



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

Product Status	Active
Core Processor	dsPIC
Core Size	16-Bit
Speed	60 MIPs
Connectivity	I²C, 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-VFTLA Exposed Pad
Supplier Device Package	44-VTLA (6x6)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/dspic33ep512mc204-h-tl

Email: info@E-XFL.COM

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

REGISTER 3-1: SR: CPU STATUS REGISTER (CONTINUED)

bit 7-5	IPL<2:0>: CPU Interrupt Priority Level Status bits ^(2,3) 111 = CPU Interrupt Priority Level is 7 (15); user interrupts are disabled 110 = CPU Interrupt Priority Level is 6 (14) 101 = CPU Interrupt Priority Level is 5 (13) 100 = CPU Interrupt Priority Level is 4 (12) 011 = CPU Interrupt Priority Level is 3 (11) 010 = CPU Interrupt Priority Level is 2 (10) 001 = CPU Interrupt Priority Level is 1 (9) 000 = CPU Interrupt Priority Level is 0 (8)
bit 4	RA: REPEAT Loop Active bit 1 = REPEAT loop in progress 0 = REPEAT loop not in progress
bit 3	N: MCU ALU Negative bit 1 = Result was negative 0 = Result was non-negative (zero or positive)
bit 2	 OV: MCU ALU Overflow bit This bit is used for signed arithmetic (2's complement). It indicates an overflow of the magnitude that causes the sign bit to change state. 1 = Overflow occurred for signed arithmetic (in this arithmetic operation) 0 = No overflow occurred
bit 1	 Z: MCU ALU Zero bit 1 = An operation that affects the Z bit has set it at some time in the past 0 = The most recent operation that affects the Z bit has cleared it (i.e., a non-zero result)
bit 0	C: MCU ALU Carry/Borrow bit 1 = A carry-out from the Most Significant bit of the result occurred 0 = No carry-out from the Most Significant bit of the result occurred
Note 1: 2:	This bit is available on dsPIC33EPXXXMC20X/50X and dsPIC33EPXXXGP50X devices only. The IPL<2:0> bits are concatenated with the IPL<3> bit (CORCON<3>) to form the CPU Interrupt Priority

- Level. The value in parentheses indicates the IPL, if IPL<3> = 1. User interrupts are disabled when IPL<3> = 1.
 3: The IPL<2:0> Status bits are read-only when the NSTDIS bit (INTCON1<15>) = 1.
- 4: A data write to the SR register can modify the SA and SB bits by either a data write to SA and SB or by clearing the SAB bit. To avoid a possible SA or SB bit write race condition, the SA and SB bits should not be modified using bit operations.

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

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
TMR1	0100								Timer1	Register								xxxx
PR1	0102								Period F	Register 1								FFFF
T1CON	0104	TON	_	TSIDL	_	_	_	_	_	_	TGATE	TCKP	S<1:0>	_	TSYNC	TCS	—	0000
TMR2	0106								Timer2	Register								xxxx
TMR3HLD	0108						Time	er3 Holding	Register (fo	r 32-bit time	r operations	only)						xxxx
TMR3	010A		Timer3 Register x													xxxx		
PR2	010C		Period Register 2														FFFF	
PR3	010E		Period Register 2 F Period Register 3 F														FFFF	
T2CON	0110	TON	_	TSIDL	_	_	_	_	_	_	TGATE	TCKP	S<1:0>	T32	_	TCS	—	0000
T3CON	0112	TON	_	TSIDL	_	_	_	_	_	_	TGATE	TCKP	S<1:0>	_	_	TCS	—	0000
TMR4	0114								Timer4	Register								xxxx
TMR5HLD	0116						Т	imer5 Holdii	ng Register	(for 32-bit o	perations on	ly)						xxxx
TMR5	0118								Timer5	Register								xxxx
PR4	011A								Period F	Register 4								FFFF
PR5	011C								Period F	Register 5								FFFF
T4CON	011E	TON	_	TSIDL	_	—	—	_	_	_	TGATE	TCKP	S<1:0>	T32	—	TCS	—	0000
T5CON	0120	TON		TSIDL	—	—	—	_	_	_	TGATE	TCKP	S<1:0>	—	_	TCS	—	0000

