



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.

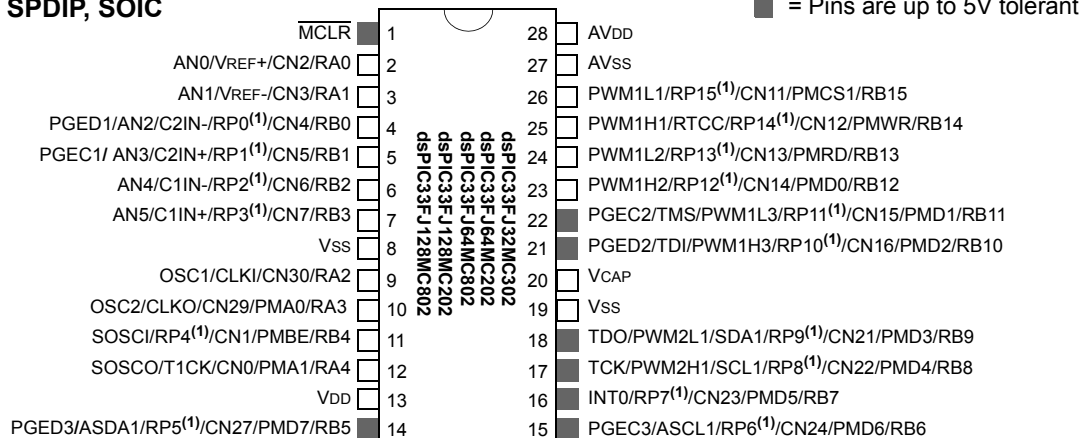
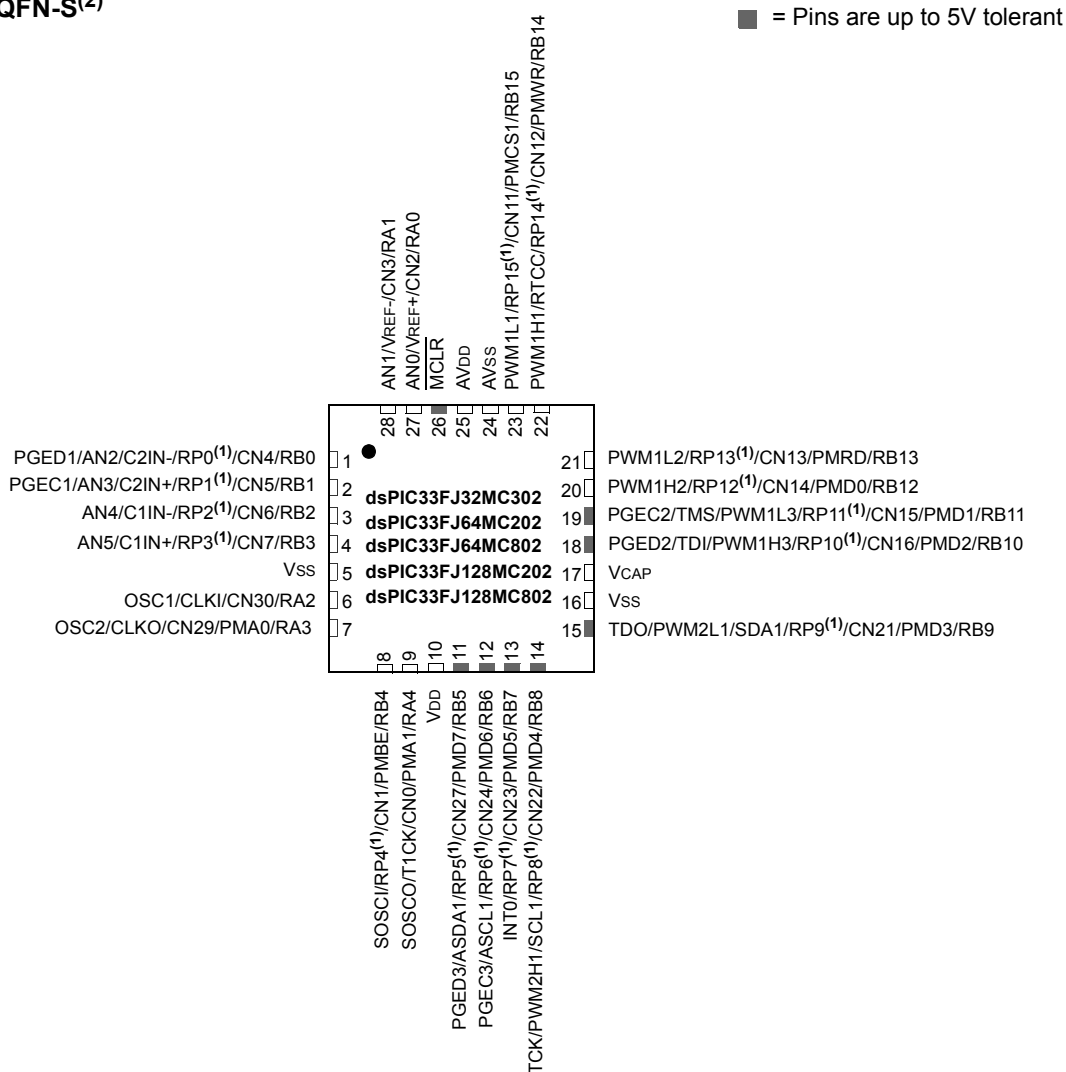
Applications of "[Embedded - Microcontrollers](#)"

Details

Product Status	Active
Core Processor	dsPIC
Core Size	16-Bit
Speed	40 MIPS
Connectivity	I ² C, IrDA, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, Motor Control PWM, POR, PWM, QEI, WDT
Number of I/O	35
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 9x10b/12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-TQFP
Supplier Device Package	44-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/dspic33fj64mc204t-i-pt

Pin Diagrams

28-Pin SPDIP, SOIC

28-Pin QFN-S⁽²⁾

Note 1: The RPx pins can be used by any remappable peripheral. See [Table 1](#) in this section for the list of available peripherals.

Note 2: The metal plane at the bottom of the device is not connected to any pins and is recommended to be connected to VSS externally.

