

Welcome to E-XFL.COM

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

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

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

#### Details

E·XFI

Product Status	Active
Core Processor	PIC
Core Size	16-Bit
Speed	60 MIPs
Connectivity	I <sup>2</sup> C, IrDA, LINbus, QEI, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, Motor Control PWM, POR, PWM, WDT
Number of I/O	35
Program Memory Size	512KB (170K x 24)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	24K x 16
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 9x10b/12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 150°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/pic24ep512mc204-h-pt

Email: info@E-XFL.COM

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

# 3.0 CPU

- Note 1: This data sheet summarizes the features of the dsPIC33EPXXXGP50X. dsPIC33EPXXXMC20X/50X and PIC24EPXXXGP/MC20X families of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to "CPU" (DS70359) 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 dsPIC33EPXXXGP50X, dsPIC33EPXXXMC20X/ 50X and PIC24EPXXXGP/MC20X CPU has a 16-bit (data) modified Harvard architecture with an enhanced instruction set, including significant support for digital signal processing. The CPU has a 24-bit instruction word with a variable length opcode field. The Program Counter (PC) is 23 bits wide and addresses up to 4M x 24 bits of user program memory space.

An instruction prefetch mechanism helps maintain throughput and provides predictable execution. Most instructions execute in a single-cycle effective execution rate, with the exception of instructions that change the program flow, the double-word move (MOV.D) instruction, PSV accesses and the table instructions. Overhead-free program loop constructs are supported using the DO and REPEAT instructions, both of which are interruptible at any point.

# 3.1 Registers

The dsPIC33EPXXXGP50X, dsPIC33EPXXXMC20X/ 50X and PIC24EPXXXGP/MC20X devices have sixteen, 16-bit working registers in the programmer's model. Each of the working registers can act as a data, address or address offset register. The 16th working register (W15) operates as a Software Stack Pointer for interrupts and calls.

# 3.2 Instruction Set

The instruction set for dsPIC33EPXXXGP50X and dsPIC33EPXXXMC20X/50X devices has two classes of instructions: the MCU class of instructions and the DSP class of instructions. The instruction set for PIC24EPXXXGP/MC20X devices has the MCU class of instructions only and does not support DSP instructions. These two instruction classes are seamlessly integrated into the architecture and execute from a single execution unit. The instruction set includes many addressing modes and was designed for optimum C compiler efficiency.

## 3.3 Data Space Addressing

The base Data Space can be addressed as 64 Kbytes (32K words).

The Data Space includes two ranges of memory, referred to as X and Y data memory. Each memory range is accessible through its own independent Address Generation Unit (AGU). The MCU class of instructions operates solely through the X memory AGU, which accesses the entire memory map as one linear Data Space. On dsPIC33EPXXXMC20X/50X and dsPIC33EPXXXGP50X devices, certain DSP instructions operate through the X and Y AGUs to support dual operand reads, which splits the data address space into two parts. The X and Y Data Spaces have memory locations that are device-specific, and are described further in the data memory maps in **Section 4.2 "Data Address Space"**.

The upper 32 Kbytes of the Data Space memory map can optionally be mapped into Program Space (PS) at any 32-Kbyte aligned program word boundary. The Program-to-Data Space mapping feature, known as Program Space Visibility (PSV), lets any instruction access Program Space as if it were Data Space. Moreover, the Base Data Space address is used in conjunction with a Read or Write Page register (DSRPAG or DSWPAG) to form an Extended Data Space (EDS) address. The EDS can be addressed as 8M words or 16 Mbytes. Refer to the "**Data Memory**" (DS70595) and "**Program Memory**" (DS70613) sections in the "*dsPIC33/PIC24 Family Reference Manual*" for more details on EDS, PSV and table accesses.

On the dsPIC33EPXXXMC20X/50X and dsPIC33EPXXXGP50X devices, overhead-free circular buffers (Modulo Addressing) are supported in both X and Y address spaces. The Modulo Addressing removes the software boundary checking overhead for DSP algorithms. The X AGU Circular Addressing can be used with any of the MCU class of instructions. The X AGU also supports Bit-Reversed Addressing to greatly simplify input or output data re-ordering for radix-2 FFT algorithms. PIC24EPXXXGP/MC20X devices do not support Modulo and Bit-Reversed Addressing.

# 3.4 Addressing Modes

The CPU supports these addressing modes:

- Inherent (no operand)
- Relative
- Literal
- · Memory Direct
- Register Direct
- Register Indirect

Each instruction is associated with a predefined addressing mode group, depending upon its functional requirements. As many as six addressing modes are supported for each instruction.

### 3.8 Arithmetic Logic Unit (ALU)

The dsPIC33EPXXXGP50X, dsPIC33EPXXXMC20X/ 50X and PIC24EPXXXGP/MC20X ALU is 16 bits wide, and is capable of addition, subtraction, bit shifts and logic operations. Unless otherwise mentioned, arithmetic operations are two's complement in nature. Depending on the operation, the ALU can affect the values of the Carry (C), Zero (Z), Negative (N), Overflow (OV) and Digit Carry (DC) Status bits in the <u>SR register. The C and DC</u> Status bits operate as Borrow and Digit Borrow bits, respectively, for subtraction operations.