TABLE 4-8: TIMER1 THROUGH TIMER5 REGISTER MAP

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

TABLE 4	4-9:	INPU	PUT CAPTURE 1 THROUGH INPUT CAPTURE 4 REGISTER MAP15Bit 13Bit 12Bit 11Bit 10Bit 9Bit 8Bit 7Bit 6Bit 5Bit 4Bit 3Bit 2Bit 1Bit 1Bit 1ICSIDL $ CTSEL<2:0>$ $ CI<1:0>$ $ COV $ $ CBNE $ $ CM<2:0>$ $ CI<1:0>$ $ COV $ $ CBNE $ $ CM<2:0>$ $ CI<2:0>$ $ CTSEL<2:0>$ $ CI<2:0>$ $ CV $ $ COV $ $ CBNE $ $ CM<2:0>$ Input Capture 1 Buffer RegisterInput Capture 1 Timer $ C <1:0>$ $ COV $ $ CBNE $ $ CM<2:0>$ $ C <1:0>$ $ COV $ $ CBNE $ $ CM<2:0>$ $ C <1:0>$ $ COV $ $ CBNE $ $ CM<2:0>$ $ CI <1:0>$ $ COV $ $ CBNE $ $ CM<2:0>$ $ CI <1:0>$ $ COV $ $ CBNE $ $ CM<2:0>$ $ CI <1:0>$ $ COV $ $ CBNE $ $ CM<2:0>$ $ CI <1:0>$ $ COV $ $ CBNE CM<2:0> CI <1:0> COV CBN CM<2:0>$															
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 Capture 1 Timer — — ICSIDL ICTSEL<2:0> — — ICI<1:0> ICOV ICBNE ICM<2:0>															0000
IC2CON1	0148	_	- - ICSIDL ICTSEL<2:0> - - - ICI<1:0> ICOV ICBNE ICM<2:0> - - - - - - - ICI<1:0> ICOV ICBNE ICM<2:0>															0000
IC2CON2	014A	_	- - - - - IC32 ICTRIG TRIGSTAT - SYNCSEL<4:0>															000D
IC2BUF	014C		IC32 ICTRIG TRIGSTAT SYNCSEL<4:0>															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	_		S	/NCSEL<4	:0>		000D
IC4BUF	015C		•	•	•	•	•	Inp	ut Capture	4 Buffer Re	gister	•	•					xxxx
IC4TMR	015E								Input Cap	ture 4 Time	r							0000

dsPIC33EPXXXGP50X, dsPIC33EPXXXMC20X/50X AND PIC24EPXXXGP/MC20X

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

TADLL 4-2		LUANT	IL GIGI				ICINE	1<02) -	· • • • • .	I I OK US			IC/GFJ					
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
C1CTRL1	0400	_	—	CSIDL	ABAT	CANCKS	R	EQOP<2:0)>	OPN	NODE<2:0	>	_	CANCAP	_	_	WIN	0480
C1CTRL2	0402	_	—	_	—	—	—	_	_	_	—	_		D	NCNT<4:0	>		0000
C1VEC	0404	_	_	_		F	ILHIT<4:0>			_				ICODE<6:0	>			0040
C1FCTRL	0406	[DMABS<2:0	>	—	—		—	_	—	—			FSA<4:0>			0000	
C1FIFO	0408	_	_			FBP<5:0>					_			FNRB	<5:0>			0000
C1INTF	040A	_	_	ТХВО	TXBP	RXBP	TXWAR	RXWAR	EWARN	IVRIF	WAKIF	ERRIF	_	FIFOIF	RBOVIF	RBIF	TBIF	0000
C1INTE	040C	_	_	_	—	_	_	_	_	IVRIE	WAKIE	ERRIE	_	FIFOIE	RBOVIE	RBIE	TBIE	0000
C1EC	040E				TERRCN	T<7:0>							RERRCM	NT<7:0>				0000
C1CFG1	0410	_	_	_	—	_	_	_	_	SJW<	1:0>			BRP	<5:0>			0000
C1CFG2	0412	_	WAKFIL	_	—	_	SI	EG2PH<2:(0>	SEG2PHTS	SAM	S	EG1PH<2	::0>	P	RSEG<2:0	>	0000
C1FEN1	0414	FLTEN15	FLTEN14	FLTEN13	FLTEN12	FLTEN11	FLTEN10	FLTEN9	FLTEN8	FLTEN7	FLTEN6	FLTEN5	FLTEN4	FLTEN3	FLTEN2	FLTEN1	FLTEN0	FFFF
C1FMSKSEL1	0418	F7MS	K<1:0>	F6MS	K<1:0>	F5MS	K<1:0>	F4MS	K<1:0>	F3MSK	<1:0>	F2MS	K<1:0>	F1MSł	<<1:0>	F0MS	<<1:0>	0000
C1FMSKSEL2	041A	F15MS	SK<1:0>	F14MS	K<1:0>	F13MS	SK<1:0>	F12MS	SK<1:0>	F11MSK	<1:0>	F10MS	K<1:0>	F9MSł	<<1:0>	F8MS	< <1:0>	0000