Table of Contents

dsPIC33FJ32MC302/304, dsPIC33FJ64MCX02/X04 and dsPIC33FJ128MCX02/X04 Product Families.....	2
1.0 Device Overview	11
2.0 Guidelines for Getting Started with 16-bit Digital Signal Controllers	17
3.0 CPU	21
4.0 Memory Organization	35
5.0 Flash Program Memory	73
6.0 Resets	79
7.0 Interrupt Controller	89
8.0 Direct Memory Access (DMA)	131
9.0 Oscillator Configuration	143
10.0 Power-Saving Features	155
11.0 I/O Ports	163
12.0 Timer1	195
13.0 Timer2/3 And Timer4/5	199
14.0 Input Capture	205
15.0 Output Compare	209
16.0 Motor Control PWM Module	213
17.0 Quadrature Encoder Interface (QEI) Module	227
18.0 Serial Peripheral Interface (SPI)	233
19.0 Inter-Integrated Circuit™ (I ² C™)	239
20.0 Universal Asynchronous Receiver Transmitter (UART)	247
21.0 Enhanced CAN (ECAN™) Module	253
22.0 10-bit/12-bit Analog-to-Digital Converter (ADC1)	281
23.0 Audio Digital-to-Analog Converter (DAC)	297
24.0 Comparator Module	303
25.0 Real-Time Clock and Calendar (RTCC)	309
26.0 Programmable Cyclic Redundancy Check (CRC) Generator	321
27.0 Parallel Master Port (PMP)	327
28.0 Special Features	335
29.0 Instruction Set Summary	345
30.0 Development Support	353
31.0 Electrical Characteristics	357
32.0 High Temperature Electrical Characteristics	413
32.0 DC and AC Device Characteristics Graphs	425
33.0 Packaging Information	429
Appendix A: Revision History	439
Index	449
The Microchip Web Site	455
Customer Change Notification Service	455
Customer Support	455
Reader Response	456
Product Identification System	457

The Overflow and Saturation Status bits can optionally be viewed in the STATUS Register (SR) as the logical OR of OA and OB (in bit OAB) and the logical OR of SA and SB (in bit SAB). Programmers can check one bit in the STATUS register to determine if either accumulator has overflowed, or one bit to determine if either accumulator has saturated. This is useful for complex number arithmetic, which typically uses both accumulators.

The device supports three Saturation and Overflow modes:

- **Bit 39 Overflow and Saturation:**
When bit 39 overflow and saturation occurs, the saturation logic loads the maximally positive 9.31 (0x7FFFFFFF) or maximally negative 9.31 value (0x80000000) into the target accumulator. The SA or SB bit is set and remains set until cleared by the user application. This condition is referred to as super saturation and provides protection against erroneous data or unexpected algorithm problems (such as gain calculations).
- **Bit 31 Overflow and Saturation:**
When bit 31 overflow and saturation occurs, the saturation logic then loads the maximally positive 1.31 value (0x007FFFFFFF) or maximally negative 1.31 value (0x0080000000) into the target accumulator. The SA or SB bit is set and remains set until cleared by the user application. When this Saturation mode is in effect, the guard bits are not used, so the OA, OB or OAB bits are never set.
- **Bit 39 Catastrophic Overflow:**
The bit 39 Overflow Status bit from the adder is used to set the SA or SB bit, which remains set until cleared by the user application. No saturation operation is performed, and the accumulator is allowed to overflow, destroying its sign. If the COVTE bit in the INTCON1 register is set, a catastrophic overflow can initiate a trap exception.

3.8.3 ACCUMULATOR WRITE BACK

The MAC class of instructions (with the exception of MPY, MPY.N, ED and EDAC) can optionally write a rounded version of the high word (bits 31 through 16) of the accumulator that is not targeted by the instruction into data space memory. The write is performed across the X bus into combined X and Y address space. The following addressing modes are supported:

- **W13, Register Direct:**
The rounded contents of the non-target accumulator are written into W13 as a 1.15 fraction.
- **[W13] + = 2, Register Indirect with Post-Increment:**
The rounded contents of the non-target accumulator are written into the address pointed to by W13 as a 1.15 fraction. W13 is then incremented by 2 (for a word write).

3.8.3.1 Round Logic

The round logic is a combinational block that performs a conventional (biased) or convergent (unbiased) round function during an accumulator write (store). The Round mode is determined by the state of the RND bit in the CORCON register. It generates a 16-bit, 1.15 data value that is passed to the data space write saturation logic. If rounding is not indicated by the instruction, a truncated 1.15 data value is stored, and the least significant word is simply discarded.

Conventional rounding zero-extends bit 15 of the accumulator and adds it to the ACCxH word (bits 16 through 31 of the accumulator).

- If the ACCxL word (bits 0 through 15 of the accumulator) is between 0x8000 and 0xFFFF (0x8000 included), ACCxH is incremented.
- If ACCxL is between 0x0000 and 0x7FFF, ACCxH is left unchanged.

A consequence of this algorithm is that over a succession of random rounding operations, the value tends to be biased slightly positive.

Convergent (or unbiased) rounding operates in the same manner as conventional rounding, except when ACCxL equals 0x8000. In this case, the Least Significant bit (bit 16 of the accumulator) of ACCxH is examined:

- If it is '1', ACCxH is incremented.
- If it is '0', ACCxH is not modified.

Assuming that bit 16 is effectively random in nature, this scheme removes any rounding bias that may accumulate.

The SAC and SAC.R instructions store either a truncated (SAC), or rounded (SAC.R) version of the contents of the target accumulator to data memory via the X bus, subject to data saturation (see [Section 3.8.3.2 "Data Space Write Saturation"](#)). For the MAC class of instructions, the accumulator write-back operation functions in the same manner, addressing combined MCU (X and Y) data space though the X bus. For this class of instructions, the data is always subject to rounding.

