

Welcome to [E-XFL.COM](https://www.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.

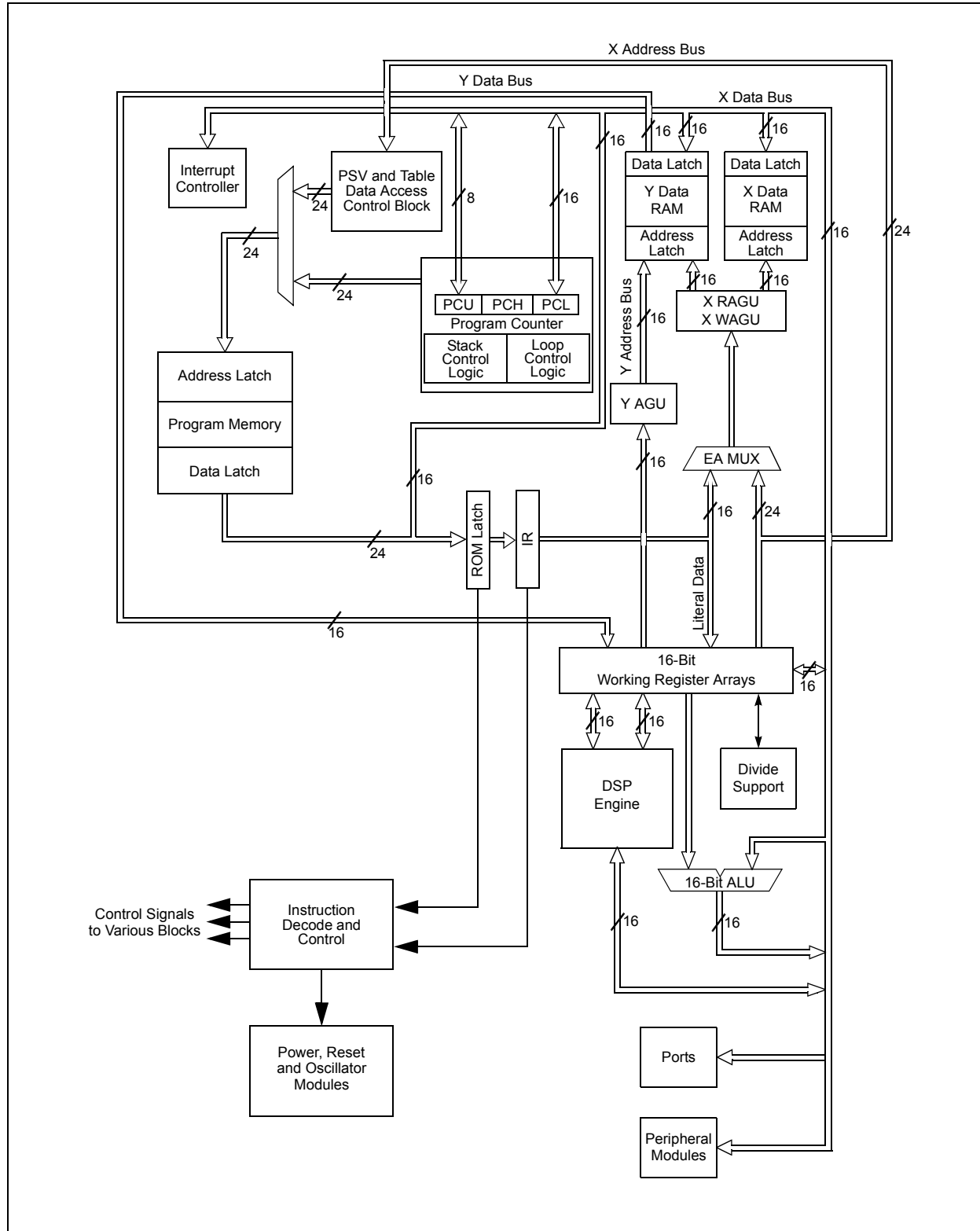
### Applications of "[Embedded - Microcontrollers](#)"

#### Details

Product Status	Active
Core Processor	dsPIC
Core Size	16-Bit
Speed	60 MIPS
Connectivity	I <sup>2</sup> C, IrDA, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	21
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 12x12b; D/A 2x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	28-UQFN Exposed Pad
Supplier Device Package	28-UQFN (6x6)
Purchase URL	<a href="https://www.e-xfl.com/product-detail/microchip-technology/dspic33ep32gs202t-e-mx">https://www.e-xfl.com/product-detail/microchip-technology/dspic33ep32gs202t-e-mx</a>

# dsPIC33EPXXGS202 FAMILY

FIGURE 3-1: dsPIC33EPXXGS202 CPU BLOCK DIAGRAM



## 3.6 CPU Resources

Many useful resources are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page contains the latest updates and additional information.