TABLE 4-21: ECAN1 REGISTER MAP WHEN WIN (C1CTRL1<0>) = 0 OR 1 FOR dsPIC33EPXXXMC/GP50X DEVICES ONLY

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

TABLE 4-22: ECAN1 REGISTER MAP WHEN WIN (C1CTRL1<0>) = 0 FOR dsPIC33EPXXXMC/GP50X 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
	0400- 041E							S	ee definition	when WIN	= x							
C1RXFUL1	0420	RXFUL15	RXFUL14	RXFUL13	RXFUL12	RXFUL11	RXFUL10	RXFUL9	RXFUL8	RXFUL7	RXFUL6	RXFUL5	RXFUL4	RXFUL3	RXFUL2	RXFUL1	RXFUL0	0000
C1RXFUL2	0422	RXFUL31	RXFUL30	RXFUL29	RXFUL28	RXFUL27	RXFUL27 RXFUL26 RXFUL25 RXFUL24 RXFUL23 RXFUL22 RXFUL21 RXFUL20 RXFUL19 RXFUL18 RXFUL17 RXFUL16 000 RXOVF11 RXOVF10 RXOVF3 RXOVF3 RXOVF1 R								0000			
C1RXOVF1	0428	RXOVF15	RXOVF14	RXOVF13	RXOVF12	RXOVF11	RXOVF10	RXOVF9	RXOVF8	RXOVF7	RXOVF6	RXOVF5	RXOVF4	RXOVF3	RXOVF2	RXOVF1	RXOVF0	0000
C1RXOVF2	042A	RXOVF31	RXOVF30	RXOVF29	RXOVF28	RXOVF27	RXOVF26	RXOVF25	RXOVF24	RXOVF23	RXOVF22	RXOVF21	RXOVF20	RXOVF19	RXOVF18	RXOVF17	RXOVF16	0000
C1TR01CON	0430	TXEN1	TXABT1	TXLARB1	TXERR1	TXREQ1	RTREN1	TX1PF	RI<1:0>	TXEN0	TXABAT0	TXLARB0	TXERR0	TXREQ0	RTREN0	TX0PF	RI<1:0>	0000
C1TR23CON	0432	TXEN3	TXABT3	TXLARB3	TXERR3	TXREQ3	RTREN3	TX3PF	RI<1:0>	TXEN2	TXABAT2	TXLARB2	TXERR2	TXREQ2	RTREN2	TX2PF	RI<1:0>	0000
C1TR45CON	0434	TXEN5	TXABT5	TXLARB5	TXERR5	TXREQ5	RTREN5	TX5PF	RI<1:0>	TXEN4	TXABAT4	TXLARB4	TXERR4	TXREQ4	RTREN4	TX4PF	RI<1:0>	0000
C1TR67CON	0436	TXEN7	TXABT7	TXLARB7	TXERR7	TXREQ7	TXREQ7 RTREN7 TX7PRI<1:0> TXEN6 TXABAT6 TXLARB6 TXER6 TXREQ6 RTREN6 TX6PRI<1:0>									xxxx		
C1RXD	0440							E	CAN1 Rece	eive Data Wo	ord							xxxx
C1TXD	0442							E	CAN1 Trans	smit Data W	ord							xxxx