4.4 Special Function Register Maps

TABLE 4-1: CPU CORE REGISTERS 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	All Resets
WREG0	0000	Working Register 0																0000
WREG1	0002	Working Register 1																0000
WREG2	0004	Working Register 2																0000
WREG3	0006	Working Register 3																0000
WREG4	0008	Working Register 4																0000
WREG5	000A	Working Register 5																0000
WREG6	000C	Working Register 6																0000
WREG7	000E	Working Register 7																0000
WREG8	0010	Working Register 8																0000
WREG9	0012	Working Register 9																0000
WREG10	0014	Working Register 10																0000
WREG11	0016	Working Register 11																0000
WREG12	0018	Working Register 12																0000
WREG13	001A	Working Register 13																0000
WREG14	001C	Working Register 14																0000
WREG15	001E	Working Register 15																0800
SPLIM	0020	Stack Pointer Limit Register																xxxx
ACCAL	0022	ACCAL																xxxx
ACCAH	0024	ACCAH																xxxx
ACCAU	0026	ACCA<39>								ACCAU								xxxx
ACCBL	0028	ACCBL																xxxx
ACCBH	002A	ACCBH																xxxx
ACCBU	002C	ACCB<39>								ACCBU								xxxx
PCL	002E	Program Counter Low Word Register																xxxx
PCH	0030	—	—	—	—	—	—	—	—	Program Counter High Byte Register								0000
TBLPAG	0032	—	—	—	—	—	—	—	—	Table Page Address Pointer Register								0000
PSVPAG	0034	—	—	—	—	—	—	—	—	Program Memory Visibility Page Address Pointer Register								0000
RCOUNT	0036	Repeat Loop Counter Register																xxxx
DCOUNT	0038	DCOUNT<15:0>																xxxx
DOSTARTL	003A	DOSTARTL<15:1>															0	xxxx
DOSTARTH	003C	—	—	—	—	—	—	—	—	—	—	DOSTARTH<5:0>						00xx
DOENDL	003E	DOENDL<15:1>															0	xxxx
DOENDH	0040	—	—	—	—	—	—	—	—	—	—	DOENDH<5:0>						00xx
SR	0042	OA	OB	SA	SB	OAB	SAB	DA	DC	IPL2	IPL1	IPL0	RA	N	OV	Z	C	0000
CORCON	0044	—	—	—	US	EDT	DL<2:0>			SATA	SATB	SATDW	ACCSAT	IPL3	PSV	RND	IF	0020

Legend: x = unknown value on Reset, — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

4.4.1 SOFTWARE STACK

In addition to its use as a working register, the W15 register in the dsPIC33FJ32MC302/304, dsPIC33FJ64MCX02/X04 and dsPIC33FJ128MCX02/X04 devices is also used as a software Stack Pointer. The Stack Pointer always points to the first available free word and grows from lower to higher addresses. It pre-decrements for stack pops and post-increments for stack pushes, as shown in Figure 4-6. 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 concatenates the SRL register to the MSb of the PC prior to the push.

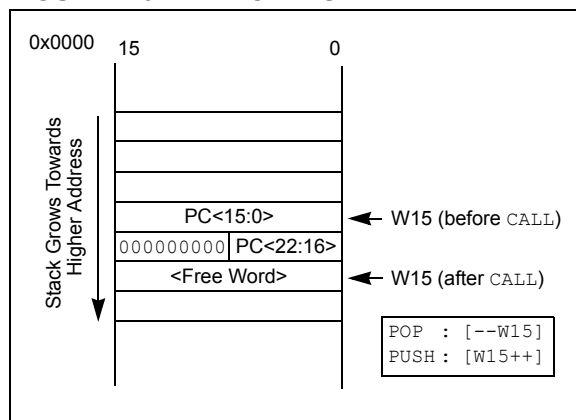
The Stack Pointer Limit register (SPLIM) associated with the Stack Pointer sets an upper address boundary for the stack. The SPLIM is uninitialized at Reset. As is the case for the Stack Pointer, the SPLIM<0> is forced to '0' because all stack operations must be word aligned.

Whenever an EA is generated using the W15 as a source or destination pointer, the resulting address is compared with the value in the SPLIM register. If the contents of the Stack Pointer (W15) and the SPLIM register are equal and a push operation is performed, a stack error trap does not occur. The stack error trap occurs on a subsequent push operation. For example, 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. This prevents 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 4-6: CALL STACK FRAME



4.4.2 DATA RAM PROTECTION FEATURE

The dsPIC33F product family supports Data RAM protection features that enable segments of RAM to be protected when used in conjunction with Boot and Secure Code Segment Security. The BSRAM (Secure RAM segment for BS) is accessible only from the Boot Segment Flash code when enabled. The SSRAM (Secure RAM segment for RAM) is accessible only from the Secure Segment Flash code when enabled. See Table 4-1 for an overview of the BSRAM and SSRAM SFRs.

4.5 Instruction Addressing Modes

The addressing modes shown in Table 4-40 form the basis of the addressing modes optimized to support the specific features of individual instructions. The addressing modes provided in the MAC class of instructions differ from those in the other instruction types.

4.5.1 FILE REGISTER INSTRUCTIONS

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

4.5.2 MCU INSTRUCTIONS

The three-operand MCU instructions are of the form:

Operand 3 = Operand 1 <function> Operand 2

where:

Operand 1 is always a working register (that is, the addressing mode can only be register direct), which is referred to as Wb.

Operand 2 can be a W register, fetched from data memory, or a 5-bit literal. The result location can be either a W register or a data memory location. The following addressing modes are supported by MCU instructions:

- Register Direct
- Register Indirect
- Register Indirect Post-Modified
- Register Indirect Pre-Modified
- 5-bit or 10-bit Literal

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

TABLE 4-40: FUNDAMENTAL ADDRESSING MODES SUPPORTED

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

4.5.3 MOVE AND ACCUMULATOR INSTRUCTIONS

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

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

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

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

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

4.5.4 MAC INSTRUCTIONS

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

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

Note: Register Indirect with Register Offset Addressing mode is available only for W9 (in X space) and W11 (in Y space).

