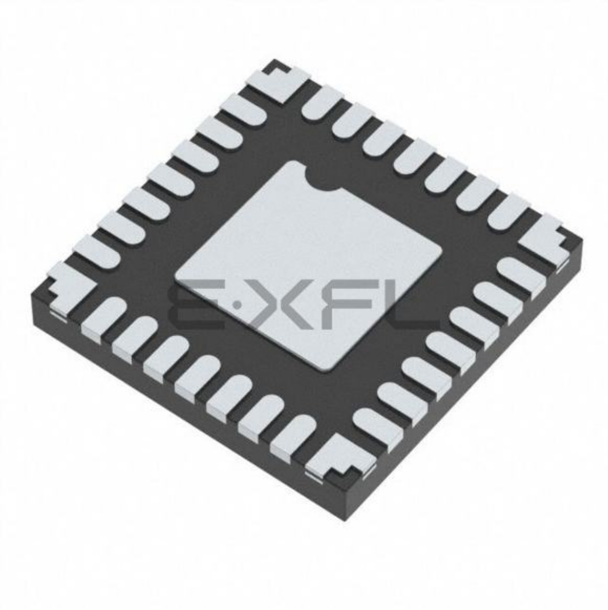
Microchip Technology - DSPIC33EP16GS202T-I/M6 Datasheet



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.

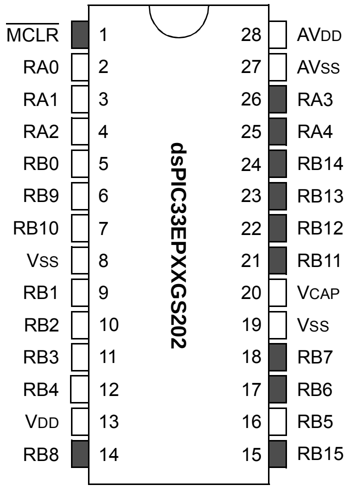
Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	dsPIC
Core Size	16-Bit
Speed	70 MIPs
Connectivity	I²C, IrDA, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	21
Program Memory Size	16KB (16K 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 ~ 85°C
Mounting Type	Surface Mount
Package / Case	28-UFQFN Exposed Pad
Supplier Device Package	28-UQFN (4x4)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/dspic33ep16gs202t-i-m6

Pin Diagrams

28-Pin SOIC,
28-Pin SSOP

■ = Pins are up to 5V tolerant



PIN FUNCTION DESCRIPTIONS

Pin	Pin Function	Pin	Pin Function
1	$\overline{\text{MCLR}}$	15	PGEC3/**RP47**/RB15
2	AN0/PGA1P1/CMP1A/RA0	16	TDO/AN9/PGA2N2/**RP37**/RB5
3	AN1/PGA1P2/PGA2P1/CMP1B/RA1	17	PGED1/TDI/AN10/SCL1/**RP38**/RB6
4	AN2/PGA1P3/PGA2P2/CMP1C/CMP2A/RA2	18	PGEC1/AN11/SDA1/**RP39**/RB7
5	AN3/PGA2P3/CMP1D/CMP2B/**RP32**/RB0	19	VSS
6	AN4/CMP2C/**RP41**/RB9	20	VCAP
7	AN5/CMP2D/**RP42**/RB10	21	TMS/PWM3H/**RP43**/RB11
8	VSS	22	TCK/PWM3L/**RP44**/RB12
9	OSC1/CLKI/AN6/**RP33**/RB1	23	PWM2H/**RP45**/RB13
10	OSC2/CLKO/AN7/PGA1N2/**RP34**/RB2	24	PWM2L/**RP46**/RB14
11	PGED2/AN8/INT0/**RP35**/RB3	25	PWM1H/RA4
12	PGEC2/ADTRG31/**RP36**/RB4	26	PWM1L/RA3
13	VDD	27	AVSS
14	PGED3/**RP40**/RB8	28	AVDD

Legend: Shaded pins are up to 5 VDC tolerant.
Note: **RPn** represents remappable peripheral functions. See Table 10-1 and Table 10-2 for the complete list of remappable sources.