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

TABLE 4-24: CRC 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
CRCCON1	0640	CRCEN	—	CSIDL		V	WORD<4:()>		CRCFUL	CRCMPT	CRCISEL	CRCGO	LENDIAN	—	-	—	0000
CRCCON2	0642	_	_	_		D	WIDTH<4:0)>		_	_	_		F	PLEN<4:0>			0000
CRCXORL	0644		X<15:1> —														0000	
CRCXORH	0646		X<15:1> — X<31:16>															0000
CRCDATL	0648								CRC Data	Input Low V	Vord							0000
CRCDATH	064A								CRC Data	Input High \	Nord							0000
CRCWDATL	064C								CRC Re	sult Low Wo	ord							0000
CRCWDATH	064E								CRC Re	sult High Wo	ord							0000

Legend: — = unimplemented, read as '0'. Shaded bits are not used in the operation of the programmable CRC module.

TABLE 4-25: PERIPHERAL PIN SELECT OUTPUT REGISTER MAP FOR dsPIC33EPXXXGP/MC202/502 AND PIC24EPXXXGP/MC202 DEVICES ONLY 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
RPOR0	0680	_	—			RP35F	R<5:0>			—	—			RP20F	R<5:0>			0000
RPOR1	0682	_	_			RP37F	२<5:0>			—	_			RP36F	२<5:0>			0000
RPOR2	0684	_	_			RP39F	२<5:0>			—	_			RP38F	२<5:0>			0000
RPOR3	0686	_	_			RP41F	२<5:0>			—	_			RP40F	२<5:0>			0000
RPOR4	0688	_	—			RP43F	R<5:0>			_	_			RP42F	२<5:0>			0000

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

TABLE 4-26: PERIPHERAL PIN SELECT OUTPUT REGISTER MAP FOR dsPIC33EPXXXGP/MC203/503 AND PIC24EPXXXGP/MC203 DEVICES ONLY 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
RPOR0	0680	—	_			RP35	R<5:0>			_	_			RP20	R<5:0>			0000
RPOR1	0682	_	_			RP37	२<5:0>			_	_			RP36	२<5:0>			0000
RPOR2	0684	_	_			RP39	२<5:0>			_	_			RP38	२<5:0>			0000
RPOR3	0686	_	_			RP41	२<5:0>			_	_			RP40	२<5:0>			0000
RPOR4	0688	_	_			RP43	२<5:0>			_	_			RP42	२<5:0>			0000
RPOR5	068A	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	0000
RPOR6	068C			-	—	_		—			_			RP56	R<5:0>			0000

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

TABLE 4-34: NVM 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	
NVMCON	0728	WR	WREN	WRERR	NVMSIDL	_	_	_	_	NVMOP<3:0>									
NVMADRL	072A								NVMAD	R<15:0>								0000	
NVMADRH	072C		_	_	_	_		_					NVMADF	२<23:16>				0000	
NVMKEY	072E	_	_		_	_	_	_	_				NVMKE	Y<7:0>				0000	

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

TABLE 4-35: SYSTEM CONTROL 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
RCON	0740	TRAPR	IOPUWR	_		VREGSF	_	СМ	VREGS	EXTR	SWR	SWDTEN	WDTO	SLEEP	IDLE	BOR	POR	Note 1
OSCCON	0742	_	(COSC<2:0>				NOSC<2:0>		CLKLOCK	IOLOCK	LOCK	_	CF	_	_	OSWEN	Note 2
CLKDIV	0744	ROI	[DOZE<2:0>		DOZEN	F	RCDIV<2:0	>	PLLPOS	T<1:0>	—		F	LLPRE<	4:0>		0030
PLLFBD	0746	_	_	_			_	_				PLLDI	V<8:0>					0030
OSCTUN	0748	_	_	_	_	_	_	_	_	_	_			TUN	<5:0>			0000

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

Note 1: RCON register Reset values are dependent on the type of Reset.