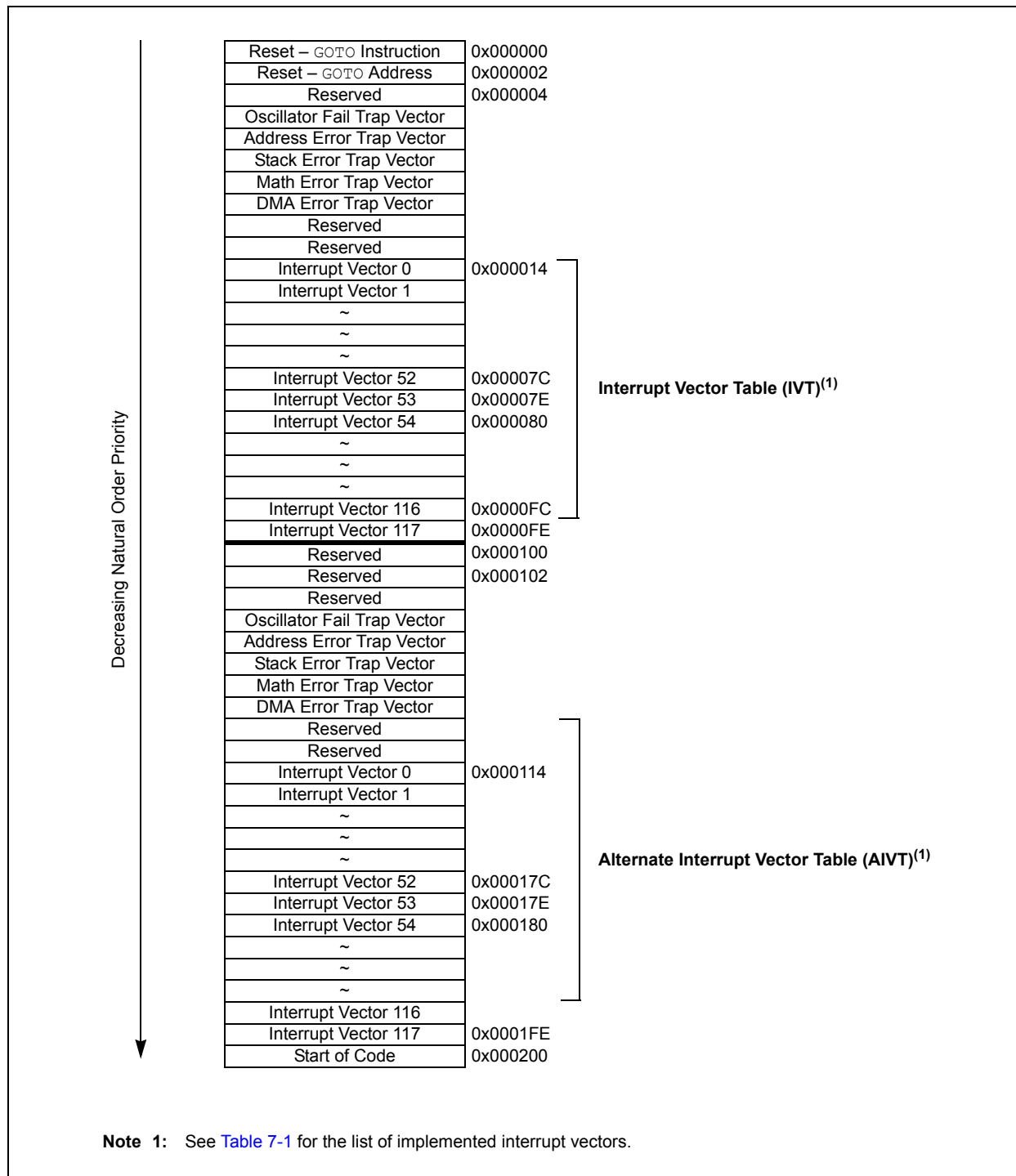
In summary, the following addressing modes are supported by the `MAC` class of instructions:

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

4.5.5 OTHER INSTRUCTIONS

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

FIGURE 7-1: dsPIC33FJ32MC302/304, dsPIC33FJ64MCX02/X04 AND dsPIC33FJ128MCX02/X04 INTERRUPT VECTOR TABLE



REGISTER 7-30: IPC18: INTERRUPT PRIORITY CONTROL REGISTER 18

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
—	QE12IP<2:0>			—	FLTA2IP<2:0>		
bit 15				bit 8			

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	PWM2IP<2:0>			—	—	—	—
bit 7				bit 0			

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 15 **Unimplemented:** Read as '0'

bit 14-12 **QE12IP<2:0>:** QE12 Interrupt Priority bits

111 = Interrupt is priority 7 (highest priority interrupt)

•
•
•

001 = Interrupt is priority 1

000 = Interrupt source is disabled

bit 11 **Unimplemented:** Read as '0'

bit 10-8 **FLTA2IP<2:0>:** PWM2 Fault A Interrupt Priority bits

111 = Interrupt is priority 7 (highest priority interrupt)

•
•
•

001 = Interrupt is priority 1

000 = Interrupt source is disabled

bit 7 **Unimplemented:** Read as '0'

bit 6-4 **PWM2IP<2:0>:** PWM2 Interrupt Priority bits

111 = Interrupt is priority 7 (highest priority interrupt)

•
•
•

001 = Interrupt is priority 1

000 = Interrupt source is disabled

bit 3-0 **Unimplemented:** Read as '0'

REGISTER 10-1: PMD1: PERIPHERAL MODULE DISABLE CONTROL REGISTER 1 (CONTINUED)

bit 2	Unimplemented: Read as '0'
bit 1	C1MD: ECAN1 Module Disable bit 1 = ECAN1 module is disabled 0 = ECAN1 module is enabled
bit 0	AD1MD: ADC1 Module Disable bit 1 = ADC1 module is disabled 0 = ADC1 module is enabled