### 3.6.1 KEY RESOURCES

- “**CPU**” (DS70359) in the “*dsPIC33/PIC24 Family Reference Manual*”
- Code Samples
- Application Notes
- Software Libraries
- Webinars
- All related “*dsPIC33/PIC24 Family Reference Manual*” Sections
- Development Tools

**TABLE 4-14: ADC REGISTER MAP (CONTINUED)**

File 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
ADCON5L	0400	SHRRDY	—	—	—	—	—	C1RDY	C0RDY	SHRPWR	—	—	—	—	—	C1PWR	C0PWR	0000
ADCON5H	0402	—	—	—	—	WARMTIME3	WARMTIME2	WARMTIME1	WARMTIME0	SHRCIE	—	—	—	—	—	C1CIE	C0CIE	0000
ADCAL0L	0404	CAL1RDY	—	—	—	CAL1SKIP	CAL1DIFF	CAL1EN	CAL1RUN	CAL0RDY	—	—	—	CAL0SKIP	CAL0DIFF	CAL0EN	CAL0RUN	0000
ADCAL1H	040A	CSHRRDY	—	—	—	CSHRSKIP	CSHRDIFF	CSHREN	CSHRRUN	—	—	—	—	—	—	—	—	0000
ADCBUF0	040C	ADC Data Buffer 0																0000
ADCBUF1	040E	ADC Data Buffer 1																0000
ADCBUF2	0410	ADC Data Buffer 2																0000
ADCBUF3	0412	ADC Data Buffer 3																0000
ADCBUF4	0414	ADC Data Buffer 4																0000
ADCBUF5	0416	ADC Data Buffer 5																0000
ADCBUF6	0418	ADC Data Buffer 6																0000
ADCBUF7	041A	ADC Data Buffer 7																0000
ADCBUF8	041C	ADC Data Buffer 8																0000
ADCBUF9	041E	ADC Data Buffer 9																0000
ADCBUF10	0420	ADC Data Buffer 10																0000
ADCBUF11	0422	ADC Data Buffer 11																0000
ADCBUF14	0428	ADC Data Buffer 14																0000

**Legend:** — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

# dsPIC33EPXXGS202 FAMILY

**FIGURE 7-1: dsPIC33EPXXGS202 FAMILY INTERRUPT VECTOR TABLE**

<div> Decreasing Natural Order Priority </div> <div> ↓ </div> <div> IVT </div> <div> ↑ </div>	Reset – GOTO Instruction	0x000000	<div> See Table 7-1 for Interrupt Vector Details </div>
	Reset – GOTO Address	0x000002	
	Oscillator Fail Trap Vector	0x000004	
	Address Error Trap Vector	0x000006	
	Generic Hard Trap Vector	0x000008	
	Stack Error Trap Vector	0x00000A	
	Math Error Trap Vector	0x00000C	
	Reserved	0x00000E	
	Generic Soft Trap Vector	0x000010	
	Reserved	0x000012	
	Interrupt Vector 0	0x000014	
	Interrupt Vector 1	0x000016	
	:	:	
	:	:	
	:	:	
	Interrupt Vector 52	0x00007C	
	Interrupt Vector 53	0x00007E	
	Interrupt Vector 54	0x000080	
	:	:	
	:	:	
	:	:	
	Interrupt Vector 116	0x0000FC	
	Interrupt Vector 117	0x0000FE	
	Interrupt Vector 118	0x000100	
	Interrupt Vector 119	0x000102	
	Interrupt Vector 120	0x000104	
	:	:	
	:	:	
	:	:	
	Interrupt Vector 244	0x0001FC	
	Interrupt Vector 245	0x0001FE	
	START OF CODE	0x000200	