2: OSCCON register Reset values are dependent on the Configuration Fuses.

TABLE 4-36: REFERENCE CLOCK 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
REFOCON	074E	ROON	_	ROSSLP	ROSEL		RODI	V<3:0>		_	—		-	-	_		—	0000

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

4.5 Instruction Addressing Modes

The addressing modes shown in Table 4-63 form the basis of the addressing modes optimized to support the specific features of individual instructions. The addressing modes provided in the MAC class of instructions differ from those in the other instruction types.

4.5.1 FILE REGISTER INSTRUCTIONS

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

4.5.2 MCU INSTRUCTIONS

The three-operand MCU instructions are of the form:

Operand 3 = Operand 1 <function> Operand 2

where Operand 1 is always a working register (that is, the addressing mode can only be Register Direct), which is referred to as Wb. Operand 2 can be a W register fetched from data memory or a 5-bit literal. The result location can either be a W register or a data memory location. The following addressing modes are supported by MCU instructions:

- Register Direct
- · Register Indirect
- · Register Indirect Post-Modified
- Register Indirect Pre-Modified
- 5-Bit or 10-Bit Literal
- Note: Not all instructions support all the addressing modes given above. Individual instructions can support different subsets of these addressing modes.

TABLE 4-63: FUNDAMENTAL ADDRESSING MODES SUPPORTED

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

In addition, DMA transfers can be triggered by timers as well as external interrupts. Each DMA channel is unidirectional. Two DMA channels must be allocated to read and write to a peripheral. If more than one channel receives a request to transfer data, a simple fixed priority scheme based on channel number, dictates which channel completes the transfer and which channel, or channels, are left pending. Each DMA channel moves a block of data, after which, it generates an interrupt to the CPU to indicate that the block is available for processing.

The DMA Controller provides these functional capabilities:

- Four DMA channels
- Register Indirect with Post-Increment Addressing mode
- Register Indirect without Post-Increment Addressing mode

- Peripheral Indirect Addressing mode (peripheral generates destination address)
- CPU interrupt after half or full block transfer complete
- Byte or word transfers
- · Fixed priority channel arbitration
- Manual (software) or automatic (peripheral DMA requests) transfer initiation
- One-Shot or Auto-Repeat Block Transfer modes
- Ping-Pong mode (automatic switch between two SRAM start addresses after each block transfer is complete)
- DMA request for each channel can be selected from any supported interrupt source
- Debug support features

The peripherals that can utilize DMA are listed in Table 8-1.

Peripheral to DMA Association	DMAxREQ Register IRQSEL<7:0> Bits	DMAxPAD Register (Values to Read from Peripheral)	DMAxPAD Register (Values to Write to Peripheral)
INT0 – External Interrupt 0	00000000	—	—
IC1 – Input Capture 1	0000001	0x0144 (IC1BUF)	—
IC2 – Input Capture 2	00000101	0x014C (IC2BUF)	_
IC3 – Input Capture 3	00100101	0x0154 (IC3BUF)	—
IC4 – Input Capture 4	00100110	0x015C (IC4BUF)	—
OC1 – Output Compare 1	00000010	_	0x0906 (OC1R) 0x0904 (OC1RS)
OC2 – Output Compare 2	00000110	_	0x0910 (OC2R) 0x090E (OC2RS)
OC3 – Output Compare 3	00011001	_	0x091A (OC3R) 0x0918 (OC3RS)
OC4 – Output Compare 4	00011010	_	0x0924 (OC4R) 0x0922 (OC4RS)
TMR2 – Timer2	00000111	—	—
TMR3 – Timer3	00001000	-	—
TMR4 – Timer4	00011011	_	—
TMR5 – Timer5	00011100	—	—
SPI1 Transfer Done	00001010	0x0248 (SPI1BUF)	0x0248 (SPI1BUF)
SPI2 Transfer Done	00100001	0x0268 (SPI2BUF)	0x0268 (SPI2BUF)
UART1RX – UART1 Receiver	00001011	0x0226 (U1RXREG)	—
UART1TX – UART1 Transmitter	00001100	—	0x0224 (U1TXREG)
UART2RX – UART2 Receiver	00011110	0x0236 (U2RXREG)	—
UART2TX – UART2 Transmitter	00011111	—	0x0234 (U2TXREG)
ECAN1 – RX Data Ready	00100010	0x0440 (C1RXD)	
ECAN1 – TX Data Request	01000110	—	0x0442 (C1TXD)
ADC1 – ADC1 Convert Done	00001101	0x0300 (ADC1BUF0)	_