The ALU can perform 8-bit or 16-bit operations, depending on the mode of the instruction that is used. Data for the ALU operation can come from the W register array or data memory, depending on the addressing mode of the instruction. Likewise, output data from the ALU can be written to the W register array or a data memory location.

Refer to the *"16-bit MCU and DSC Programmer's Reference Manual"* (DS70157) for information on the SR bits affected by each instruction.

The core CPU incorporates hardware support for both multiplication and division. This includes a dedicated hardware multiplier and support hardware for 16-bit divisor division.

#### 3.8.1 MULTIPLIER

Using the high-speed 17-bit x 17-bit multiplier, the ALU supports unsigned, signed, or mixed-sign operation in several MCU multiplication modes:

- 16-bit x 16-bit signed
- 16-bit x 16-bit unsigned
- 16-bit signed x 5-bit (literal) unsigned
- 16-bit signed x 16-bit unsigned
- 16-bit unsigned x 5-bit (literal) unsigned
- 16-bit unsigned x 16-bit signed
- 8-bit unsigned x 8-bit unsigned

### 3.8.2 DIVIDER

The divide block supports 32-bit/16-bit and 16-bit/16-bit signed and unsigned integer divide operations with the following data sizes:

- 32-bit signed/16-bit signed divide
- 32-bit unsigned/16-bit unsigned divide
- 16-bit signed/16-bit signed divide
- 16-bit unsigned/16-bit unsigned divide

The quotient for all divide instructions ends up in W0 and the remainder in W1. The 16-bit signed and unsigned DIV instructions can specify any W register for both the 16-bit divisor (Wn) and any W register (aligned) pair (W(m + 1):Wm) for the 32-bit dividend. The divide algorithm takes one cycle per bit of divisor, so both 32-bit/16-bit and 16-bit/16-bit instructions take the same number of cycles to execute.

## 3.9 DSP Engine (dsPIC33EPXXXMC20X/50X and dsPIC33EPXXXGP50X Devices Only)

The DSP engine consists of a high-speed 17-bit x 17-bit multiplier, a 40-bit barrel shifter and a 40-bit adder/subtracter (with two target accumulators, round and saturation logic).

The DSP engine can also perform inherent accumulatorto-accumulator operations that require no additional data. These instructions are ADD, SUB and NEG.

The DSP engine has options selected through bits in the CPU Core Control register (CORCON), as listed below:

- Fractional or integer DSP multiply (IF)
- · Signed, unsigned or mixed-sign DSP multiply (US)
- · Conventional or convergent rounding (RND)
- · Automatic saturation on/off for ACCA (SATA)
- Automatic saturation on/off for ACCB (SATB)
- Automatic saturation on/off for writes to data memory (SATDW)
- Accumulator Saturation mode selection (ACCSAT)

	SOMMAN	
Instruction	Algebraic Operation	ACC Write Back
CLR	A = 0	Yes
ED	$A = (x - y)^2$	No
EDAC	$A = A + (x - y)^2$	No
MAC	$A = A + (x \bullet y)$	Yes
MAC	$A = A + x^2$	No
MOVSAC	No change in A	Yes
MPY	$A = x \bullet y$	No
MPY	$A = x^2$	No
MPY.N	$A = -x \bullet y$	No
MSC	$A = A - x \bullet y$	Yes

TABLE 3-2: DSP INSTRUCTIONS SUMMARY

TABLE 4	4-9:	INPU		URE 1 T	HROUG	SH INPU	IT CAPI	URE 4	REGIST	ER MA	Р							
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
IC1CON1	0140	_	_	ICSIDL		CTSEL<2:0	>	_	_	_	ICI<	1:0>	ICOV	ICBNE		ICM<2:0>		0000
IC1CON2	0142	_	—	_	—	—	—	—	IC32	ICTRIG	TRIGSTAT	_		S	/NCSEL<4	:0>		000D
IC1BUF	0144							Inp	ut Capture	1 Buffer Re	gister							xxxx
IC1TMR	0146								Input Cap	ture 1 Time	r							0000
IC2CON1	0148	_	—	ICSIDL		CTSEL<2:0	>	_	—	_	ICI<'	1:0>	ICOV	ICBNE		ICM<2:0>		0000
IC2CON2	014A	_																
IC2BUF	014C							Inp	ut Capture	2 Buffer Re	gister							xxxx
IC2TMR	014E								Input Cap	ture 2 Time	r							0000
IC3CON1	0150	_	—	ICSIDL		CTSEL<2:0	>	_	—	_	ICI<'	1:0>	ICOV	ICBNE		ICM<2:0>		0000
IC3CON2	0152	_	—	_	—	—	—	—	IC32	ICTRIG	TRIGSTAT	_		S	/NCSEL<4	:0>		000D
IC3BUF	0154							Inp	ut Capture	3 Buffer Re	gister							xxxx
IC3TMR	0156								Input Cap	ture 3 Time	r							0000
IC4CON1	0158	_	_	ICSIDL	I	CTSEL<2:0	>	_	_	_	ICI<	1:0>	ICOV	ICBNE		ICM<2:0>		0000
IC4CON2	015A	_	IC32 ICTRIG TRIGSTAT - SYNCSEL<4:0> 000D															
IC4BUF	015C		•	•	•	•	•	Inp	ut Capture	4 Buffer Re	gister	•	•					xxxx
IC4TMR	015E		Input Capture 4 Timer 0000															