TABLE 4-14: ADC REGISTER MAP

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
ADCON1L	0300	ADON	—	ADSIDL	—	—	—	—	—	—	—	—	—	—	—	—	—	1000
ADCON1H	0302	—	—	—	—	—	—	—	—	FORM	SHRRES1	SHRRES0	—	—	—	—	—	0060
ADCON2L	0304	REFCIE	REFERCIE	—	EIEN	—	SHREISEL2	SHREISEL1	SHREISEL0	—	SHRADCS6	SHRADCS5	SHRADCS4	SHRADCS3	SHRADCS2	SHRADCS1	SHRADCS0	0000
ADCON2H	0306	REFRDY	REFERR	—	—	—	—	SHRSAMC9	SHRSAMC8	SHRSAMC7	SHRSAMC6	SHRSAMC5	SHRSAMC4	SHRSAMC3	SHRSAMC2	SHRSAMC1	SHRSAMC0	0000
ADCON3L	0308	REFSEL2	REFSEL1	REFSEL0	SUSPEND	SUSPCIE	SUSPRDY	SHRSAMP	CNVRTCH	SWLCTRG	SWCTRG	CNVCHSEL5	CNVCHSEL4	CNVCHSEL3	CNVCHSEL2	CNVCHSEL1	CNVCHSEL0	0000
ADCON3H	030A	CLKSEL1	CLKSEL0	CLKDIV5	CLKDIV4	CLKDIV3	CLKDIV2	CLKDIV1	CLKDIV0	SHREN	—	—	—	—	—	C1EN	C0EN	0000
ADCON4L	030C	—	—	—	—	—	—	SYNCTRG1	SYNCTRG0	—	—	—	—	—	—	SAMC1EN	SAMC0EN	0000
ADCON4H	030E	—	—	—	—	—	—	—	—	—	—	—	—	C1CHS1	C1CHS0	C0CHS1	C0CHS0	0000
ADMOD0L	0310	—	SIGN7	—	SIGN6	—	SIGN5	—	SIGN4	—	SIGN3	—	SIGN2	DIFF1	SIGN1	DIFF0	SIGN0	0000
ADMOD0H	0312	—	—	DIFF14	SIGN14	—	SIGN13	—	SIGN12	—	SIGN11	—	SIGN10	—	SIGN9	—	SIGN8	0000
ADIEL	0320	—	IE14	—	—	IE<11:0												0000
ADSTATL	0330	—	AN14RDY	—	—	AN11RDY	AN10RDY	AN9RDY	AN8RDY	AN7RDY	AN6RDY	AN5RDY	AN4RDY	AN3RDY	AN2RDY	AN1RDY	AN0RDY	0000
ADCMP0ENL	0338	—	CMPEN14	—	—	CMPEN<11:0>												0000
ADCMP0LO	033C	ADC CMPLO Register																0000
ADCMP0HI	033E	ADC CMPHI Register																0000
ADCMP1ENL	0340	—	CMPEN14	—	—	CMPEN<11:0>												0000
ADCMP1LO	0344	ADC CMPLO Register																0000
ADCMP1HI	0346	ADC CMPHI Register																0000
ADFL0DAT	0368	ADC FLDATA Register																0000
ADFL0CON	036A	FLEN	MODE1	MODE0	OVRSAM2	OVRSAM1	OVRSAM0	IE	RDY	—	—	—	FLCHSEL4	FLCHSEL3	FLCHSEL2	FLCHSEL1	FLCHSEL0	0000
ADTRIG0L	0380	—	—	—	TRGSRC1<4:0>					—	—	—	TRGSRC0<4:0>					0000
ADTRIG0H	0382	—	—	—	TRGSRC3<4:0>					—	—	—	TRGSRC2<4:0>					0000
ADTRIG1L	0384	—	—	—	TRGSRC5<4:0>					—	—	—	TRGSRC4<4:0>					0000
ADTRIG1H	0386	—	—	—	TRGSRC7<4:0>					—	—	—	TRGSRC6<4:0>					0000
ADTRIG2L	0388	—	—	—	TRGSRC9<4:0>					—	—	—	TRGSRC8<4:0>					0000
ADTRIG2H	038A	—	—	—	TRGSRC11<4:0>					—	—	—	TRGSRC10<4:0>					0000
ADTRIG3L	038C	—	—	—	TRGSRC13<4:0>					—	—	—	TRGSRC12<4:0>					0000
ADTRIG3H	038E	—	—	—	—	—	—	—	—	—	—	—	TRGSRC14<4:0>					0000
ADCMP0CON	03A0	—	—	—	CHNL4	CHNL3	CHNL2	CHNL1	CHNL0	CMPEN	IE	STAT	BTWN	HIHI	HILO	LOHI	LOLO	0000
ADCMP1CON	03A4	—	—	—	CHNL4	CHNL3	CHNL2	CHNL1	CHNL0	CMPEN	IE	STAT	BTWN	HIHI	HILO	LOHI	LOLO	0000
ADLVLTRGL	03D0	—	LVLEN14	—	—	LVLEN<11:0>												0000
ADCORE0L	03D4	—	—	—	—	—	—	SAMC<9:0>										0000
ADCORE0H	03D6	—	—	—	EISEL2	EISEL1	EISEL0	RES1	RES0	—	ADCS6	ADCS5	ADCS4	ADCS3	ADCS2	ADCS1	ADCS0	0000
ADCORE1L	03D8	—	—	—	—	—	—	SAMC<9:0>										0000
ADCORE1H	03DA	—	—	—	EISEL2	EISEL1	EISEL0	RES1	RES0	—	ADCS6	ADCS5	ADCS4	ADCS3	ADCS2	ADCS1	ADCS0	0000
ADEIEL	03F0	—	EIEN14	—	—	EIEN<11:0												0000
ADEISTATL	03F8	—	EISTAT14	—	—	EISTAT<11:0>												0000

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

7.0 INTERRUPT CONTROLLER

> **Note 1:** This data sheet summarizes the features of the dsPIC33EPXXGS202 family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **"Interrupts"** (DS70000600) in the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).
>
> **2:** Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

The dsPIC33EPXXGS202 family interrupt controller reduces the numerous peripheral interrupt request signals to a single interrupt request signal to the dsPIC33EPXXGS202 family CPU.