TABLE 8-1: DMA CHANNEL TO PERIPHERAL ASSOCIATIONS

11.0 I/O PORTS

- 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 "I/O Ports" (DS70598) 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.

Many of the device pins are shared among the peripherals and the parallel I/O ports. All I/O input ports feature Schmitt Trigger inputs for improved noise immunity.

11.1 Parallel I/O (PIO) Ports

Generally, a parallel I/O port that shares a pin with a peripheral is subservient to the peripheral. The peripheral's output buffer data and control signals are provided to a pair of multiplexers. The multiplexers select whether the peripheral or the associated port has ownership of the output data and control signals of the I/O pin. The logic also prevents "loop through," in which a port's digital output can drive the input of a peripheral that shares the same pin. Figure 11-1 illustrates how ports are shared with other peripherals and the associated I/O pin to which they are connected.

When a peripheral is enabled and the peripheral is actively driving an associated pin, the use of the pin as a general purpose output pin is disabled. The I/O pin can be read, but the output driver for the parallel port bit is disabled. If a peripheral is enabled, but the peripheral is not actively driving a pin, that pin can be driven by a port.

All port pins have eight registers directly associated with their operation as digital I/O. The Data Direction register (TRISx) determines whether the pin is an input or an output. If the data direction bit is a '1', then the pin is an input. All port pins are defined as inputs after a Reset. Reads from the Latch register (LATx) read the latch. Writes to the Latch write the latch. Reads from the port (PORTx) read the port pins, while writes to the port pins write the latch.

Any bit and its associated data and control registers that are not valid for a particular device is disabled. This means the corresponding LATx and TRISx registers and the port pin are read as zeros.

When a pin is shared with another peripheral or function that is defined as an input only, it is nevertheless regarded as a dedicated port because there is no other competing source of outputs.





- g) The TRISx registers control only the digital I/O output buffer. Any other dedicated or remappable active "output" will automatically override the TRIS setting. The TRISx register does not control the digital logic "input" buffer. Remappable digital "inputs" do not automatically override TRIS settings, which means that the TRISx bit must be set to input for pins with only remappable input function(s) assigned
- h) All analog pins are enabled by default after any Reset and the corresponding digital input buffer on the pin has been disabled. Only the Analog Pin Select registers control the digital input buffer, *not* the TRISx register. The user must disable the analog function on a pin using the Analog Pin Select registers in order to use any "digital input(s)" on a corresponding pin, no exceptions.

11.6 I/O Ports 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, which can be accessed using this link, contains the latest updates and additional information.

Note:	In the event you are not able to access the
	product page using the link above, enter
	this URL in your browser:
	http://www.microchip.com/wwwproducts/
	Devices.aspx?dDocName=en555464

11.6.1 KEY RESOURCES

- "I/O Ports" (DS70598) in the "dsPIC33/PIC24 Family Reference Manual"
- Code Samples
- Application Notes
- Software Libraries
- Webinars
- All Related "dsPIC33/PIC24 Family Reference Manual" Sections
- Development Tools

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
TON ⁽¹⁾	—	TSIDL ⁽²⁾	—	_	—	—	—
bit 15							bit 8
U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	U-0
—	TGATE ⁽¹⁾	TCKPS1 ⁽¹⁾	TCKPS0 ⁽¹⁾	_	—	TCS ^(1,3)	—
bit 7							bit 0