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

# TABLE 4-19: SPI1 AND SPI2 REGISTER MAP

SFR Name	Addr.	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
SPI1STAT	0240	SPIEN	—	SPISIDL	—	—	:	SPIBEC<2:0	)>	SRMPT	SPIROV	SRXMPT		SISEL<2:0>		SPITBF	SPIRBF	0000
SPI1CON1	0242	_	_	_	DISSCK	DISSDO	MODE16	SMP	CKE	SSEN	CKP	MSTEN		SPRE<2:0>		PPRE	<1:0>	0000
SPI1CON2	0244	FRMEN	SPIFSD	FRMPOL	_	_	_	_	_	_	_	—	_	_	_	FRMDLY	SPIBEN	0000
SPI1BUF	0248							SPI1 Tra	ansmit and F	Receive Buf	fer Registe	r						0000
SPI2STAT	0260	SPIEN	—	SPISIDL	—	—	:	SPIBEC<2:0	)>	SRMPT	SPIROV	SRXMPT		SISEL<2:0>		SPITBF	SPIRBF	0000
SPI2CON1	0262	_	—		DISSCK	DISSDO	MODE16	SMP	CKE	SSEN	CKP	MSTEN		SPRE<2:0>		PPRE	<1:0>	0000
SPI2CON2	0264	FRMEN	SPIFSD	FRMPOL	_	_	_	_	_	_	_	—	_	_	_	FRMDLY	SPIBEN	0000
SPI2BUF	0268							SPI2 Tra	ansmit and F	Receive Buf	fer Registe	r						0000

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

### TABLE 4-37: PMD REGISTER MAP FOR PIC24EPXXXGP20X DEVICES ONLY

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
PMD1	0760	T5MD	T4MD	T3MD	T2MD	T1MD	-	-	-	I2C1MD	U2MD	U1MD	SPI2MD	SPI1MD	—	—	AD1MD	0000
PMD2	0762	_	_	—	_	IC4MD	IC3MD	IC2MD	IC1MD	_	_	_	-	OC4MD	OC3MD	OC2MD	OC1MD	0000
PMD3	0764	_	_	_	_	_	CMPMD	_	_	CRCMD	_	_	_	_	_	I2C2MD	_	0000
PMD4	0766	_	_	_	_	_	_	_	_	_	_	_	_	REFOMD	CTMUMD	_	_	0000
PMD6	076A	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	0000
													DMA0MD					
	0760												DMA1MD	DTCMD				0000
PIVID7	0760	_	_	_	_	_	_	_	_	_	_	_	DMA2MD	PIGMD	_	_	_	0000
													DMA3MD					

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

#### TABLE 4-38: PMD REGISTER MAP FOR PIC24EPXXXMC20X DEVICES ONLY

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
PMD1	0760	T5MD	T4MD	T3MD	T2MD	T1MD	QEI1MD	PWMMD	_	I2C1MD	U2MD	U1MD	SPI2MD	SPI1MD	_	_	AD1MD	0000
PMD2	0762	_	—	—	—	IC4MD	IC3MD	IC2MD	IC1MD	—	_	—		OC4MD	OC3MD	OC2MD	OC1MD	0000
PMD3	0764	_	_	_	_	_	CMPMD	_	_	CRCMD	_	_	_	_	_	I2C2MD	_	0000
PMD4	0766	_	_	_	_	_	_	_	_	_	_	_	_	REFOMD	CTMUMD	_	_	0000
PMD6	076A	_	_	_	_	_	PWM3MD	PWM2MD	PWM1MD	_	_	_	_	_	_	_	_	0000
													DMA0MD					
	0760												DMA1MD	DTCMD				0000
FIND	0/00	_	_	_	_	_	_	_	_	_	_	_	DMA2MD	FIGND	_	_		0000
													DMA3MD					1

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

### 4.4.4 SOFTWARE STACK

The W15 register serves as a dedicated Software Stack Pointer (SSP) and is automatically modified by exception processing, subroutine calls and returns; however, W15 can be referenced by any instruction in the same manner as all other W registers. This simplifies reading, writing and manipulating of the Stack Pointer (for example, creating stack frames).

Note:	То	protec	t	agains	st	misal	stack			
	acc	esses,	W	15<0>	is	fixed	to	'0'	by	the
	hard	dware.								

W15 is initialized to 0x1000 during all Resets. This address ensures that the SSP points to valid RAM in all dsPIC33EPXXXGP50X, dsPIC33EPXXXMC20X/50X and PIC24EPXXXGP/MC20X devices, and permits stack availability for non-maskable trap exceptions. These can occur before the SSP is initialized by the user software. You can reprogram the SSP during initialization to any location within Data Space.

The Software Stack Pointer always points to the first available free word and fills the software stack working from lower toward higher addresses. Figure 4-19 illustrates how it pre-decrements for a stack pop (read) and post-increments for a stack push (writes).

When the PC is pushed onto the stack, PC<15:0> are pushed onto the first available stack word, then PC<22:16> are pushed into the second available stack location. For a PC push during any CALL instruction, the MSB of the PC is zero-extended before the push, as shown in Figure 4-19. During exception processing, the MSB of the PC is concatenated with the lower 8 bits of the CPU STATUS Register, SR. This allows the contents of SRL to be preserved automatically during interrupt processing.