15.2 Output Compare Resources

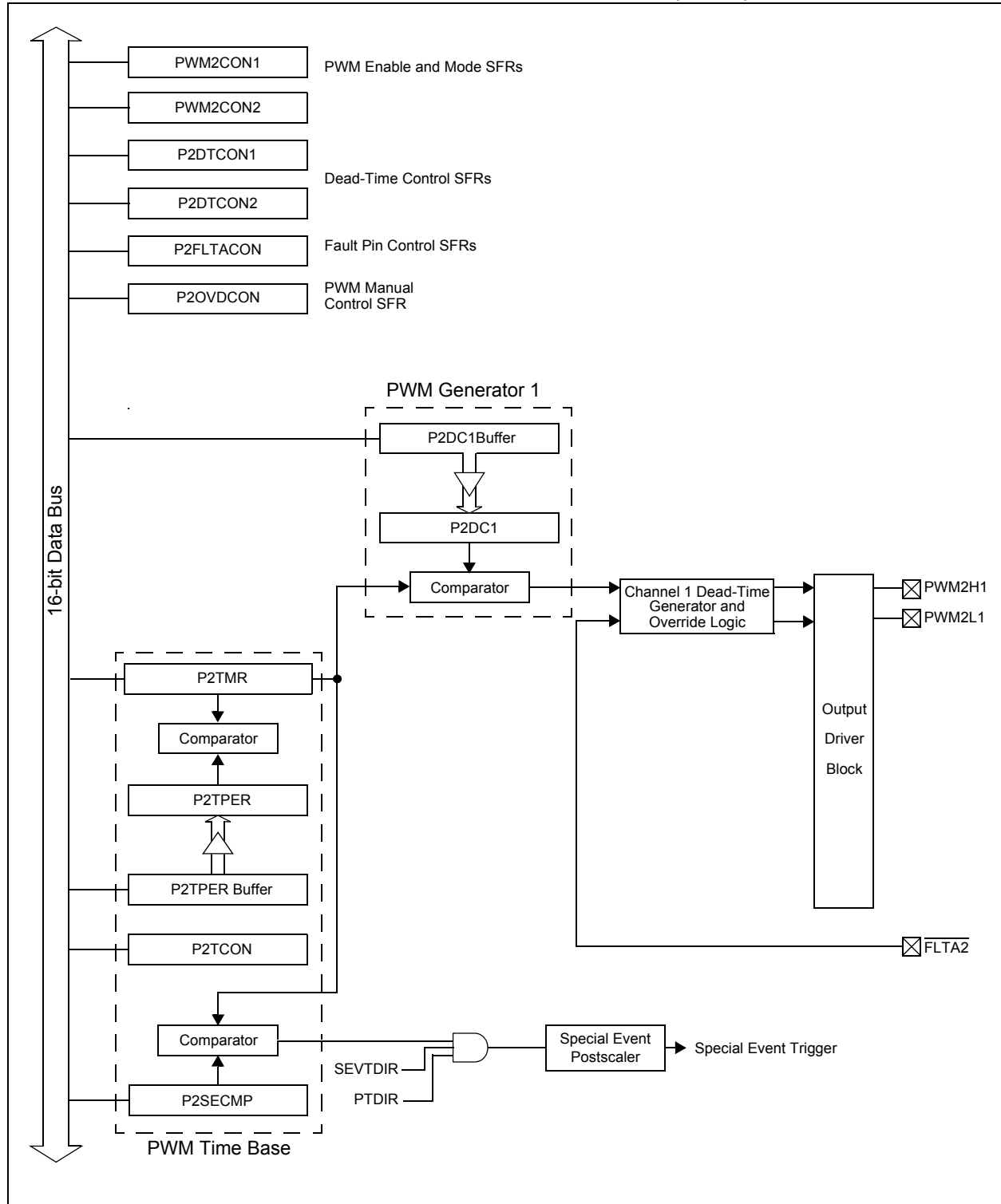
Many useful resources related to Output Compare are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page, which can be accessed using this [link](http://www.microchip.com/wwwproducts/Devices.aspx?dDocName=en532315), contains the latest updates and additional information.

<p>Note: In the event you are not able to access the product page using the link above, enter this URL in your browser: http://www.microchip.com/wwwproducts/Devices.aspx?dDocName=en532315</p>
--

15.2.1 KEY RESOURCES

- **Section 13. “Output Compare”** (DS70209)
- Code Samples
- Application Notes
- Software Libraries
- Webinars
- All related dsPIC33F/PIC24H Family Reference Manuals Sections
- Development Tools

FIGURE 16-2: 2-CHANNEL PWM MODULE BLOCK DIAGRAM (PWM2)



REGISTER 16-5: PWMxCON1: PWM CONTROL REGISTER 1⁽²⁾

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	PMOD3	PMOD2	PMOD1
bit 15					bit 8		

U-0	R/W-1	R/W-1	R/W-1	U-0	R/W-1	R/W-1	R/W-1
—	PEN3H ⁽¹⁾	PEN2H ⁽¹⁾	PEN1H ⁽¹⁾	—	PEN3L ⁽¹⁾	PEN2L ⁽¹⁾	PEN1L ⁽¹⁾
bit 7					bit 0		

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 15-11 **Unimplemented:** Read as '0'

bit 10-8 **PMOD3:PMOD1:** PWM I/O Pair Mode bits

1 = PWM I/O pin pair is in the Independent PWM Output mode

0 = PWM I/O pin pair is in the Complementary Output mode

bit 7 **Unimplemented:** Read as '0'

bit 6-4 **PEN3H:PEN1H:** PWMxH I/O Enable bits⁽¹⁾

1 = PWMxH pin is enabled for PWM output

0 = PWMxH pin disabled, I/O pin becomes general purpose I/O

bit 3 **Unimplemented:** Read as '0'

bit 2-0 **PEN3L:PEN1L:** PWMxL I/O Enable bits⁽¹⁾

1 = PWMxL pin is enabled for PWM output

0 = PWMxL pin disabled, I/O pin becomes general purpose I/O

Note 1: Reset condition of the PENxH and PENxL bits depends on the value of the PWMPIN Configuration bit in the FPOR Configuration register.

2: PWM2 supports only one PWM I/O pin pair.