REGISTER 13-2: TyCON: (TIMER3 AND TIMER5) CONTROL REGISTER

Legend:									
R = Read	able bit	W = Writable bit	U = Unimplemented bit,	read as '0'					
-n = Value	e at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown					
bit 15	TON: Tim	iery On bit ⁽¹⁾							
	1 = Starts 16-bit Timery								
	0 = Stops	16-bit Timery							
bit 14	Unimpler	nented: Read as '0'							
bit 13	TSIDL: T	mery Stop in Idle Mode bit ⁽²	2)						
	 1 = Discontinues module operation when device enters Idle mode 0 = Continues module operation in Idle mode 								
bit 12-7	Unimpler	Unimplemented: Read as '0'							
bit 6	TGATE:	TGATE: Timery Gated Time Accumulation Enable bit ⁽¹⁾							
	When TC	<u>S = 1:</u>							
	This bit is	ignored.							
	When TC	$\underline{S} = 0$:							
	1 = Gateo	time accumulation is enabled time accumulation is disabled	led						
bit 5 1		1.0	accelo Soloct hits(1)						
DIL 3-4	11 = 1.25								
	10 = 1:64								
	01 = 1:8								
	00 = 1:1								
bit 3-2	Unimpler	nented: Read as '0'							
bit 1	TCS: Tim	ery Clock Source Select bit	(1,3)						
	1 = Extern 0 = Intern	nal clock is from pin, TyCK (al clock (FP)	on the rising edge)						
bit 0	Unimpler	nented: Read as '0'							
Note 1:	When 32-bit op functions are s	peration is enabled (T2CON- set through TxCON.	<3> = 1), these bits have no ef	fect on Timery operation; all tir	ner				

2: When 32-bit timer operation is enabled (T32 = 1) in the Timerx Control register (TxCON<3>), the TSIDL bit must be cleared to operate the 32-bit timer in Idle mode.

3: The TyCK pin is not available on all timers. See the "Pin Diagrams" section for the available pins.

18.3 SPIx Control Registers

R/W-0 U-0 R/W-0 U-0 R/W-0 R/W-0 R/W-0 U-0 SPIEN SPISIDL SPIBEC<2:0> _____ bit 15 R/W-0 R/W-0 R/W-0 R/C-0, HS R/W-0 R/W-0 R-0, HS, HC R-0, HS, HC SRMPT SPIROV SRXMPT SISEL2 SISEL1 SISEL0 SPITBF SPIRBF bit 7 bit 0 Legend: C = Clearable bit HS = Hardware Settable bit HC = Hardware Clearable 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 SPIEN: SPIx Enable bit 1 = Enables the module and configures SCKx, SDOx, SDIx and \overline{SSx} as serial port pins 0 = Disables the module bit 14 Unimplemented: Read as '0' bit 13 SPISIDL: SPIx Stop in Idle Mode bit 1 = Discontinues the module operation when device enters Idle mode 0 = Continues the module operation in Idle mode bit 12-11 Unimplemented: Read as '0' bit 10-8 SPIBEC<2:0>: SPIx Buffer Element Count bits (valid in Enhanced Buffer mode) Master mode: Number of SPIx transfers that are pending. Slave mode: Number of SPIx transfers that are unread. SRMPT: SPIx Shift Register (SPIxSR) Empty bit (valid in Enhanced Buffer mode) bit 7 1 = SPIx Shift register is empty and Ready-To-Send or receive the data 0 = SPIx Shift register is not empty bit 6 SPIROV: SPIx Receive Overflow Flag bit 1 = A new byte/word is completely received and discarded; the user application has not read the previous data in the SPIxBUF register