# dsPIC33EPXXGS202 FAMILY

**FIGURE 7-2: dsPIC33EPXXGS202 FAMILY ALTERNATE INTERRUPT VECTOR TABLE**

<div style="display: flex; flex-direction: column; align-items: center;"> <div style="writing-mode: vertical-rl; transform: rotate(180deg);">Decreasing Natural Order Priority</div> <div style="writing-mode: vertical-rl; transform: rotate(180deg);">AINT</div> </div>	Reserved	$\text{BSLIM}<12:0>^{(1)} + 0x000000$	<div style="display: flex; flex-direction: column; align-items: center;"> <div style="writing-mode: vertical-rl; transform: rotate(180deg);">See Table 7-1 for Interrupt Vector Details</div> </div>
	Reserved	$\text{BSLIM}<12:0>^{(1)} + 0x000002$	
	Oscillator Fail Trap Vector	$\text{BSLIM}<12:0>^{(1)} + 0x000004$	
	Address Error Trap Vector	$\text{BSLIM}<12:0>^{(1)} + 0x000006$	
	Generic Hard Trap Vector	$\text{BSLIM}<12:0>^{(1)} + 0x000008$	
	Stack Error Trap Vector	$\text{BSLIM}<12:0>^{(1)} + 0x00000A$	
	Math Error Trap Vector	$\text{BSLIM}<12:0>^{(1)} + 0x00000C$	
	Reserved	$\text{BSLIM}<12:0>^{(1)} + 0x00000E$	
	Generic Soft Trap Vector	$\text{BSLIM}<12:0>^{(1)} + 0x000010$	
	Reserved	$\text{BSLIM}<12:0>^{(1)} + 0x000012$	
	Interrupt Vector 0	$\text{BSLIM}<12:0>^{(1)} + 0x000014$	
	Interrupt Vector 1	$\text{BSLIM}<12:0>^{(1)} + 0x000016$	
	:	:	
	:	:	
	:	:	
	:	:	
	Interrupt Vector 52	$\text{BSLIM}<12:0>^{(1)} + 0x00007C$	
	Interrupt Vector 53	$\text{BSLIM}<12:0>^{(1)} + 0x00007E$	
	Interrupt Vector 54	$\text{BSLIM}<12:0>^{(1)} + 0x000080$	
	:	:	
	:	:	
	:	:	
	Interrupt Vector 116	$\text{BSLIM}<12:0>^{(1)} + 0x0000FC$	
	Interrupt Vector 117	$\text{BSLIM}<12:0>^{(1)} + 0x0000FE$	
	Interrupt Vector 118	$\text{BSLIM}<12:0>^{(1)} + 0x000100$	
	Interrupt Vector 119	$\text{BSLIM}<12:0>^{(1)} + 0x000102$	
	Interrupt Vector 120	$\text{BSLIM}<12:0>^{(1)} + 0x000104$	
	:	:	
	:	:	
	:	:	
	Interrupt Vector 244	$\text{BSLIM}<12:0>^{(1)} + 0x0001FC$	
	Interrupt Vector 245	$\text{BSLIM}<12:0>^{(1)} + 0x0001FE$	

**Note 1:** The address depends on the size of the Boot Segment defined by BSLIM<12:0>.  $[(\text{BSLIM}<12:0> - 1) \times 0x400] + \text{Offset}$ .

# dsPIC33EPXXGS202 FAMILY

---

## REGISTER 8-1: OSCCON: OSCILLATOR CONTROL REGISTER<sup>(1)</sup> (CONTINUED)

bit 4	<b>Unimplemented:</b> Read as '0'
bit 3	<b>CF:</b> Clock Fail Detect bit <sup>(3)</sup> 1 = FSCM has detected a clock failure 0 = FSCM has not detected a clock failure
bit 2-1	<b>Unimplemented:</b> Read as '0'
bit 0	<b>OSWEN:</b> Oscillator Switch Enable bit 1 = Requests oscillator switch to selection specified by the NOSC<2:0> bits 0 = Oscillator switch is complete

- Note 1:** Writes to this register require an unlock sequence.
- 2:** Direct clock switches between any Primary Oscillator mode with PLL and FRCPLL mode are not permitted. This applies to clock switches in either direction. In these instances, the application must switch to FRC mode as a transitional clock source between the two PLL modes.
- 3:** This bit should only be cleared in software. Setting the bit in software (= 1) will have the same effect as an actual oscillator failure and trigger an oscillator failure trap.