REGISTER 16-9: PxFLTACON: FAULT A CONTROL REGISTER⁽¹⁾

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	FAOV3H	FAOV3L	FAOV2H	FAOV2L	FAOV1H	FAOV1L
bit 15						bit 8	

R/W-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
FLTAM	—	—	—	—	FAEN3	FAEN2	FAEN1
bit 7						bit 0	

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 15-14 **Unimplemented:** Read as '0'

bit 13-8 **FAOVxH<3:1>:FAOVxL<3:1>:** Fault Input A PWM Override Value bits

1 = The PWM output pin is driven active on an external Fault input event

0 = The PWM output pin is driven inactive on an external Fault input event

bit 7 **FLTAM:** Fault A Mode bit

1 = The Fault A input pin functions in the Cycle-by-Cycle mode

0 = The Fault A input pin latches all control pins to the programmed states in PxFLTACON<13:8>

bit 6-3 **Unimplemented:** Read as '0'

bit 2 **FAEN3:** Fault Input A Enable bit

1 = PWMxH3/PWMxL3 pin pair is controlled by Fault Input A

0 = PWMxH3/PWMxL3 pin pair is not controlled by Fault Input A

bit 1 **FAEN2:** Fault Input A Enable bit

1 = PWMxH2/PWMxL2 pin pair is controlled by Fault Input A

0 = PWMxH2/PWMxL2 pin pair is not controlled by Fault Input A

bit 0 **FAEN1:** Fault Input A Enable bit

1 = PWMxH1/PWMxL1 pin pair is controlled by Fault Input A

0 = PWMxH1/PWMxL1 pin pair is not controlled by Fault Input A

Note 1: PWM2 supports only one PWM I/O pin pair.

17.2 QEI Control Registers

REGISTER 17-1: QEIXCON: QEIX CONTROL REGISTER (x = 1 or 2)

R/W-0	U-0	R/W-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
CNTERR ⁽¹⁾	—	QEISIDL	INDEX	UPDN ⁽²⁾	QEIM<2:0>		
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SWPAB	PCDOUT	TQGATE	TQCKPS<1:0>		POSRES	TQCS	UPDN_SRC
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

- bit 15 **CNTERR:** Count Error Status Flag bit⁽¹⁾
 1 = Position count error has occurred
 0 = No position count error has occurred
- bit 14 **Unimplemented:** Read as '0'
- bit 13 **QEISIDL:** Stop in Idle Mode bit
 1 = Discontinue module operation when device enters Idle mode
 0 = Continue module operation in Idle mode
- bit 12 **INDEX:** Index Pin State Status bit (read-only)
 1 = Index pin is High
 0 = Index pin is Low
- bit 11 **UPDN:** Position Counter Direction Status bit⁽²⁾
 1 = Position Counter Direction is positive (+)
 0 = Position Counter Direction is negative (-)
- bit 10-8 **QEIM<2:0>:** Quadrature Encoder Interface Mode Select bits
 111 = Quadrature Encoder Interface enabled (x4 mode) with position counter reset by match (MAXxCNT)
 110 = Quadrature Encoder Interface enabled (x4 mode) with Index Pulse reset of position counter
 101 = Quadrature Encoder Interface enabled (x2 mode) with position counter reset by match (MAXxCNT)
 100 = Quadrature Encoder Interface enabled (x2 mode) with Index Pulse reset of position counter
 011 = Unused (Module disabled)
 010 = Unused (Module disabled)
 001 = Starts 16-bit Timer
 000 = Quadrature Encoder Interface/Timer off
- bit 7 **SWPAB:** Phase A and Phase B Input Swap Select bit
 1 = Phase A and Phase B inputs swapped
 0 = Phase A and Phase B inputs not swapped
- bit 6 **PCDOUT:** Position Counter Direction State Output Enable bit
 1 = Position Counter Direction Status Output Enable (QEI logic controls state of I/O pin)
 0 = Position Counter Direction Status Output Disabled (Normal I/O pin operation)

Note 1: This bit only applies when QEIM<2:0> = '110' or '100'.

2: Read-only bit when QEIM<2:0> = '1xx'. Read/write bit when QEIM<2:0> = '001'.

3: Prescaler utilized for 16-bit Timer mode only.

4: This bit applies only when QEIM<2:0> = 100 or 110.

5: When configured for QEI mode, this control bit is a 'don't care'.

REGISTER 18-2: SPIxCON1: SPIx CONTROL REGISTER 1

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	DISSCK	DISSDO	MODE16	SMP	CKE ⁽¹⁾
bit 15							
							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SSEN ⁽³⁾	CKP	MSTEN	SPRE<2:0> ⁽²⁾			PPRE<1:0> ⁽²⁾	
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 15-13 **Unimplemented:** Read as '0'

bit 12 **DISSCK:** Disable SCKx Pin bit (SPI Master modes only)

1 = Internal SPI clock is disabled, pin functions as I/O

0 = Internal SPI clock is enabled

bit 11 **DISSDO:** Disable SDOx Pin bit

1 = SDOx pin is not used by module; pin functions as I/O

0 = SDOx pin is controlled by the module

bit 10 **MODE16:** Word/Byte Communication Select bit

1 = Communication is word-wide (16 bits)

0 = Communication is byte-wide (8 bits)

bit 9 **SMP:** SPIx Data Input Sample Phase bit

Master mode:

1 = Input data sampled at end of data output time

0 = Input data sampled at middle of data output time

Slave mode:

SMP must be cleared when SPIx is used in Slave mode.

bit 8 **CKE:** SPIx Clock Edge Select bit⁽¹⁾

1 = Serial output data changes on transition from active clock state to Idle clock state (see bit 6)

0 = Serial output data changes on transition from Idle clock state to active clock state (see bit 6)

bit 7 **SSEN:** Slave Select Enable bit (Slave mode)⁽³⁾

1 = \overline{SSx} pin used for Slave mode

0 = \overline{SSx} pin not used by module. Pin controlled by port function

bit 6 **CKP:** Clock Polarity Select bit

1 = Idle state for clock is a high level; active state is a low level

0 = Idle state for clock is a low level; active state is a high level

bit 5 **MSTEN:** Master Mode Enable bit

1 = Master mode

0 = Slave mode

Note 1: This bit is not used in Framed SPI modes. Program this bit to '0' for the Framed SPI modes (FRMEN = 1).

2: Do not set both Primary and Secondary prescalers to a value of 1:1.

3: This bit must be cleared when FRMEN = 1.