The interrupt controller has the following features:

- Six Processor Exceptions and Software Traps
- Seven User-Selectable Priority Levels
- Interrupt Vector Table (IVT) with a Unique Vector for each Interrupt or Exception Source
- Fixed Priority within a Specified User Priority Level
- Fixed Interrupt Entry and Return Latencies
- Alternate Interrupt Vector Table (AIVT) for Debug Support

7.1 Interrupt Vector Table

The dsPIC33EPXXGS202 family Interrupt Vector Table (IVT), shown in Figure 7-1, resides in program memory, starting at location, 000004h. The IVT contains six non-maskable trap vectors and up to fifty sources of interrupts. In general, each interrupt source has its own vector. Each interrupt vector contains a 24-bit wide address. The value programmed into each interrupt vector location is the starting address of the associated Interrupt Service Routine (ISR).

Interrupt vectors are prioritized in terms of their natural priority. This priority is linked to their position in the vector table. Lower addresses generally have a higher natural priority. For example, the interrupt associated with Vector 0 takes priority over interrupts at any other vector address.

7.1.1 ALTERNATE INTERRUPT VECTOR TABLE

The Alternate Interrupt Vector Table (AIVT), shown in Figure 7-2, is available only when the Boot Segment (BS) is defined and the AIVT has been enabled. To enable the Alternate Interrupt Vector Table, the Configuration bit, AIVTDIS in the FSEC register, must be programmed and the AIVTEN bit must be set (INTCON2<8> = 1). When the AIVT is enabled, all interrupt and exception processes use the alternate vectors instead of the default vectors. The AIVT begins at the start of the last page of the Boot Segment, defined by BSLIM<12:0>. The second half of the page is no longer usable space. The Boot Segment must be at least 2 pages to enable the AIVT.

> **Note:** Although the Boot Segment must be enabled in order to enable the AIVT, application code does not need to be present inside of the Boot Segment. The AIVT (and IVT) will inherit the Boot Segment code protection.

The AIVT supports debugging by providing a means to switch between an application and a support environment without requiring the interrupt vectors to be reprogrammed. This feature also enables switching between applications for evaluation of different software algorithms at run time.

7.2 Reset Sequence

A device Reset is not a true exception because the interrupt controller is not involved in the Reset process. The dsPIC33EPXXGS202 family devices clear their registers in response to a Reset, which forces the PC to zero. The device then begins program execution at location, 0x000000. A GOTO instruction at the Reset address can redirect program execution to the appropriate start-up routine.

> **Note:** Any unimplemented or unused vector locations in the IVT should be programmed with the address of a default interrupt handler routine that contains a RESET instruction.

8.1 CPU Clocking System

The dsPIC33EPXXGS202 family of devices provides six system clock options:

- Fast RC (FRC) Oscillator
- FRC Oscillator with Phase-Locked Loop (PLL)
- FRC Oscillator with Postscaler
- Primary (XT, HS or EC) Oscillator
- Primary Oscillator with PLL
- Low-Power RC (LPRC) Oscillator

Instruction execution speed or device operating frequency, FCY, is given by Equation 8-1.

EQUATION 8-1: DEVICE OPERATING FREQUENCY

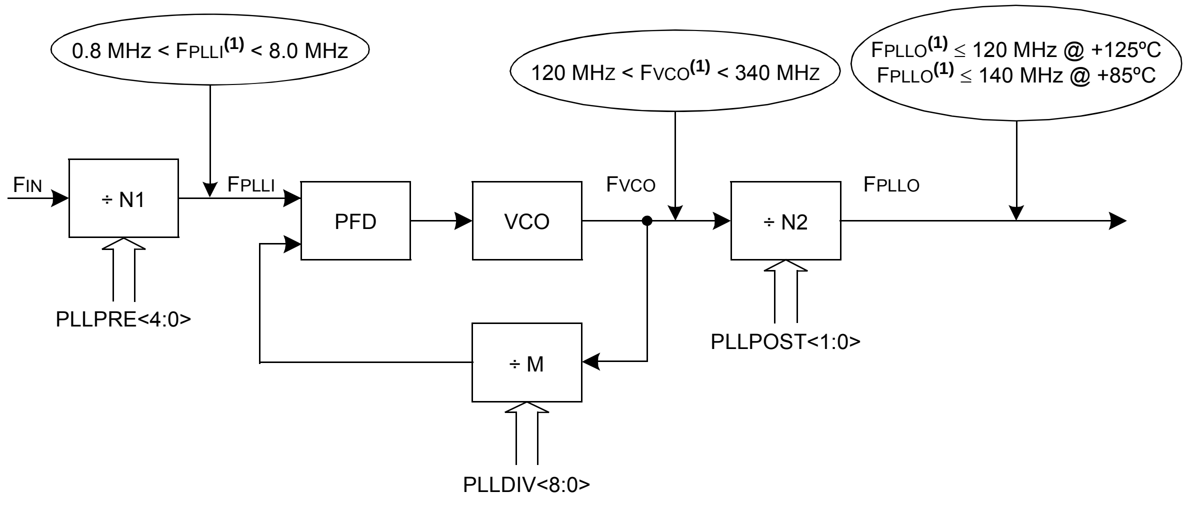
$$F_{CY} = F_{OSC}/2$$

Figure 8-2 is a block diagram of the PLL module.