# dsPIC33EPXXGS202 FAMILY

## REGISTER 9-4: PMD6: PERIPHERAL MODULE DISABLE CONTROL REGISTER 6

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	PWM3MD	PWM2MD	PWM1MD
bit 15					bit 8		

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

bit 10 **PWM3MD:** PWM3 Module Disable bit

1 = PWM3 module is disabled

0 = PWM3 module is enabled

bit 9 **PWM2MD:** PWM2 Module Disable bit

1 = PWM2 module is disabled

0 = PWM2 module is enabled

bit 8 **PWM1MD:** PWM1 Module Disable bit

1 = PWM1 module is disabled

0 = PWM1 module is enabled

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



# dsPIC33EPXXGS202 FAMILY

## REGISTER 15-27: PWMCAPx: PWMx PRIMARY TIME BASE CAPTURE REGISTER

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
PWMCAP<12:5> <sup>(1,2,3,4)</sup>							
bit 15							bit 8

R-0	R-0	R-0	R-0	R-0	U-0	U-0	U-0
PWMCAP<4:0> <sup>(1,2,3,4)</sup>					—	—	—
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 15-3 **PWMCAP<12:0>:** Captured PWMx Time Base Value bits<sup>(1,2,3,4)</sup>

The value in this register represents the captured PWMx time base value when a leading edge is detected on the current-limit input.

bit 2-0 **Unimplemented:** Read as '0'

**Note 1:** The capture feature is only available on a primary output (PWMxH).

**2:** This feature is active only after LEB processing on the current-limit input signal is complete.

**3:** The minimum capture resolution is 8.32 ns.

**4:** This feature can be used when the XPRES bit (PWMCONx<1>) is set to '0'.

# dsPIC33EPXXGS202 FAMILY

---

NOTES:

# dsPIC33EPXXGS202 FAMILY

---

## REGISTER 17-1: I2C1CONL: I2C1 CONTROL REGISTER LOW (CONTINUED)

- bit 6      **STREN:** SCL1 Clock Stretch Enable bit (when operating as I<sup>2</sup>C slave)  
Used in conjunction with the SCLREL bit.  
1 = Enables software or receives clock stretching  
0 = Disables software or receives clock stretching
- bit 5      **ACKDT:** Acknowledge Data bit (when operating as I<sup>2</sup>C master, applicable during master receive)  
Value that is transmitted when the software initiates an Acknowledge sequence.  
1 = Sends NACK during Acknowledge  
0 = Sends ACK during Acknowledge
- bit 4      **ACKEN:** Acknowledge Sequence Enable bit  
(when operating as I<sup>2</sup>C master, applicable during master receive)  
1 = Initiates Acknowledge sequence on the SDA1 and SCL1 pins and transmits the ACKDT data bit.  
Hardware clears it at the end of the master Acknowledge sequence.  
0 = Acknowledge sequence is not in progress
- bit 3      **RCEN:** Receive Enable bit (when operating as I<sup>2</sup>C master)  
1 = Enables Receive mode for I<sup>2</sup>C. Hardware clears it at the end of the eighth bit of the master receive data byte.  
0 = Receive sequence is not in progress
- bit 2      **PEN:** Stop Condition Enable bit (when operating as I<sup>2</sup>C master)  
1 = Initiates Stop condition on the SDA1 and SCL1 pins. Hardware clears it at the end of the master Stop sequence.  
0 = Stop condition is not in progress
- bit 1      **RSEN:** Repeated Start Condition Enable bit (when operating as I<sup>2</sup>C master)  
1 = Initiates Repeated Start condition on the SDA1 and SCL1 pins. Hardware clears it at the end of the master Repeated Start sequence.  
0 = Repeated Start condition is not in progress
- bit 0      **SEN:** Start Condition Enable bit (when operating as I<sup>2</sup>C master)  
1 = Initiates Start condition on the SDA1 and SCL1 pins. Hardware clears it at the end of the master Start sequence.  
0 = Start condition is not in progress

# dsPIC33EPXXGS202 FAMILY

## REGISTER 17-4: I2C1MSK: I2C1 SLAVE MODE ADDRESS MASK REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	AMSK<9:8>	
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
AMSK<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 15-10

**Unimplemented:** Read as '0'

bit 9-0

**AMSK<9:0>:** Address Mask Select bits

For 10-Bit Address:

1 = Enables masking for bit Ax of incoming message address; bit match is not required in this position

0 = Disables masking for bit Ax; bit match is required in this position

For 7-Bit Address (I2C1MSK<6:0> only):