REGISTER 25-3: ALCFGRPT: ALARM CONFIGURATION REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ALRMEN	CHIME	AMASK<3:0>				ALRMPTR<1:0>	
bit 15						bit 8	

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ARPT<7:0>							
bit 7							bit 0

Legend:

R = Readable bit W = Writable bit U = Unimplemented bit, read as '0'
 -n = Value at POR '1' = Bit is set '0' = Bit is cleared x = Bit is unknown

- bit 15 **ALRMEN:** Alarm Enable bit
 1 = Alarm is enabled (cleared automatically after an alarm event whenever ARPT<7:0> = 0x00 and CHIME = 0)
 0 = Alarm is disabled
- bit 14 **CHIME:** Chime Enable bit
 1 = Chime is enabled; ARPT<7:0> bits are allowed to roll over from 0x00 to 0xFF
 0 = Chime is disabled; ARPT<7:0> bits stop once they reach 0x00
- bit 13-10 **AMASK<3:0>:** Alarm Mask Configuration bits
 11xx = Reserved – do not use
 101x = Reserved – do not use
 1001 = Once a year (except when configured for February 29th, once every 4 years)
 1000 = Once a month
 0111 = Once a week
 0110 = Once a day
 0101 = Every hour
 0100 = Every 10 minutes
 0011 = Every minute
 0010 = Every 10 seconds
 0001 = Every second
 0000 = Every half second
- bit 9-8 **ALRMPTR<1:0>:** Alarm Value Register Window Pointer bits
 Points to the corresponding Alarm Value registers when reading ALRMVALH and ALRMVALL registers; the ALRMPTR<1:0> value decrements on every read or write of ALRMVALH until it reaches '00'.
ALRMVAL<15:8>:
 11 = Unimplemented
 10 = ALRMMNTH
 01 = ALRMWD
 00 = ALRMMIN
ALRMVAL<7:0>:
 11 = Unimplemented
 10 = ALRMDAY
 01 = ALRMHR
 00 = ALRMSEC
- bit 7-0 **ARPT<7:0>:** Alarm Repeat Counter Value bits
 11111111 = Alarm will repeat 255 more times
 •
 •
 •
 00000000 = Alarm will not repeat
 The counter decrements on any alarm event. The counter is prevented from rolling over from 0x00 to 0xFF unless CHIME = 1.

TABLE 31-17: PLL CLOCK TIMING SPECIFICATIONS (V_{DD} = 3.0V TO 3.6V)

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)				
			Operating temperature -40°C ≤ T _A ≤ +85°C for Industrial -40°C ≤ T _A ≤ +125°C for Extended				
Param No.	Symbol	Characteristic	Min	Typ ⁽¹⁾	Max	Units	Conditions
OS50	FPLLI	PLL Voltage Controlled Oscillator (VCO) Input Frequency Range	0.8	—	8	MHz	ECPLL, XTPLL modes
OS51	FSYS	On-Chip VCO System Frequency	100	—	200	MHz	—
OS52	TLOCK	PLL Start-up Time (Lock Time)	0.9	1.5	3.1	mS	—
OS53	DCLK	CLKO Stability (Jitter) ⁽²⁾	-3	0.5	3	%	Measured over 100 ms period

Note 1: Data in “Typ” column is at 3.3V, 25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

- 2:** These parameters are characterized by similarity, but are not tested in manufacturing. This specification is based on clock cycle by clock cycle measurements. To calculate the effective jitter for individual time bases or communication clocks use this formula:

$$\text{Peripheral Clock Jitter} = \frac{DCLK}{\sqrt{\left(\frac{FOSC}{\text{Peripheral Bit Rate Clock}}\right)}}$$

For example: Fosc = 32 MHz, DCLK = 3%, SPI bit rate clock, (i.e., SCK) is 2 MHz.

$$\text{SPI SCK Jitter} = \left[\frac{DCLK}{\sqrt{\left(\frac{32 \text{ MHz}}{2 \text{ MHz}}\right)}} \right] = \left[\frac{3\%}{\sqrt{16}} \right] = \left[\frac{3\%}{4} \right] = 0.75\%$$

TABLE 31-18: AC CHARACTERISTICS: INTERNAL RC ACCURACY

AC CHARACTERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)					
		Operating temperature -40°C ≤TA ≤+85°C for industrial 					

Note 1: Frequency calibrated at 25°C and 3.3V. TUN bits can be used to compensate for temperature drift.

TABLE 31-19: INTERNAL RC ACCURACY

AC CHARACTERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended					
Param No.	Characteristic	Min	Typ	Max	Units	Conditions	
	LPRC @ 32.768 kHz ⁽¹⁾						
F21	LPRC	-20	±6	+20	%	-40°C ≤ TA ≤ +85°C	VDD = 3.0-3.6V
	LPRC	-30	—	+30	%	-40°C ≤ TA ≤ +125°C	VDD = 3.0-3.6V

Note 1: Change of LPRC frequency as V_{DD} changes.