- **Note 1:** To maintain system Stack Pointer (W15) coherency, W15 is never subject to (EDS) paging, and is therefore restricted to an address range of 0x0000 to 0xFFFF. The same applies to the W14 when used as a Stack Frame Pointer (SFA = 1).
  - 2: As the stack can be placed in, and can access X and Y spaces, care must be taken regarding its use, particularly with regard to local automatic variables in a C development environment

FIGURE 4-19: CALL STACK FRAME



#### FIGURE 7-1: dsPIC33EPXXXGP50X, dsPIC33EPXXXMC20X/50X AND PIC24EPXXXGP/MC20X INTERRUPT VECTOR TABLE



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
—				OCFAR<6:0>	>		
bit 7	-						bit 0
Legend:							

### REGISTER 11-6: RPINR11: PERIPHERAL PIN SELECT INPUT REGISTER 11

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read	l 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 6-0 OCFAR<6:0>: Assign Output Compare Fault A (OCFA) to the Corresponding RPn Pin bits (see Table 11-2 for input pin selection numbers) 1111001 = Input tied to RPI121

> . 0000001 = Input tied to CMP1 0000000 = Input tied to Vss

# 15.2 Output Compare Control Registers

# REGISTER 15-1: OCxCON1: OUTPUT COMPARE x CONTROL REGISTER 1

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0
		OCSIDL	OCTSEL2	OCTSEL1	OCTSEL0	_	ENFLTB
bit 15							bit 8
R/W-0	U-0	R/W-0, HSC	R/W-0, HSC	R/W-0	R/W-0	R/W-0	R/W-0
ENFLTA		OCFLTB	OCFLTA	TRIGMODE	OCM2	OCM1	OCM0
bit 7							bit 0
Legend:		HSC = Hardw	are Settable/Cl	earable bit			
R = Reada	ible bit	W = Writable I	bit	U = Unimplem	nented bit, read	as '0'	
-n = Value	at POR	'1' = Bit is set		'0' = Bit is clea	ared	x = Bit is unkr	nown
bit 15-14	Unimplemen	ted: Read as '0	)'				
bit 13	OCSIDL: Out	tput Compare x	Stop in Idle Mo	de Control bit			
	1 = Output C	compare x Halts	in CPU Idle me	ode via CDU Idia m	odo		
bit 12 10			nues lo operale		oue		
DIL 12-10	111 = Perinh	eral clock (Ep)	pare x Clock S				
	110 = Reserv	/ed					
	101 <b>= PTGO</b>	x clock <sup>(2)</sup>					
	100 = T1CLK	is the clock so	urce of the OC	k (only the sync	hronous clock	is supported)	
	011 = 15CLK	is the clock sou	urce of the OC	х ~			
	010 = T4CLK 001 = T3CLK	is the clock so	urce of the OC	x X			
	000 = T2CLK	is the clock so	urce of the OC	ĸ			
bit 9	Unimplemen	ted: Read as '0	)'				
bit 8	ENFLTB: Fau	ult B Input Enab	le bit				
	1 = Output C 0 = Output C	compare Fault B compare Fault B	input (OCFB) input (OCFB)	is enabled is disabled			
bit 7	ENFLTA: Fau	ult A Input Enabl	le bit				
	1 = Output C	ompare Fault A	input (OCFA)	is enabled			
	0 = Output C	ompare Fault A	input (OCFA)	is disabled			
bit 6	Unimplemen	ted: Read as '0	)'				
bit 5	OCFLTB: PW	M Fault B Cond	dition Status bit				
	1 = PWM Fa 0 = No PWM	ult B condition of Fault B condition	on OCFB pin ha on on OCFB pi	as occurred n has occurred			
bit 4	OCFLTA: PW	/M Fault A Cond	dition Status bit				
	1 = PWM Fa	ult A condition of	on OCFA pin ha	as occurred			
	0 = No PWM	I Fault A condition	on on OCFA pi	n has occurred			
Note 1:	OCxR and OCxF	RS are double-b	ouffered in PWN	A mode only.			
2:	Each Output Cor	mpare x module	(OCx) has one	PTG clock sou	urce. See <b>Secti</b>	on 24.0 "Perip	oheral Trigger
	Generator (PTG PTGO4 = OC1	) wodule" for r	nore informatio	n.			
	PTGO5 = OC2						
	PTGO6 = OC3						
	PTGO7 = OC4						

REGISTER 16-2: PTCON2: PWMx PRIMARY MASTER CLOCK DIVIDER SELECT REGISTER
--

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	_	_	—	—	—
bit 15							bit 8
U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	_	—	—	—	PCLKDIV2 <sup>(1)</sup>	PCLKDIV1 <sup>(1)</sup>	PCLKDIV0(1)
bit 7							bit 0
Legend:							
R = Readable	bit	W = Writable	bit	U = Unimpler	mented bit, read	as '0'	
-n = Value at F	POR	'1' = Bit is set		'0' = Bit is cle	ared	x = Bit is unkn	iown
bit 15 2	Unimplomon	tod. Dood on '	۰ <b>'</b>				

#### bit 15-3 Unimplemented: Read as '0'

bit 2-0 PCLKDIV<2:0>: PWMx Input Clock Prescaler (Divider) Select bits<sup>(1)</sup>

- 111 = Reserved 110 = Divide-by-64 101 = Divide-by-32
- 100 = Divide-by-32100 = Divide-by-16
- 011 = Divide-by-8
- 010 = Divide-by-4
- 001 = Divide-by-2
- 000 = Divide-by-1, maximum PWMx timing resolution (power-on default)
- **Note 1:** These bits should be changed only when PTEN = 0. Changing the clock selection during operation will yield unpredictable results.