Equation 8-2 provides the relationship between input frequency (FIN) and output frequency (FPLLO).

Equation 8-3 provides the relationship between input frequency (FIN) and VCO frequency (FVCO).

FIGURE 8-2: PLL BLOCK DIAGRAM


0.8 MHz < FPLLI(1) < 8.0 MHz
120 MHz < FVCO(1) < 340 MHz
FPLLO(1) ≤ 120 MHz @ +125ºC
FPLLO(1) ≤ 140 MHz @ +85ºC
FIN
÷ N1
FPLLI
PFD
VCO
FVCO
÷ N2
FPLLO
PLLPRE<4:0>
÷ M
PLLPOST<1:0>
PLLDIV<8:0>

Note 1: This frequency range must be met at all times.

EQUATION 8-2: FPLLO CALCULATION

$$F_{PLLO} = F_{IN} \times \left(\frac{M}{N1 \times N2}\right) = F_{IN} \times \left(\frac{(PLLDIV\text{<8:0>} + 2)}{(PLLPRE\text{<4:0>} + 2) \times 2(PLLPOST\text{<1:0>} + 1)}\right)$$

Where:

$N1 = PLLPRE\text{<4:0>} + 2$

$N2 = 2 \text{ x } (PLLPOST\text{<1:0>} + 1)$

$M = PLLDIV\text{<8:0>} + 2$

EQUATION 8-3: FVCO CALCULATION

$$F_{VCO} = F_{IN} \times \left(\frac{M}{N1}\right) = F_{IN} \times \left(\frac{(PLLDIV\text{<8:0>} + 2)}{(PLLPRE\text{<4:0>} + 2)}\right)$$

REGISTER 9-2: PMD2: PERIPHERAL MODULE DISABLE CONTROL REGISTER 2

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	IC1MD
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	OC1MD
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 15-9 **Unimplemented:** Read as ‘0’

bit 8 **IC1MD:** Input Capture 1 Module Disable bit
1 = Input Capture 1 module is disabled
0 = Input Capture 1 module is enabled

bit 7-1 **Unimplemented:** Read as ‘0’

bit 0 **OC1MD:** Output Compare 1 Module Disable bit
1 = Output Compare 1 module is disabled
0 = Output Compare 1 module is enabled

REGISTER 9-3: PMD3: PERIPHERAL MODULE DISABLE CONTROL REGISTER 3

U-0	U-0	U-0	U-0	U-0	R/W-0	U-0	U-0
—	—	—	—	—	CMPMD	—	—
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 **CMPMD:** Comparator Module Disable bit
1 = Comparator module is disabled
0 = Comparator module is enabled

bit 9-0 **Unimplemented:** Read as ‘0’

NOTES:

REGISTER 15-1: PTCON: PWM TIME BASE CONTROL REGISTER

R/W-0	U-0	R/W-0	HS/HC-0	R/W-0	R/W-0	R/W-0	R/W-0
PTEN	—	PTSIDL	SESTAT	SEIEN	EIPU[1]	SYNCPOL[1]	SYNCOEN[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
SYNCEN[1]	SYNCSRC2[1]	SYNCSRC1[1]	SYNCSRC0[1]	SEVTPS3[1]	SEVTPS2[1]	SEVTPS1[1]	SEVTPS0[1]
bit 7							bit 0

Legend:	HC = Hardware Clearable bit	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 **PTEN:** PWM Module Enable bit
1 = PWM module is enabled
0 = PWM module is disabled

bit 14 **Unimplemented:** Read as ‘0’

bit 13 **PTSIDL:** PWM Time Base Stop in Idle Mode bit
1 = PWM time base halts in CPU Idle mode
0 = PWM time base runs in CPU Idle mode

bit 12 **SESTAT:** Special Event Interrupt Status bit
1 = Special event interrupt is pending
0 = Special event interrupt is not pending

bit 11 **SEIEN:** Special Event Interrupt Enable bit
1 = Special event interrupt is enabled
0 = Special event interrupt is disabled

bit 10 **EIPU:** Enable Immediate Period Updates bit[1]
1 = Active Period register is updated immediately
0 = Active Period register updates occur on PWM cycle boundaries

bit 9 **SYNCPOL:** Synchronize Input and Output Polarity bit[1]
1 = SYNCIx/SYNCO1 polarity is inverted (active-low)
0 = SYNCIx/SYNCO1 is active-high

bit 8 **SYNCOEN:** Primary Time Base Synchronization Enable bit[1]
1 = SYNCO1 output is enabled
0 = SYNCO1 output is disabled

bit 7 **SYNCEN:** External Time Base Synchronization Enable bit[1]
1 = External synchronization of primary time base is enabled
0 = External synchronization of primary time base is disabled

bit 6-4 **SYNCSRC<2:0>:** Synchronous Source Selection bits[1]
111 = Reserved
101 = Reserved
100 = Reserved
011 = Reserved
010 = Reserved
001 = SYNCI2
000 = SYNCI1

Note 1: These bits should be changed only when PTEN = 0. In addition, when using the SYNCIx feature, the user application must program the Period register with a value that is slightly larger than the expected period of the external synchronization input signal.