1 = Enables masking for bit Ax + 1 of incoming message address; bit match is not required in this position

0 = Disables masking for bit Ax + 1; bit match is required in this position

# dsPIC33EPXXGS202 FAMILY

**REGISTER 19-2: ADCON1H: ADC CONTROL REGISTER 1 HIGH**

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-1	R/W-1	U-0	U-0	U-0	U-0	U-0
FORM	SHRRES1	SHRRES0	—	—	—	—	—
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 15-8 **Unimplemented:** Read as '0'

bit 7 **FORM:** Fractional Data Output Format bit

1 = Fractional

0 = Integer

bit 6-5 **SHRRES<1:0>:** Shared ADC Core Resolution Selection bits

11 = 12-bit resolution

10 = 10-bit resolution

01 = 8-bit resolution

00 = 6-bit resolution

bit 4-0 **Unimplemented:** Read as '0'

# dsPIC33EPXXGS202 FAMILY

## REGISTER 19-20: ADTRIGxL: ADC CHANNEL TRIGGER x SELECTION REGISTER LOW (x = 0 to 3)

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	TRGSRC(4x+1)<4:0>				
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	TRGSRC(4x)<4:0>				
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 15-13 **Unimplemented:** Read as '0'

bit 12-8 **TRGSRC(4x+1)<4:0>**: Trigger Source Selection for Corresponding Analog Inputs bits

11111 = ADTRG31  
 11110 = Reserved  
 11101 = Reserved  
 11100 = Reserved  
 11011 = Reserved  
 11010 = PWM Generator 3 current-limit trigger  
 11001 = PWM Generator 2 current-limit trigger  
 11000 = PWM Generator 1 current-limit trigger  
 10111 = Reserved  
 10110 = Output Compare 1 trigger  
 10101 = Reserved  
 10100 = Reserved  
 10011 = Reserved  
 10010 = Reserved  
 10001 = PWM Generator 3 secondary trigger  
 10000 = PWM Generator 2 secondary trigger  
 01111 = PWM Generator 1 secondary trigger  
 01110 = PWM secondary Special Event Trigger  
 01101 = Timer2 period match  
 01100 = Timer1 period match  
 01011 = Reserved  
 01010 = Reserved  
 01001 = Reserved  
 01000 = Reserved  
 00111 = PWM Generator 3 primary trigger  
 00110 = PWM Generator 2 primary trigger  
 00101 = PWM Generator 1 primary trigger  
 00100 = PWM Special Event Trigger  
 00011 = Reserved  
 00010 = Level software trigger  
 00001 = Common software trigger  
 00000 = No trigger is enabled

bit 7-5 **Unimplemented:** Read as '0'

# dsPIC33EPXXGS202 FAMILY

## REGISTER 19-23: ADCAL1H: ADC CALIBRATION REGISTER 1 HIGH

R/W-0, HS	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
CSHRRDY	—	—	—	CSHRSKIP	CSHRDIFF	CSHREN	CSHRRUN
bit 15				bit 8			

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7				bit 0			

<b>Legend:</b>	HS = Hardware Settable bit		
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      **CSHRRDY:** Shared ADC Core Calibration Status Flag bit  
1 = Shared ADC core calibration is finished  
0 = Shared ADC core calibration is in progress
- bit 14-12    **Unimplemented:** Read as '0'
- bit 11      **CSHRSKIP:** Shared ADC Core Calibration Bypass bit  
1 = After power-up, the shared ADC core will not be calibrated  
0 = After power-up, the shared ADC core will be calibrated
- bit 10      **CSHRDIFF:** Shared ADC Core Pseudo-Differential Input Mode Calibration bit  
1 = Shared ADC core will be calibrated in Pseudo-Differential Input mode  
0 = Shared ADC core will be calibrated in Single-Ended Input mode
- bit 9        **CSHREN:** Shared ADC Core Calibration Enable bit  
1 = Shared ADC core calibration bits (CSHRRDY, CSHRSKIP, CSHRDIF and CSHRRUN) can be accessed by software  
0 = Shared ADC core calibration bits are disabled
- bit 8        **CSHRRUN:** Shared ADC Core Calibration Start bit  
1 = If this bit is set by software, the shared ADC core calibration cycle is started; this bit is cleared automatically by hardware  
0 = Software can start the next calibration cycle
- bit 7-0      **Unimplemented:** Read as '0'