#### FIGURE 17-1: QEI BLOCK DIAGRAM





#### FIGURE 18-1: SPIx MODULE BLOCK DIAGRAM

R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	
ADRC	—	—	SAMC4 <sup>(1)</sup>	SAMC3 <sup>(1)</sup>	SAMC2 <sup>(1)</sup>	SAMC1 <sup>(1)</sup>	SAMC0 <sup>(1)</sup>	
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	
ADCS7(2	<sup>2)</sup> ADCS6 <sup>(2)</sup>	ADCS5 <sup>(2)</sup>	ADCS4 <sup>(2)</sup>	ADCS3 <sup>(2)</sup>	ADCS2 <sup>(2)</sup>	ADCS1 <sup>(2)</sup>	ADCS0 <sup>(2)</sup>	
bit 7							bit 0	
Legend:								
R = Reada		vv = vvritable t	DIT		nented bit, read			
-n = value	at POR	"1" = Bit is set		$0^{\circ}$ = Bit is cle	ared	x = Bit is unkr	nown	
bit 15	ADRC: ADC1 1 = ADC inter 0 = Clock deri	Conversion Cl nal RC clock ved from syste	ock Source bit m clock	:				
bit 14-13	Unimplement	ted: Read as '0	3					
bit 12-8	SAMC<4:0>:	Auto-Sample T	ime bits <sup>(1)</sup>					
	11111 = 31 T. • • • • • •	AD						
hit 7 0	00000 = 0 IA		ion Clock Colo	at hita(2)				
Dit 7-0	$00000 = 0 \text{ TAD}$ t 7-0 $ADCS<7:0>: ADC1 \text{ Conversion Clock Select bits}^{(2)}$ $11111111 = TP \cdot (ADCS<7:0> + 1) = TP \cdot 256 = TAD$ $\cdot$ $\cdot$ $00000010 = TP \cdot (ADCS<7:0> + 1) = TP \cdot 3 = TAD$ $00000010 = TP \cdot (ADCS<7:0> + 1) = TP \cdot 2 = TAD$ $00000001 = TP \cdot (ADCS<7:0> + 1) = TP \cdot 2 = TAD$ $00000000 = TP \cdot (ADCS<7:0> + 1) = TP \cdot 1 = TAD$							
Note 1: 2:	This bit is only use This bit is not used	d if SSRC<2:0> if ADRC (AD10	· (AD1CON1< CON3<15>) =	7:5>) = 111 ar 1.	nd SSRCG (AD	1CON1<4>) =	0.	

#### REGISTER 23-3: AD1CON3: ADC1 CONTROL REGISTER 3

### REGISTER 23-6: AD1CHS0: ADC1 INPUT CHANNEL 0 SELECT REGISTER (CONTINUED)

bit 4-0	CH0SA<4:0>: Channel 0 Positive Input Select for Sample MUXA bits <sup>(1)</sup>
	11111 = Open; use this selection with CTMU capacitive and time measurement
	11110 = Channel 0 positive input is connected to the CTMU temperature measurement diode (CTMU TEMP)
	11101 - Reserved
	11011 = Reserved
	11010 = Channel 0 positive input is the output of OA3/AN6 <sup>(2,3)</sup>
	11001 = Channel 0 positive input is the output of OA2/AN0 <sup>(2)</sup>
	11000 = Channel 0 positive input is the output of OA1/AN3 <sup>(-)</sup>
	•
	•
	•
	10000 = Reserved
	01111 = Channel 0 positive input is AN15 <sup>(1,3)</sup>
	01110 = Channel 0 positive input is AN14 <sup>(1,3)</sup>
	01101 = Channel 0 positive input is AN13 <sup>(1,3)</sup>
	•
	•
	• (1 2)
	00010 = Channel 0 positive input is AN2 <sup>(1,3)</sup>
	00001 = Channel 0 positive input is AN1(1,3)
	00000 = Channel 0 positive input is AN0(',3)

- **Note 1:** AN0 through AN7 are repurposed when comparator and op amp functionality is enabled. See Figure 23-1 to determine how enabling a particular op amp or comparator affects selection choices for Channels 1, 2 and 3.
  - 2: The OAx input is used if the corresponding op amp is selected (OPMODE (CMxCON<10>) = 1); otherwise, the ANx input is used.
  - 3: See the "Pin Diagrams" section for the available analog channels for each device.

# REGISTER 24-4: PTGT0LIM: PTG TIMER0 LIMIT REGISTER<sup>(1)</sup>

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
			PTGT0	LIM<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
			PTGTC	LIM<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		'1' = Bit is set		'0' = Bit is cleared			nown

#### bit 15-0 **PTGT0LIM<15:0>:** PTG Timer0 Limit Register bits General Purpose Timer0 Limit register (effective only with a PTGT0 Step command).