REGISTER 15-13: PDCx: PWMx GENERATOR DUTY CYCLE REGISTER(1,2,3)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PDCx<15:8>							
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
PDCx<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 15-0 **PDCx<15:0>:** PWMx Generator Duty Cycle Value bits

Note 1: In Independent PWM mode, the PDCx register controls the PWMxH duty cycle only. In the Complementary, Redundant and Push-Pull PWM modes, the PDCx register controls the duty cycle of both the PWMxH and PWMxL.

2: The smallest pulse width that can be generated on the PWMx output corresponds to a value of 0x0008, while the maximum pulse width generated corresponds to a value of Period – 0x0008.

3: As the duty cycle gets closer to 0% or 100% of the PWM period (0 to 40 ns, depending on the mode of operation), PWM duty cycle resolution will increase from 1 to 3 LSBs.

REGISTER 15-14: SDCx: PWMx SECONDARY DUTY CYCLE REGISTER(1,2,3)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SDCx<15:8>							
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
SDCx<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 15-0 **SDCx<15:0>:** Secondary Duty Cycle for PWMxL Output Pin bits

Note 1: The SDCx register is used in Independent PWM mode only. When used in Independent PWM mode, the SDCx register controls the PWMxL duty cycle.

2: The smallest pulse width that can be generated on the PWM output corresponds to a value of 0x0008, while the maximum pulse width generated corresponds to a value of Period – 0x0008.

3: As the duty cycle gets closer to 0% or 100% of the PWM period (0 to 40 ns, depending on the mode of operation), PWM duty cycle resolution will increase from 1 to 3 LSBs.

REGISTER 15-24: LEBCONx: PWMx LEADING-EDGE BLANKING (LEB) CONTROL REGISTER (CONTINUED)

bit 1 **BPLH:** Blanking in PWMxL High Enable bit
1 = State blanking (of current-limit and/or Fault input signals) when the PWMxL output is high
0 = No blanking when the PWMxL output is high

bit 0 **BPLL:** Blanking in PWMxL Low Enable bit
1 = State blanking (of current-limit and/or Fault input signals) when the PWMxL output is low
0 = No blanking when the PWMxL output is low

Note 1: The blanking signal is selected via the BLANKSEL<3:0> bits in the AUXCONx register.

REGISTER 15-25: LEBDLYx: PWMx LEADING-EDGE BLANKING DELAY REGISTER

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	LEB<8:5>			
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
LEB<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

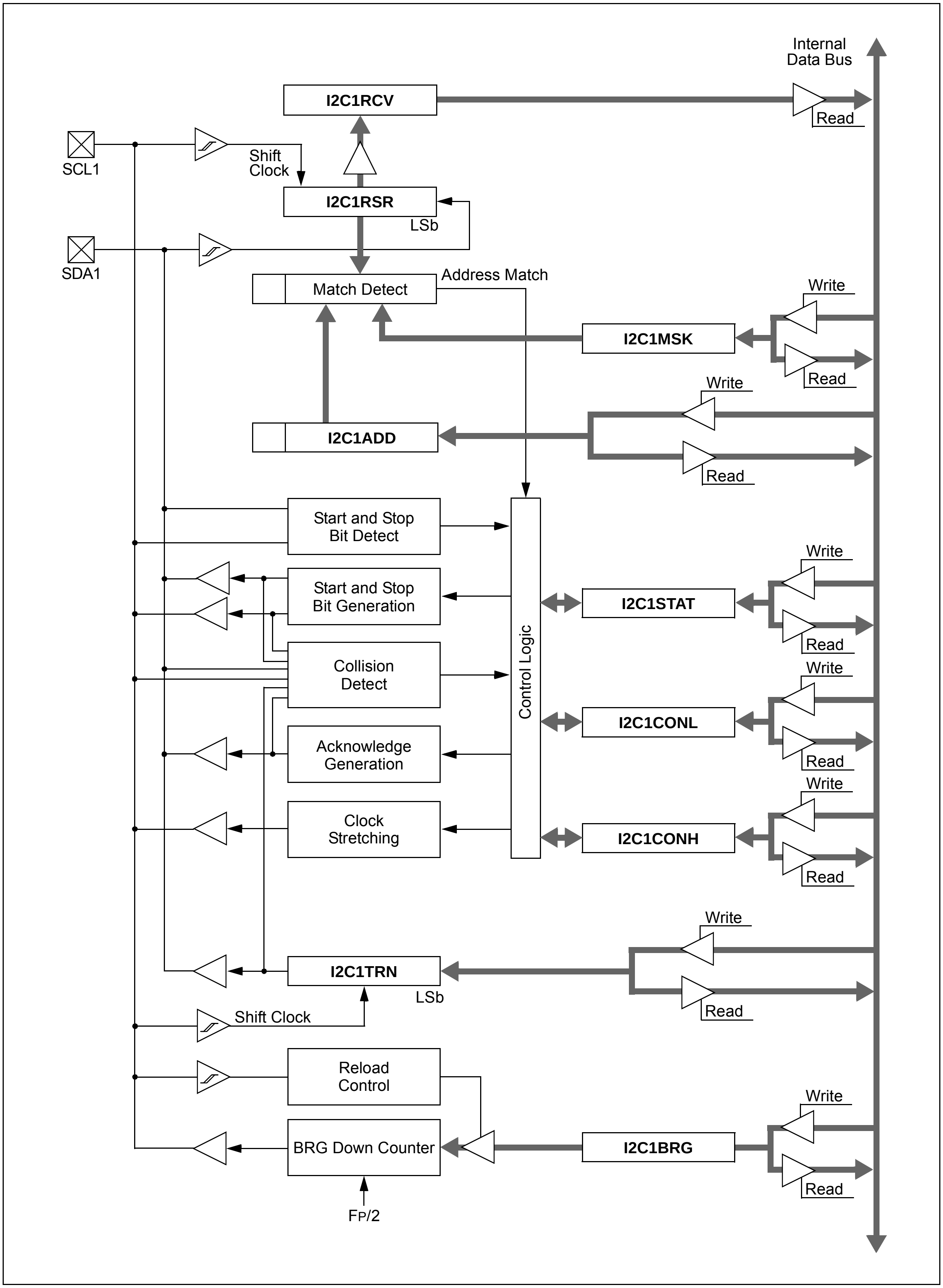
bit 15-12 **Unimplemented:** Read as ‘0’