TABLE 32-6: DC CHARACTERISTICS: I/O PIN OUTPUT SPECIFICATIONS

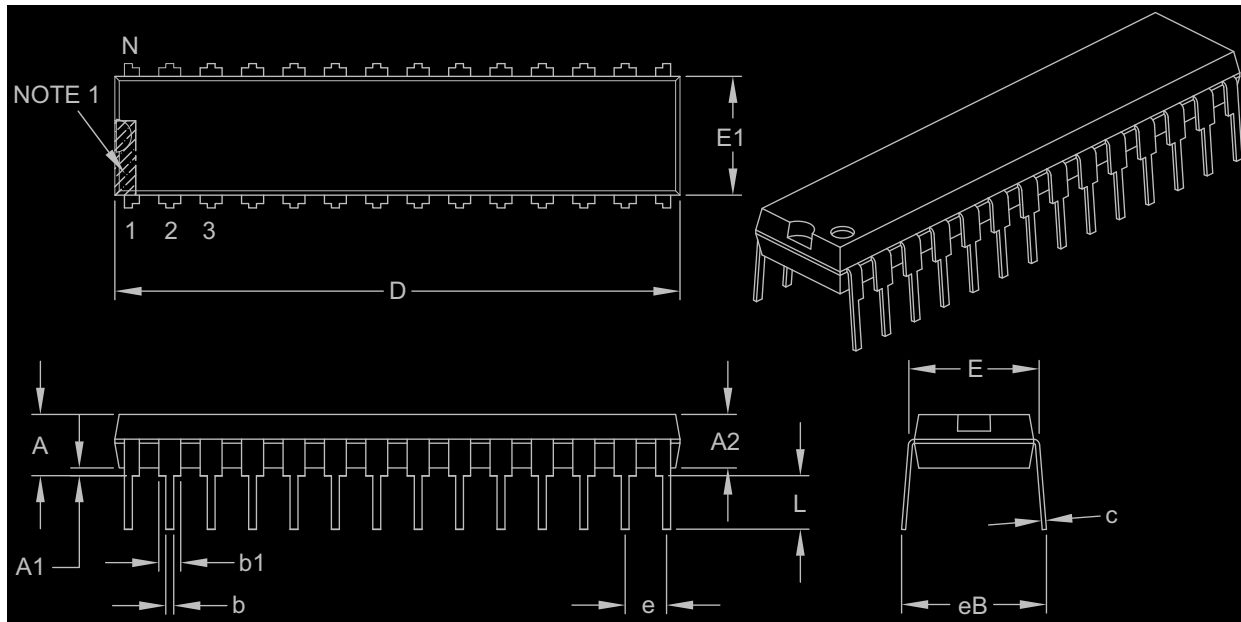
DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +150^{\circ}\text{C}$ for High Temperature				
Param.	Symbol	Characteristic	Min.	Typ.	Max.	Units	Conditions
DO10	VOL	Output Low Voltage I/O Pins: 2x Sink Driver Pins - RA2, RA7-RA10, RB10, RB11, RB7, RB4, RC3-RC9	—	—	0.4	V	$I_{OL} \leq 1.8 \text{ mA}$, $V_{DD} = 3.3\text{V}$ See Note 1
		Output Low Voltage I/O Pins: 4x Sink Driver Pins - RA0, RA1, RB0-RB3, RB5, RB6, RB8, RB9, RB12-RB15, RC0-RC2	—	—	0.4	V	$I_{OL} \leq 3.6 \text{ mA}$, $V_{DD} = 3.3\text{V}$ See Note 1
		Output Low Voltage I/O Pins: 8x Sink Driver Pins - RA3, RA4	—	—	0.4	V	$I_{OL} \leq 6 \text{ mA}$, $V_{DD} = 3.3\text{V}$ See Note 1
DO20	VOH	Output High Voltage I/O Pins: 2x Source Driver Pins - RA2, RA7-RA10, RB4, RB7, RB10, RB11, RC3-RC9	2.4	—	—	V	$I_{OL} \geq -1.8 \text{ mA}$, $V_{DD} = 3.3\text{V}$ See Note 1
		Output High Voltage I/O Pins: 4x Source Driver Pins - RA0, RA1, RB0-RB3, RB5, RB6, RB8, RB9, RB12-RB15, RC0-RC2	2.4	—	—	V	$I_{OL} \geq -3 \text{ mA}$, $V_{DD} = 3.3\text{V}$ See Note 1
		Output High Voltage I/O Pins: 8x Source Driver Pins - RA4, RA3	2.4	—	—	V	$I_{OL} \geq -6 \text{ mA}$, $V_{DD} = 3.3\text{V}$ See Note 1
DO20A	VOH1	Output High Voltage I/O Pins: 2x Source Driver Pins - RA2, RA7-RA10, RB4, RB7, RB10, RB11, RC3-RC9	1.5	—	—	V	$I_{OH} \geq -1.9 \text{ mA}$, $V_{DD} = 3.3\text{V}$ See Note 1
			2.0	—	—		$I_{OH} \geq -1.85 \text{ mA}$, $V_{DD} = 3.3\text{V}$ See Note 1
			3.0	—	—		$I_{OH} \geq -1.4 \text{ mA}$, $V_{DD} = 3.3\text{V}$ See Note 1
		Output High Voltage 4x Source Driver Pins - RA0, RA1, RB0-RB3, RB5, RB6, RB8, RB9, RB12-RB15, RC0-RC2	1.5	—	—	V	$I_{OH} \geq -3.9 \text{ mA}$, $V_{DD} = 3.3\text{V}$ See Note 1
			2.0	—	—		$I_{OH} \geq -3.7 \text{ mA}$, $V_{DD} = 3.3\text{V}$ See Note 1
			3.0	—	—		$I_{OH} \geq -2 \text{ mA}$, $V_{DD} = 3.3\text{V}$ See Note 1
		Output High Voltage I/O Pins: 8x Source Driver Pins - RA3, RA4	1.5	—	—	V	$I_{OH} \geq -7.5 \text{ mA}$, $V_{DD} = 3.3\text{V}$ See Note 1
			2.0	—	—		$I_{OH} \geq -6.8 \text{ mA}$, $V_{DD} = 3.3\text{V}$ See Note 1
			3.0	—	—		$I_{OH} \geq -3 \text{ mA}$, $V_{DD} = 3.3\text{V}$ See Note 1

Note 1: Parameters are characterized, but not tested.

33.1 Package Details

28-Lead Skinny Plastic Dual In-Line (SP) – 300 mil Body [SPDIP]

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



Units		INCHES		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N	28		
Pitch	e	.100 BSC		
Top to Seating Plane	A	—	—	.200
Molded Package Thickness	A2	.120	.135	.150
Base to Seating Plane	A1	.015	—	—
Shoulder to Shoulder Width	E	.290	.310	.335
Molded Package Width	E1	.240	.285	.295
Overall Length	D	1.345	1.365	1.400
Tip to Seating Plane	L	.110	.130	.150
Lead Thickness	c	.008	.010	.015
Upper Lead Width	b1	.040	.050	.070
Lower Lead Width	b	.014	.018	.022
Overall Row Spacing §	eB	—	—	.430

Notes:

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- § Significant Characteristic.
- Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" per side.
- Dimensioning and tolerancing per ASME Y14.5M.
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-070B

Using the RCON Status Bits 87

V

Voltage Regulator (On-Chip) 339

W

Watchdog Time-out Reset (WDTR) 86

Watchdog Timer (WDT) 335, 340

 Programming Considerations 340

WWW Address..... 455

WWW, On-Line Support 9