**Note 1:** This register is read-only when the PTG module is executing Step commands (PTGEN = 1 and PTGSTRT = 1).

# REGISTER 24-5: PTGT1LIM: PTG TIMER1 LIMIT REGISTER<sup>(1)</sup>

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
			PTGT1LI	IM<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
			PTGT1L	_IM<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 **PTGT1LIM<15:0>:** PTG Timer1 Limit Register bits

General Purpose Timer1 Limit register (effective only with a PTGT1 Step command).

**Note 1:** This register is read-only when the PTG module is executing Step commands (PTGEN = 1 and PTGSTRT = 1).

#### REGISTER 25-4: CMxMSKSRC: COMPARATOR x MASK SOURCE SELECT CONTROL REGISTER

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	RW-0
—	—	—	—	SELSRCC3	SELSRCC2	SELSRCC1	SELSRCC0
bit 15							bit 8

| R/W-0    |
|----------|----------|----------|----------|----------|----------|----------|----------|
| SELSRCB3 | SELSRCB2 | SELSRCB1 | SELSRCB0 | SELSRCA3 | SELSRCA2 | SELSRCA1 | SELSRCA0 |
| 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 11-8	SELSRCC<3:0>: Mask C Input Select bits
	1111 <b>= FLT4</b>
	1110 <b>= FLT2</b>
	1101 <b>= PTGO19</b>
	1100 <b>= PTGO18</b>
	1011 = Reserved
	1010 = Reserved
	1001 = Reserved
	1000 = Reserved
	0111 = Reserved
	0110 = Reserved
	0101 <b>= PWM3H</b>
	0100 = PWM3L
	0011 = PWM2H
	0010 = PWM2L
	0001 = PWM1H
	0000 = PWM1L
bit 7-4	SELSRCB<3:0>: Mask B Input Select bits
bit 7-4	SELSRCB<3:0>: Mask B Input Select bits 1111 = FLT4
bit 7-4	SELSRCB<3:0>: Mask B Input Select bits 1111 = FLT4 1110 = FLT2
bit 7-4	SELSRCB<3:0>: Mask B Input Select bits 1111 = FLT4 1110 = FLT2 1101 = PTGO19
bit 7-4	SELSRCB<3:0>: Mask B Input Select bits 1111 = FLT4 1110 = FLT2 1101 = PTGO19 1100 = PTGO18
bit 7-4	SELSRCB<3:0>: Mask B Input Select bits 1111 = FLT4 1110 = FLT2 1101 = PTGO19 1100 = PTGO18 1011 = Reserved
bit 7-4	SELSRCB<3:0>: Mask B Input Select bits 1111 = FLT4 1110 = FLT2 1101 = PTGO19 1100 = PTGO18 1011 = Reserved 1010 = Reserved
bit 7-4	SELSRCB<3:0>: Mask B Input Select bits 1111 = FLT4 1110 = FLT2 1101 = PTGO19 1100 = PTGO18 1011 = Reserved 1010 = Reserved 1001 = Reserved
bit 7-4	SELSRCB<3:0>: Mask B Input Select bits 1111 = FLT4 1110 = FLT2 1101 = PTGO19 1100 = PTGO18 1011 = Reserved 1010 = Reserved 1001 = Reserved 1000 = Reserved
bit 7-4	SELSRCB<3:0>: Mask B Input Select bits 1111 = FLT4 1110 = FLT2 1101 = PTGO19 1100 = PTGO18 1011 = Reserved 1001 = Reserved 1000 = Reserved 0011 = Reserved 0111 = Reserved
bit 7-4	SELSRCB<3:0>: Mask B Input Select bits 1111 = FLT4 1110 = FLT2 1101 = PTGO19 1100 = PTGO18 1011 = Reserved 1010 = Reserved 1000 = Reserved 0111 = Reserved 0111 = Reserved 0110 = Reserved 0110 = Reserved
bit 7-4	SELSRCB<3:0>: Mask B Input Select bits 1111 = FLT4 1110 = FLT2 1101 = PTGO19 1100 = PTGO18 1011 = Reserved 1010 = Reserved 1001 = Reserved 1000 = Reserved 0111 = Reserved 0110 = Reserved 0110 = Reserved 0110 = PWM3H 0100 = PWM3I
bit 7-4	SELSRCB<3:0>: Mask B Input Select bits 1111 = FLT4 1110 = FLT2 1101 = PTGO19 1100 = PTGO18 1011 = Reserved 1010 = Reserved 1000 = Reserved 0111 = Reserved 0111 = Reserved 0110 = Reserved 0110 = PWM3H 0100 = PWM3L 0011 = PWM2H
bit 7-4	SELSRCB<3:0>: Mask B Input Select bits 1111 = FLT4 1110 = FLT2 1101 = PTGO19 1100 = PTGO18 1011 = Reserved 1010 = Reserved 1001 = Reserved 1000 = Reserved 0111 = Reserved 0110 = Reserved 0110 = PWM3H 0100 = PWM3L 0011 = PWM2H 0010 = PWM2I
bit 7-4	SELSRCB<3:0>: Mask B Input Select bits 1111 = FLT4 1110 = FLT2 1101 = PTGO19 1100 = PTGO18 1011 = Reserved 1010 = Reserved 1001 = Reserved 0101 = Reserved 0111 = Reserved 0110 = Reserved 0110 = PWM3H 0100 = PWM3L 0011 = PWM2H 0010 = PWM2L 0001 = PWM1H
bit 7-4	SELSRCB<3:0>: Mask B Input Select bits 1111 = FLT4 1110 = FLT2 1101 = PTGO19 1100 = PTGO18 1011 = Reserved 1010 = Reserved 1001 = Reserved 0101 = Reserved 0111 = Reserved 0110 = Reserved 0110 = PWM3H 0100 = PWM3L 0011 = PWM2H 0010 = PWM2L 0001 = PWM1H 0000 = PWM1I