bit 11-3 **LEB<8:0>:** Leading-Edge Blanking Delay for Current-Limit and Fault Inputs bits
The value is in 8.32 ns increments.

bit 2-0 **Unimplemented:** Read as ‘0’

NOTES:

FIGURE 17-1: I2C1 BLOCK DIAGRAM



REGISTER 17-2: I2C1CONH: I2C1 CONTROL REGISTER HIGH

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

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN
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 15-7 **Unimplemented:** Read as ‘0’

bit 6 **PCIE:** Stop Condition Interrupt Enable bit (I^2C Slave mode only)
1 = Enables interrupt on detection of Stop condition
0 = Stop detection interrupts are disabled

bit 5 **SCIE:** Start Condition Interrupt Enable bit (I^2C Slave mode only)
1 = Enables interrupt on detection of Start or Restart conditions
0 = Start detection interrupts are disabled

bit 4 **BOEN:** Buffer Overwrite Enable bit (I^2C Slave mode only)
1 = I2C1RCV is updated and an ACK is generated for a received address/data byte, ignoring the state of the I2COV bit only if the RBF bit = 0
0 = I2C1RCV is only updated when I2COV is clear

bit 3 **SDAHT:** SDA1 Hold Time Selection bit
1 = Minimum of 300 ns hold time on SDA1 after the falling edge of SCL1
0 = Minimum of 100 ns hold time on SDA1 after the falling edge of SCL1

bit 2 **SBCDE:** Slave Mode Bus Collision Detect Enable bit (I^2C Slave mode only)
1 = Enables slave bus collision interrupts
0 = Slave bus collision interrupts are disabled
If the rising edge of SCL1 and SDA1 is sampled low when the module is in a high state, the BCL bit is set and the bus goes Idle. This Detection mode is only valid during data and ACK transmit sequences.

bit 1 **AHEN:** Address Hold Enable bit (I^2C Slave mode only)
1 = Following the 8th falling edge of SCL1 for a matching received address byte, the SCLREL (I2C1CONL<12>) bit will be cleared and SCL1 will be held low
0 = Address holding is disabled

bit 0 **DHEN:** Data Hold Enable bit (I^2C Slave mode only)
1 = Following the 8th falling edge of SCL1 for a received data byte, the slave hardware clears the SCLREL (I2C1CONL<12>) bit and SCL1 is held low
0 = Data holding is disabled

REGISTER 19-7: ADCON4L: ADC CONTROL REGISTER 4 LOW

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	SYNCTRG1[1]	SYNCTRG0[1]
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	SAMC1EN	SAMC0EN
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 9-8 **SYNCTRG<1:0>** Dedicated ADC Core x Trigger Synchronization bits[1]

1 = All triggers are synchronized with the Core Source Clock (TCORESRC)
0 = The ADC core triggers are not synchronized

bit 7-2 **Unimplemented:** Read as ‘0’

bit 1-0 **SAMC1EN:SAMC0EN:** Dedicated ADC Core x Conversion Delay Enable bits

1 = After trigger, the conversion will be delayed and the ADC core will continue sampling during the time specified by the SAMC<9:0> bits in the ADCORExL register
0 = After trigger, the sampling will be stopped immediately and the conversion will be started on the next core clock cycle.

Note 1: For proper ADC performance, this bit must be set when using level-sensitive triggers and cleared for edge-sensitive triggers.