# dsPIC33EPXXGS202 FAMILY

**TABLE 23-2: INSTRUCTION SET OVERVIEW (CONTINUED)**

Base Instr #	Assembly Mnemonic	Assembly Syntax	Description	# of Words	# of Cycles	Status Flags Affected
9	BSW	BSW.C Ws,Wb	Write C bit to Ws<Wb>	1	1	None
		BSW.Z Ws,Wb	Write Z bit to Ws<Wb>	1	1	None
10	BTG	BTG f,#bit4	Bit Toggle f	1	1	None
		BTG Ws,#bit4	Bit Toggle Ws	1	1	None
11	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
12	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
13	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
14	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
15	CALL	CALL lit23	Call subroutine	2	4	SFA
		CALL Wn	Call indirect subroutine	1	4	SFA
		CALL.L Wn	Call indirect subroutine (long address)	1	4	SFA
16	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
17	CLRWDT	CLRWDT	Clear Watchdog Timer	1	1	WDTO,SLEEP
18	COM	COM f	f = $\bar{f}$	1	1	N,Z
		COM f,WREG	WREG = $\bar{f}$	1	1	N,Z
		COM Ws,Wd	Wd = $\overline{Ws}$	1	1	N,Z
19	CP	CP f	Compare f with WREG	1	1	C,DC,N,OV,Z
		CP Wb,#lit8	Compare Wb with lit8	1	1	C,DC,N,OV,Z
		CP Wb,Ws	Compare Wb with Ws (Wb – Ws)	1	1	C,DC,N,OV,Z
20	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
21	CPB	CPB f	Compare f with WREG, with Borrow	1	1	C,DC,N,OV,Z
		CPB Wb,#lit8	Compare Wb with lit8, with Borrow	1	1	C,DC,N,OV,Z
		CPB Wb,Ws	Compare Wb with Ws, with Borrow (Wb – Ws – C)	1	1	C,DC,N,OV,Z
22	CPSEQ	CPSEQ Wb,Wn	Compare Wb with Wn, skip if =	1	1 (2 or 3)	None
	CPBEQ	CPBEQ Wb,Wn,Expr	Compare Wb with Wn, branch if =	1	1 (5)	None
23	CPSGT	CPSGT Wb,Wn	Compare Wb with Wn, skip if >	1	1 (2 or 3)	None
	CPBGT	CPBGT Wb,Wn,Expr	Compare Wb with Wn, branch if >	1	1 (5)	None
24	CPSLT	CPSLT Wb,Wn	Compare Wb with Wn, skip if <	1	1 (2 or 3)	None
	CPBLT	CPBLT Wb,Wn,Expr	Compare Wb with Wn, branch if <	1	1 (5)	None
25	CPSNE	CPSNE Wb,Wn	Compare Wb with Wn, skip if ≠	1	1 (2 or 3)	None
	CPBNE	CPBNE Wb,Wn,Expr	Compare Wb with Wn, branch if ≠	1	1 (5)	None

**Note:** Read and Read-Modify-Write (e.g., bit operations and logical operations) on non-CPU SFRs incur an additional instruction cycle.



# dsPIC33EPXXGS202 FAMILY

---

## 24.11 Demonstration/Development Boards, Evaluation Kits and Starter Kits

A wide variety of demonstration, development and evaluation boards for various PIC MCUs and dsPIC DSCs allows quick application development on fully functional systems. Most boards include prototyping areas for adding custom circuitry and provide application firmware and source code for examination and modification.

The boards support a variety of features, including LEDs, temperature sensors, switches, speakers, RS-232 interfaces, LCD displays, potentiometers and additional EEPROM memory.

The demonstration and development boards can be used in teaching environments, for prototyping custom circuits and for learning about various microcontroller applications.

In addition to the PICDEM™ and dsPICDEM™ demonstration/development board series of circuits, Microchip has a line of evaluation kits and demonstration software for analog filter design, KEELOQ® security ICs, CAN, IrDA®, PowerSmart battery management, SEEVAL® evaluation system, Sigma-Delta ADC, flow rate sensing, plus many more.

Also available are starter kits that contain everything needed to experience the specified device. This usually includes a single application and debug capability, all on one board.

Check the Microchip web page ([www.microchip.com](http://www.microchip.com)) for the complete list of demonstration, development and evaluation kits.

## 24.12 Third-Party Development Tools

Microchip also offers a great collection of tools from third-party vendors. These tools are carefully selected to offer good value and unique functionality.