REGISTER 25-5:	CMxMSKCON: COMPARATOR x MASK GATING
	CONTROL REGISTER

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0				
HLMS		OCEN	OCNEN	OBEN	OBNEN	OAEN	OANEN				
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				
NAGS	PAGS	ACEN	ACNEN	ABEN	ABNEN	AAEN	AANEN				
bit 7							bit 0				
Legend											
R = Readable	e hit	W = Writable	hit	=   Inimple	mented hit read	1 as '0'					
n = Value at	POR	'1' = Rit is set		(0) = 0	eared	x = Ritis unk	nown				
	1010	1 - Dit 13 3C			carca		nown				
bit 15	HLMS: Hiah	or Low-Level	/asking Select	bits							
	1 = The mask	king (blanking)	function will pre	event any asse	erted ('0') compa	rator signal fro	m propagating				
	0 = The mas	king (blanking)	function will pre	event any asse	erted ('1') compa	rator signal fro	m propagating				
bit 14	Unimpleme	nted: Read as	'0'								
bit 13	OCEN: OR (	Gate C Input Er	nable bit								
	1 = MCI is co	1 = MCI is connected to OR gate									
	0 = MCI is no	ot connected to	OR gate								
bit 12	OCNEN: OR Gate C Input Inverted Enable bit										
	1 = Inverted MCI is connected to OR gate										
hit 11		0 – Inventeu ivici is not connected to OK gate <b>OBEN:</b> OR Gate B Input Enable bit									
bit II	1 = MBI is co	UBEN: OR Gate B Input Enable bit 1 = MBL is connected to OR gate									
	0 = MBI is no	ot connected to	OR gate								
bit 10	OBNEN: OR	OBNEN: OR Gate B Input Inverted Enable bit									
	1 = Inverted	MBI is connect	ed to OR gate								
	0 = Inverted MBI is not connected to OR gate										
bit 9	OAEN: OR (	Gate A Input Er	nable bit								
	1 = MAI is connected to OR gate										
hit 8			Norted Enable	o hit							
DILO	1 = Inverted	OANEN: OR Gate A Input Inverted Enable bit									
	0 = Inverted MAI is not connected to OR gate										
bit 7	NAGS: AND	Gate Output In	nverted Enable	e bit							
	1 = Inverted	ANDI is conne	cted to OR gat	e							
	0 = Inverted	ANDI is not co		gate							
bit 6		PAGS: AND Gate Output Enable bit									
	1 = ANDI is 0 0 = ANDI is r	not connected to O	to OR gate								
bit 5	ACEN: AND	Gate C Input E	Enable bit								
	1 = MCI is co	onnected to AN	D gate								
	0 = MCI is no	ot connected to	AND gate								
bit 4	ACNEN: AN	D Gate C Input	Inverted Enab	ole bit							
	1 = Inverted	MCI is connect	ed to AND gat	e,							
	0 = Inverted	MCI is not con	nected to AND	gate							

Bit Field	Description
WDTPRE	Watchdog Timer Prescaler bit 1 = 1:128 0 = 1:32
WDTPOST<3:0>	Watchdog Timer Postscaler bits 1111 = 1:32,768 1110 = 1:16,384 • • • • • • • • • • • • •
WDTWIN<1:0>	Watchdog Window Select bits 11 = WDT window is 25% of WDT period 10 = WDT window is 37.5% of WDT period 01 = WDT window is 50% of WDT period 00 = WDT window is 75% of WDT period
ALTI2C1	Alternate I2C1 pin 1 = I2C1 is mapped to the SDA1/SCL1 pins 0 = I2C1 is mapped to the ASDA1/ASCL1 pins
ALTI2C2	Alternate I2C2 pin 1 = I2C2 is mapped to the SDA2/SCL2 pins 0 = I2C2 is mapped to the ASDA2/ASCL2 pins
JTAGEN <sup>(2)</sup>	JTAG Enable bit 1 = JTAG is enabled 0 = JTAG is disabled
ICS<1:0>	ICD Communication Channel Select bits 11 = Communicate on PGEC1 and PGED1 10 = Communicate on PGEC2 and PGED2 01 = Communicate on PGEC3 and PGED3 00 = Reserved, do not use

### TABLE 27-2: CONFIGURATION BITS DESCRIPTION (CONTINUED)

Note 1: This bit is only available on dsPIC33EPXXXMC20X/50X and PIC24EPXXXMC20X devices.