REGISTER 19-21: ADTRIGxH: ADC CHANNEL TRIGGER x SELECTION REGISTER HIGH (x = 0 to 3) (CONTINUED)

bit 4-0 **TRGSRC(4x+2)<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

REGISTER 21-1: PGAxCON: PGAx CONTROL REGISTER (x = 1,2) (CONTINUED)

bit 2-0 **GAIN<2:0>:** PGAx Gain Selection bits

111 = Reserved
110 = Gain of 64
101 = Gain of 32
100 = Gain of 16
011 = Gain of 8
010 = Gain of 4
001 = Reserved
000 = Reserved

REGISTER 21-2: PGAxCAL: PGAx CALIBRATION REGISTER (x = 1,2)

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

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	PGACAL<5:0>					
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 15-6 **Unimplemented:** Read as ‘0’

bit 5-0 **PGACAL<5:0>:** PGAx Offset Calibration bits

The calibration values for PGA1 and PGA2 must be copied from Flash addresses, 0x800E48 and 0x800E4C, respectively, into these bits before the module is enabled. Refer to the Device Calibration Addresses table (Table 22-3) in **Section 22.0 “Special Features”** for more information.

TABLE 23-2: INSTRUCTION SET OVERVIEW

Base Instr #	Assembly Mnemonic	Assembly Syntax		Description	# of Words	# of Cycles	Status Flags Affected
1	ADD	ADD	Acc	Add Accumulators	1	1	OA,OB,SA,SB
		ADD	f	f = f + WREG	1	1	C,DC,N,OV,Z
		ADD	f,WREG	WREG = f + WREG	1	1	C,DC,N,OV,Z
		ADD	#lit10,Wn	Wd = lit10 + Wd	1	1	C,DC,N,OV,Z
		ADD	Wb,Ws,Wd	Wd = Wb + Ws	1	1	C,DC,N,OV,Z
		ADD	Wb,#lit5,Wd	Wd = Wb + lit5	1	1	C,DC,N,OV,Z
		ADD	Wso,#Slit4,Acc	16-bit Signed Add to Accumulator	1	1	OA,OB,SA,SB
2	ADDC	ADDC	f	f = f + WREG + (C)	1	1	C,DC,N,OV,Z
		ADDC	f,WREG	WREG = f + WREG + (C)	1	1	C,DC,N,OV,Z
		ADDC	#lit10,Wn	Wd = lit10 + Wd + (C)	1	1	C,DC,N,OV,Z
		ADDC	Wb,Ws,Wd	Wd = Wb + Ws + (C)	1	1	C,DC,N,OV,Z
		ADDC	Wb,#lit5,Wd	Wd = Wb + lit5 + (C)	1	1	C,DC,N,OV,Z
3	AND	AND	f	f = f .AND. WREG	1	1	N,Z
		AND	f,WREG	WREG = f .AND. WREG	1	1	N,Z
		AND	#lit10,Wn	Wd = lit10 .AND. Wd	1	1	N,Z
		AND	Wb,Ws,Wd	Wd = Wb .AND. Ws	1	1	N,Z
		AND	Wb,#lit5,Wd	Wd = Wb .AND. lit5	1	1	N,Z
4	ASR	ASR	f	f = Arithmetic Right Shift f	1	1	C,N,OV,Z
		ASR	f,WREG	WREG = Arithmetic Right Shift f	1	1	C,N,OV,Z
		ASR	Ws,Wd	Wd = Arithmetic Right Shift Ws	1	1	C,N,OV,Z
		ASR	Wb,Wns,Wnd	Wnd = Arithmetic Right Shift Wb by Wns	1	1	N,Z
		ASR	Wb,#lit5,Wnd	Wnd = Arithmetic Right Shift Wb by lit5	1	1	N,Z
5	BCLR	BCLR	f,#bit4	Bit Clear f	1	1	None
		BCLR	Ws,#bit4	Bit Clear Ws	1	1	None
7	BRA	BRA	C,Expr	Branch if Carry	1	1 (4)	None
		BRA	GE,Expr	Branch if greater than or equal	1	1 (4)	None
		BRA	GEU,Expr	Branch if unsigned greater than or equal	1	1 (4)	None
		BRA	GT,Expr	Branch if greater than	1	1 (4)	None
		BRA	GTU,Expr	Branch if unsigned greater than	1	1 (4)	None
		BRA	LE,Expr	Branch if less than or equal	1	1 (4)	None
		BRA	LEU,Expr	Branch if unsigned less than or equal	1	1 (4)	None
		BRA	LT,Expr	Branch if less than	1	1 (4)	None
		BRA	LTU,Expr	Branch if unsigned less than	1	1 (4)	None
		BRA	N,Expr	Branch if Negative	1	1 (4)	None
		BRA	NC,Expr	Branch if Not Carry	1	1 (4)	None
		BRA	NN,Expr	Branch if Not Negative	1	1 (4)	None
		BRA	NOV,Expr	Branch if Not Overflow	1	1 (4)	None
		BRA	NZ,Expr	Branch if Not Zero	1	1 (4)	None
		BRA	OA,Expr	Branch if Accumulator A overflow	1	1 (4)	None
		BRA	OB,Expr	Branch if Accumulator B overflow	1	1 (4)	None
		BRA	OV,Expr	Branch if Overflow	1	1 (4)	None
		BRA	SA,Expr	Branch if Accumulator A saturated	1	1 (4)	None
		BRA	SB,Expr	Branch if Accumulator B saturated	1	1 (4)	None
		BRA	Expr	Branch Unconditionally	1	4	None
		BRA	Z,Expr	Branch if Zero	1	1 (4)	None
		BRA	Wn	Computed Branch	1	4	None
8	BSET	BSET	f,#bit4	Bit Set f	1	1	None
		BSET	Ws,#bit4	Bit Set Ws	1	1	None