- Device Programmers and Gang Programmers from companies, such as SoftLog and CCS
- Software Tools from companies, such as Gimpel and Trace Systems
- Protocol Analyzers from companies, such as Saleae and Total Phase
- Demonstration Boards from companies, such as MikroElektronika, Digilent® and Olimex
- Embedded Ethernet Solutions from companies, such as EZ Web Lynx, WIZnet and IPLogika®

# dsPIC33EPXXGS202 FAMILY

**TABLE 25-6: DC CHARACTERISTICS: OPERATING CURRENT (IDD)**

DC 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			
Parameter No.	Typ.	Max.	Units	Conditions		
Operating Current (IDD) <sup>(1)</sup>						
DC20d	5	10	mA	-40°C	3.3V	10 MIPS
DC20a	5	10	mA	+25°C		
DC20b	5	10	mA	+85°C		
DC20c	5	10	mA	+125°C		
DC22d	10	15	mA	-40°C	3.3V	20 MIPS
DC22a	10	15	mA	+25°C		
DC22b	10	15	mA	+85°C		
DC22c	10	15	mA	+125°C		
DC24d	15	20	mA	-40°C	3.3V	40 MIPS
DC24a	15	20	mA	+25°C		
DC24b	15	20	mA	+85°C		
DC24c	15	20	mA	+125°C		
DC25d	20	28	mA	-40°C	3.3V	60 MIPS
DC25a	20	28	mA	+25°C		
DC25b	20	28	mA	+85°C		
DC25c	20	28	mA	+125°C		
DC26d	30	35	mA	-40°C	3.3V	70 MIPS
DC26a	30	35	mA	+25°C		
DC26b	30	35	mA	+85°C		

**Note 1:** IDD is primarily a function of the operating voltage and frequency. Other factors, such as I/O pin loading and switching rate, oscillator type, internal code execution pattern and temperature, also have an impact on the current consumption. The test conditions for all IDD measurements are as follows:

- Oscillator is configured in EC mode with PLL, OSC1 is driven with external square wave from rail-to-rail (EC Clock Overshoot/Undershoot < 250 mV required)
- CLK0 is configured as an I/O input pin in the Configuration Word
- All I/O pins are configured as outputs and driving low
- MCLR = VDD, WDT and FSCM are disabled
- CPU, SRAM, program memory and data memory are operational
- No peripheral modules are operating or being clocked (defined PMDx bits are all ones)
- CPU executing:

```
while(1)
{
    NOP();
}
```
- JTAG is disabled

# dsPIC33EPXXGS202 FAMILY

**TABLE 25-10: DC CHARACTERISTICS: DOZE CURRENT (IDOZE)**

DC 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				
Parameter No.	Typ.	Max.	Doze Ratio	Units	Conditions		
Doze Current (IDoZE) <sup>(1)</sup>							
DC73a <sup>(2)</sup>	15	20	1:2	mA	-40°C	3.3V	Fosc = 140 MHz
DC73g	7	9	1:128	mA			
DC70a <sup>(2)</sup>	15	20	1:2	mA	+25°C	3.3V	Fosc = 140 MHz
DC70g	7	9	1:128	mA			
DC71a <sup>(2)</sup>	15	20	1:2	mA	+85°C	3.3V	Fosc = 140 MHz
DC71g	7	9	1:128	mA			
DC72a <sup>(2)</sup>	15	20	1:2	mA	+125°C	3.3V	Fosc = 120 MHz
DC72g	7	9	1:128	mA			

**Note 1:** IDOZE is primarily a function of the operating voltage and frequency. Other factors, such as I/O pin loading and switching rate, oscillator type, internal code execution pattern and temperature, also have an impact on the current consumption. The test conditions for all IDOZE measurements are as follows:

- Oscillator is configured in EC mode and external clock is active, OSC1 is driven with external square wave from rail-to-rail (EC Clock Overshoot/Undershoot < 250 mV required)
- CLK0 is configured as an I/O input pin in the Configuration Word
- All I/O pins are configured as outputs and driving low
- MCLR = VDD, WDT and FSCM are disabled
- CPU, SRAM, program memory and data memory are operational
- No peripheral modules are operating or being clocked (defined PMDx bits are all ones)
- CPU executing:

```
while(1)
{
    NOP();
}
```
- JTAG is disabled