2: When JTAGEN = 1, an internal pull-up resistor is enabled on the TMS pin. Erased devices default to JTAGEN = 1. Applications requiring I/O pins in a high-impedance state (tri-state) in Reset should use pins other than TMS for this purpose.

AC CHARACTERISTICS			$\begin{array}{l} \mbox{Standard Operating Conditions: 3.0V to 3.6V} \\ \mbox{(unless otherwise stated)}^{(1)} \\ \mbox{Operating temperature} & -40^{\circ}C \leq TA \leq +85^{\circ}C \mbox{ for Industrial} \\ & -40^{\circ}C \leq TA \leq +125^{\circ}C \mbox{ for Extended} \end{array}$							
Param No.	Symbol	Characteristic	Min.	Тур.	Max.	Units	Conditions			
		ADC /	Accuracy	/ (12-Bit	Mode)					
AD20a	Nr	Resolution	12 Data Bits		bits					
AD21a	INL	Integral Nonlinearity	-2.5		2.5	LSb	-40°C ≤ TA ≤ +85°C (Note 2)			
			-5.5	_	5.5	LSb	+85°C < TA $\leq$ +125°C (Note 2)			
AD22a	DNL	Differential Nonlinearity	-1		1	LSb	-40°C $\leq$ TA $\leq$ +85°C (Note 2)			
			-1		1	LSb	+85°C < TA $\leq$ +125°C (Note 2)			
AD23a	Gerr	Gain Error <sup>(3)</sup>	-10		10	LSb	-40°C $\leq$ TA $\leq$ +85°C (Note 2)			
			-10		10	LSb	+85°C < TA $\leq$ +125°C (Note 2)			
AD24a	EOFF	Offset Error	-5		5	LSb	$-40^{\circ}C \le TA \le +85^{\circ}C$ (Note 2)			
			-5		5	LSb	+85°C < TA $\leq$ +125°C (Note 2)			
AD25a	—	Monotonicity	—			—	Guaranteed			
Dynamic Performance (12-Bit Mode)										
AD30a	THD	Total Harmonic Distortion <sup>(3)</sup>	—	75		dB				
AD31a	SINAD	Signal to Noise and Distortion <sup>(3)</sup>		68	-	dB				
AD32a	SFDR	Spurious Free Dynamic Range <sup>(3)</sup>	_	80	_	dB				
AD33a	Fnyq	Input Signal Bandwidth <sup>(3)</sup>	_	250		kHz				
AD34a	ENOB	Effective Number of Bits <sup>(3)</sup>	11.09	11.3	_	bits				

## TABLE 30-58: ADC MODULE SPECIFICATIONS (12-BIT MODE)

**Note 1:** Device is functional at VBORMIN < VDD < VDDMIN, but will have degraded performance. Device functionality is tested, but not characterized. Analog modules (ADC, op amp/comparator and comparator voltage reference) may have degraded performance. Refer to Parameter BO10 in Table 30-13 for the minimum and maximum BOR values.

2: For all accuracy specifications, VINL = AVSS = VREFL = 0V and AVDD = VREFH = 3.6V.

3: Parameters are characterized but not tested in manufacturing.

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature $-40^{\circ}C \le TA \le +150^{\circ}C$					
Param.	Symbol	Characteristic	Min.	Тур.	Max.	Units	Conditions	
HDO10	Vol	Output Low Voltage 4x Sink Driver Pins <sup>(2)</sup>	_	—	0.4	V	IOL ≤ 5 mA, VDD = 3.3V (Note 1)	
		Output Low Voltage 8x Sink Driver Pins <sup>(3)</sup>	—	_	0.4	V	IOL ≤ 8 mA, VDD = 3.3V (Note 1)	
HDO20	Vон	Output High Voltage 4x Source Driver Pins <sup>(2)</sup>	2.4	—	—	V	IOH ≥ -10 mA, VDD = 3.3V (Note 1)	
		Output High Voltage 8x Source Driver Pins <sup>(3)</sup>	2.4	—	—	V	IOH ≥ 15 mA, VDD = 3.3V <b>(Note 1)</b>	
HDO20A	Voh1	Output High Voltage 4x Source Driver Pins <sup>(2)</sup>	1.5	—	—	V	IOH ≥ -3.9 mA, VDD = 3.3V (Note 1)	
			2.0	—	—		IOH ≥ -3.7 mA, VDD = 3.3V (Note 1)	
			3.0	—	—		IOH ≥ -2 mA, VDD = 3.3V (Note 1)	
		Output High Voltage 8x Source Driver Pins <sup>(3)</sup>	1.5	_	_	V	IOH ≥ -7.5 mA, VDD = 3.3V (Note 1)	
			2.0	_	_		IOH ≥ -6.8 mA, VDD = 3.3V (Note 1)	
			3.0	_	_		IOH ≥ -3 mA, VDD = 3.3V (Note 1)	

# TABLE 31-8: DC CHARACTERISTICS: I/O PIN OUTPUT SPECIFICATIONS

**Note 1:** Parameters are characterized, but not tested.

2: Includes all I/O pins that are not 8x Sink Driver pins (see below).

Includes the following pins:
 For devices with less than 64 pins: RA3, RA4, RA9, RB<15:7> and RC3
 For 64-pin devices: RA4, RA9, RB<15:7>, RC3 and RC15