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

24.6 MPLAB X SIM Software Simulator

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

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

24.7 MPLAB REAL ICE In-Circuit Emulator System

The MPLAB REAL ICE In-Circuit Emulator System is Microchip's next generation high-speed emulator for Microchip Flash DSC and MCU devices. It debugs and programs all 8, 16 and 32-bit MCU, and DSC devices with the easy-to-use, powerful graphical user interface of the MPLAB X IDE.

The emulator is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with either a connector compatible with in-circuit debugger systems (RJ-11) or with the new high-speed, noise tolerant, Low-Voltage Differential Signal (LVDS) interconnection (CAT5).

The emulator is field upgradable through future firmware downloads in MPLAB X IDE. MPLAB REAL ICE offers significant advantages over competitive emulators including full-speed emulation, run-time variable watches, trace analysis, complex breakpoints, logic probes, a ruggedized probe interface and long (up to three meters) interconnection cables.

24.8 MPLAB ICD 3 In-Circuit Debugger System

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

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

24.9 PICkit 3 In-Circuit Debugger/ Programmer

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

24.10 MPLAB PM3 Device Programmer

The MPLAB PM3 Device Programmer is a universal, CE compliant device programmer with programmable voltage verification at V_{DDMIN} and V_{DDMAX} for maximum reliability. It features a large LCD display (128 x 64) for menus and error messages, and a modular, detachable socket assembly to support various package types. The ICSP cable assembly is included as a standard item. In Stand-Alone mode, the MPLAB PM3 Device Programmer can read, verify and program PIC devices without a PC connection. It can also set code protection in this mode. The MPLAB PM3 connects to the host PC via an RS-232 or USB cable. The MPLAB PM3 has high-speed communications and optimized algorithms for quick programming of large memory devices, and incorporates an MMC card for file storage and data applications.

TABLE 25-6: DC CHARACTERISTICS: OPERATING CURRENT (I_{DD})

DC 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			
Parameter No.	**Typ.**	**Max.**	**Units**	**Conditions**		
Operating Current (I_{DD})[1]						
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: I_{DD} 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 I_{DD} 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)
- CLKO is configured as an I/O input pin in the Configuration Word
- All I/O pins are configured as outputs and driving low
- $\overline{\text{MCLR}}$ = V_{DD}, 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

TABLE 25-42: ADC MODULE SPECIFICATIONS

AC CHARACTERISTICS			**Standard Operating Conditions: 3.0V to 3.6V** **(unless otherwise stated)[4]** Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristics[3]**	**Min.**	**Typical**	**Max.**	**Units**	**Conditions**
Device Supply							
AD01	AVDD	Module VDD Supply	Greater of: VDD – 0.3 or 3.0	—	Lesser of: VDD + 0.3 or 3.6	V	The difference between AVDD supply and VDD supply must not exceed ±300 mV at all times, including device power-up
AD02	AVSS	Module VSS Supply	VSS	—	VSS + 0.3	V	
Analog Input							
AD12	VINH-VINL	Full-Scale Input Span	AVSS	—	AVDD	V	
AD14	VIN	Absolute Input Voltage	AVSS – 0.3	—	AVDD + 0.3	V	
AD15	VIN+	Pseudo-Differential Mode	0	—	3.3	V	VIN- = (VR+ + VR-)/2 ±150 mV
AD16	VIN-	Pseudo-Differential Mode	0	—	3.3	V	VIN+ = (VR+ + VR-)/2 ±150 mV
AD17	RIN	Recommended Impedance of Analog Voltage Source	—	100	—	Ω	For minimum sampling time **(Note 1)**
AD66	VREF1	Internal Voltage Reference Source	—	1.2	—	V	
ADC Accuracy: Pseudo-Differential Input							
AD20a	Nr	Resolution	12			bits	
AD21a	INL	Integral Nonlinearity	> -4	—	< 4	LSb	AVSS = 0V, AVDD = 3.3V
AD22a	DNL	Pseudo-Differential Nonlinearity	> -1	—	< 1	LSb	AVSS = 0V, AVDD = 3.3V **(Note 5)**
AD23a	GERR	Gain Error (Dedicated Core)	> -5	—	< 5	LSb	AVSS = 0V, AVDD = 3.3V
AD24a	EOFF	Offset Error (Dedicated Core)	> -5	—	< 5	LSb	AVSS = 0V, AVDD = 3.3V
AD25a	—	Monotonicity	—	—	—	—	Guaranteed

Note 1: These parameters are not characterized or tested in manufacturing.

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

3: Characterized with a 1 kHz sine wave.

4: The ADC module is functional at VBORMIN < VDD < VDDMIN, but with degraded performance. Unless otherwise stated, module functionality is ensured, but not characterized.

5: No missing codes, limits are based on the characterization results.

NOTES: