREG	WREG = Left Shift f	1	1	C,N,OV,Z
		SL	Ws,Wd	Wd = Left Shift Ws	1	1	C,N,OV,Z
		SL	Wb,Wns,Wnd	Wnd = Left Shift Wb by Wns	1	1	N,Z
		SL	Wb,#lit5,Wnd	Wnd = Left Shift Wb by lit5	1	1	N,Z
72	SUB	SUB	Acc	Subtract Accumulators	1	1	OA,OB,OAB, SA,SB,SAB
		SUB	f	f = f - WREG	1	1	C,DC,N,OV,Z
		SUB	f,WREG	WREG = f - WREG	1	1	C,DC,N,OV,Z
		SUB	#lit10,Wn	Wn = Wn - lit10	1	1	C,DC,N,OV,Z
		SUB	Wb,Ws,Wd	Wd = Wb - Ws	1	1	C,DC,N,OV,Z
		SUB	Wb,#lit5,Wd	Wd = Wb - lit5	1	1	C,DC,N,OV,Z
73	SUBB	SUBB	f	$f = f - WREG - (\overline{C})$	1	1	C,DC,N,OV,Z
		SUBB	f,WREG	WREG = f - WREG - $(\overline{C})$	1	1	C,DC,N,OV,Z
		SUBB	#lit10,Wn	$Wn = Wn - lit10 - (\overline{C})$	1	1	C,DC,N,OV,Z
		SUBB	Wb,Ws,Wd	$Wd = Wb - Ws - (\overline{C})$	1	1	C,DC,N,OV,Z
		SUBB	Wb,#lit5,Wd	$Wd = Wb - lit5 - (\overline{C})$	1	1	C,DC,N,OV,Z
74	SUBR	SUBR	f	f = WREG - f	1	1	C,DC,N,OV,Z
		SUBR	f,WREG	WREG = WREG - f	1	1	C,DC,N,OV,Z
		SUBR	Wb,Ws,Wd	Wd = Ws - Wb	1	1	C,DC,N,OV,Z
		SUBR	Wb,#lit5,Wd	Wd = lit5 - Wb	1	1	C,DC,N,OV,Z
75	SUBBR	SUBBR	f	$f = WREG - f - (\overline{C})$	1	1	C,DC,N,OV,Z
		SUBBR	f,WREG	WREG = WREG -f - $(\overline{C})$	1	1	C,DC,N,OV,Z
		SUBBR	Wb,Ws,Wd	$Wd = Ws - Wb - (\overline{C})$	1	1	C,DC,N,OV,Z
		SUBBR	Wb,#lit5,Wd	$Wd = lit5 - Wb - (\overline{C})$	1	1	C,DC,N,OV,Z
76	SWAP	SWAP.b	Wn	Wn = nibble swap Wn	1	1	None
		SWAP	Wn	Wn = byte swap Wn	1	1	None
77	TBLRDH	TBLRDH	Ws,Wd	Read Prog<23:16> to Wd<7:0>	1	2	None
78	TBLRDL	TBLRDL	Ws,Wd	Read Prog<15:0> to Wd	1	2	None
79	TBLWTH	TBLWTH	Ws,Wd	Write Ws<7:0> to Prog<23:16>	1	2	None
80	TBLWTL	TBLWTL	Ws,Wd	Write Ws to Prog<15:0>	1	2	None
81	ULNK	ULNK		Unlink frame pointer	1	1	None
82	XOR	XOR	f	f = f .XOR. WREG	1	1	N,Z
		XOR	f,WREG	WREG = f .XOR. WREG	1	1	N,Z
		XOR	#lit10,Wn	Wd = lit10 .XOR. Wd	1	1	N,Z
		XOR	Wb,Ws,Wd	Wd = Wb .XOR. Ws	1	1	N,Z
		XOR	Wb,#lit5,Wd	Wd = Wb .XOR. lit5	1	1	N,Z
83	ZE	ZE	Ws,Wnd	Wnd = Zero-extend Ws	1	1	C,Z,N

DC CHA	ARACTER	ISTICS	Standard ( (unless ot Operating	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	Min Typ <sup>(1)</sup> Max Units Conditions								
	Vol	Output Low Voltage <sup>(2)</sup>										
DO10		I/O ports	—	—	0.6	V	Iol = 8.5 mA, Vdd = 5V					
			—	—	0.15	V	IOL = 2.0 mA, VDD = 3V					
DO16		OSC2/CLKO	—	—	0.6	V	IOL = 1.6 mA, VDD = 5V					
		(RC or EC Osc mode)	—	—	0.72	V	IOL = 2.0 mA, VDD = 3V					
	Voh	Output High Voltage <sup>(2)</sup>										
DO20		I/O ports	Vdd - 0.7	—	—	V	Iон = -3.0 mA, Vdd = 5V					
			Vdd - 0.2	—	—	V	Iон = -2.0 mA, Vdd = 3V					
DO26		OSC2/CLKO	Vdd - 0.7	—	—	V	IOH = -1.3 mA, VDD = 5V					
		(RC or EC Osc mode)	Vdd - 0.1	—	—	V	Iон = -2.0 mA, Vdd = 3V					
		Capacitive Loading Specs on Output Pins <sup>(2)</sup>										
DO50	Cosc2	OSC2/SOSC2 pin	_	_	15	pF	In XTL, XT, HS and LP modes when external clock is used to drive OSC1.					
DO56	Сю	All I/O pins and OSC2	—	—	50	pF	RC or EC Osc mode					
DO58	Св	SCL, SDA	_	—	400	pF	In I <sup>2</sup> C mode					
Note 1:	Data in	"Typ" column is at 5V, 25°C unle	ess otherwis	e stated	. Param	eters ar	e for design guidance only and					

### TABLE 20-9: DC CHARACTERISTICS: I/O PIN OUTPUT SPECIFICATIONS

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

# 28-Lead Plastic Small Outline (SO) – Wide, 7.50 mm Body [SOIC]

**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 Pins	N		28				
Pitch	е		1.27 BSC				
Overall Height	A	-	-	2.65			
Molded Package Thickness	A2	2.05	-	-			
Standoff §	A1	0.10	_	0.30			
Overall Width	E		10.30 BSC				
Molded Package Width	E1	7.50 BSC					
Overall Length	D		17.90 BSC				
Chamfer (optional)	h	0.25	-	0.75			
Foot Length	L	0.40	-	1.27			
Footprint	L1		1.40 REF				
Foot Angle Top	ф	0°	-	8°			
Lead Thickness	С	0.18	-	0.33			
Lead Width	b	0.31	-	0.51			
Mold Draft Angle Top	α	5°	_	15°			
Mold Draft Angle Bottom	β	5°	-	15°			

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.

2. § Significant Characteristic.

3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.15 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 C04-052B