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

# dsPIC33EPXXGS202 FAMILY

---

NOTES:

# dsPIC33EPXXGS202 FAMILY

Instruction Addressing Modes.....	53
File Register Instructions .....	53
Fundamental Modes Supported.....	53
MAC Instructions.....	54
MCU Instructions .....	53
Move and Accumulator Instructions.....	54
Other Instructions.....	54
Instruction Set .....	
Overview .....	254
Summary.....	251
Symbols Used in Opcode Descriptions.....	252
Instruction-Based Power-Saving Modes .....	99
Idle .....	100
Sleep.....	100
Inter-Integrated Circuit (I <sup>2</sup> C).....	185
Control Registers .....	187
Resources .....	185
Inter-Integrated Circuit. <i>See</i> I <sup>2</sup> C.	
Internet Address.....	338
Interrupt Controller .....	
Alternate Interrupt Vector Table (AIVT) .....	73
Control and Status Registers .....	78
INTCON1 .....	78
INTCON2 .....	78
INTCON3 .....	78
INTCON4 .....	78
INTTREG .....	78
Interrupt Vector Details .....	76
Interrupt Vector Table (IVT) .....	73
Reset Sequence .....	73
Resources .....	78
Interrupts Coincident with Power Save Instructions.....	100
<b>J</b>	
JTAG Boundary Scan Interface .....	239
JTAG Interface.....	248
<b>L</b>	
Leading-Edge Blanking (LEB).....	151
LPRC Oscillator .....	
Use with WDT .....	247
<b>M</b>	
Memory Organization.....	27
Resources .....	33
Microchip Internet Web Site .....	338
Modulo Addressing .....	55
Applicability .....	56
Operation Example .....	55
Start and End Address.....	55
W Address Register Selection .....	55
MPLAB Assembler, Linker, Librarian .....	262
MPLAB ICD 3 In-Circuit Debugger .....	263
MPLAB PM3 Device Programmer .....	263
MPLAB REAL ICE In-Circuit Emulator System.....	263
MPLAB X Integrated Development .....	
Environment Software.....	261
MPLINK Object Linker/MPLIB Object Librarian .....	262
<b>O</b>	
Oscillator .....	
Control Registers .....	91
Resources .....	90
OTP Memory Area .....	246
Output Compare .....	145
Control Registers .....	146
Resources .....	145

## P

Packaging .....	315
Details.....	316
Marking.....	315
Peripheral Module Disable (PMD) .....	101
Peripheral Pin Select (PPS).....	109
Available Peripherals .....	109
Available Pins .....	109
Control .....	109
Control Registers.....	115
Input Mapping .....	110
Output Mapping .....	112
Output Selection for Remappable Pins .....	112
Selectable Input Sources.....	111
Peripheral Pin Select. <i>See</i> PPS.	
PICKIT 3 In-Circuit Debugger/Programmer .....	263
Pinout I/O Descriptions (table).....	8
Power-Saving Features .....	99
Clock Frequency and Switching .....	99
Control Registers.....	102
Resources .....	101
Program Address Space.....	27
Construction .....	58
Data Access from Program Memory Using .....	
Table Instructions .....	59
Memory Map (dsPIC33EP16GS202 Devices).....	28
Memory Map (dsPIC33EP32GS202 Devices).....	29
Table Read High Instructions (TBLRDH) .....	59
Table Read Low Instructions (TBLRDL).....	59
Program Memory .....	
Organization .....	30
Reset Vector.....	30
Programmable Gain Amplifier (PGA).....	235
Control Registers.....	237
Description.....	236
Resources .....	237
Programmable Gain Amplifier. <i>See</i> PGA.	
Programmer's Model .....	19
Register Descriptions .....	19
Pulse-Width Modulation. <i>See</i> PWM.	

## R

### Register Maps

ADC .....	43
Analog Comparator .....	47
CPU Core .....	34
I2C1 .....	42
Input Capture 1 .....	38
Interrupt Controller.....	36
NVM.....	46
Output Compare 1 .....	38
Peripheral Pin Select Output .....	45
PMD.....	46
PORTA .....	48
PORTB .....	48
Programmable Gain Amplifier .....	47
PWM .....	39
PWM Generator 1.....	39
PWM Generator 2.....	40
PWM Generator 3.....	41
SPI1 .....	42
System Control.....	46
Timer1 through Timer3 .....	38
UART1 .....	42