REGISTER 18-1: SPIxSTAT: SPIx STATUS AND CONTROL REGISTER

0 = No overflow has occurred SRXMPT: SPIx Receive FIFO Empty bit (valid in Enhanced Buffer mode) bit 5 1 = RX FIFO is empty 0 = RX FIFO is not empty bit 4-2 SISEL<2:0>: SPIx Buffer Interrupt Mode bits (valid in Enhanced Buffer mode) 111 = Interrupt when the SPIx transmit buffer is full (SPITBF bit is set) 110 = Interrupt when last bit is shifted into SPIxSR and as a result, the TX FIFO is empty 101 = Interrupt when the last bit is shifted out of SPIxSR and the transmit is complete 100 = Interrupt when one data is shifted into the SPIxSR and as a result, the TX FIFO has one open memory location

- 011 = Interrupt when the SPIx receive buffer is full (SPIRBF bit is set)
- 010 = Interrupt when the SPIx receive buffer is 3/4 or more full
- 001 = Interrupt when data is available in the receive buffer (SRMPT bit is set)
- 000 = Interrupt when the last data in the receive buffer is read and as a result, the buffer is empty (SRXMPT bit is set)

bit 8

19.1 I²C 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, which can be accessed using this link, contains the latest updates and additional information.

Note:	In the event you are not able to access the
	product page using the link above, enter
	this URL in your browser:
	http://www.microchip.com/wwwproducts/
	Devices.aspx?dDocName=en555464

19.1.1 KEY RESOURCES

- "Inter-Integrated Circuit (I²C)" (DS70330) in the "dsPIC33/PIC24 Family Reference Manual"
- Code Samples
- Application Notes
- · Software Libraries
- Webinars
- All Related "dsPIC33/PIC24 Family Reference Manual" Sections
- Development Tools

NOTES:

dsPIC33EPXXXGP50X, dsPIC33EPXXXMC20X/50X AND PIC24EPXXXGP/MC20X

R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADRC	—	—	SAMC4 ⁽¹⁾	SAMC3 ⁽¹⁾	SAMC2 ⁽¹⁾	SAMC1 ⁽¹⁾	SAMC0 ⁽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
ADCS7(2	²⁾ ADCS6 ⁽²⁾	ADCS5 ⁽²⁾	ADCS4 ⁽²⁾	ADCS3 ⁽²⁾	ADCS2 ⁽²⁾	ADCS1 ⁽²⁾	ADCS0 ⁽²⁾
bit 7							bit 0
Legend:							
R = Reada		vv = vvritable t	DIT		nented bit, read		
-n = value	at POR	"1" = Bit is set		0° = Bit is cle	ared	x = Bit is unkr	nown
bit 15	5 ADRC: ADC1 Conversion Clock Source bit 1 = ADC internal RC clock 0 = Clock derived from system clock						
bit 14-13	Unimplement	ted: Read as '0	3				
bit 12-8	SAMC<4:0>:	Auto-Sample T	ime bits ⁽¹⁾				
	11111 = 31 T. • • • • • •	AD					
hit 7 0	00000 = 0 IA		ion Clock Colo	at hita(2)			
Dit 7-0	ADC3<7.051 11111111 = -	TP • (ADCS<7: TP • (ADCS<7: TP • (ADCS<7: TP • (ADCS<7: TP • (ADCS<7:	0> + 1) = TP • 0> + 1) = TP • 0> + 1) = TP • 0> + 1) = TP •	256 = TAD 3 = TAD 2 = TAD 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

24.0 PERIPHERAL TRIGGER GENERATOR (PTG) MODULE

- 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 "Peripheral Trigger Generator (PTG)" (DS70669) 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.

24.1 Module Introduction

The Peripheral Trigger Generator (PTG) provides a means to schedule complex high-speed peripheral operations that would be difficult to achieve using software. The PTG module uses 8-bit commands, called "Steps", that the user writes to the PTG Queue registers (PTGQUE0-PTGQUE7), which perform operations, such as wait for input signal, generate output trigger and wait for timer.

The PTG module has the following major features:

- Multiple clock sources
- Two 16-bit general purpose timers
- Two 16-bit general limit counters
- Configurable for rising or falling edge triggering
- Generates processor interrupts to include:
 - Four configurable processor interrupts
 - Interrupt on a Step event in Single-Step modeInterrupt on a PTG Watchdog Timer time-out
- Able to receive trigger signals from these peripherals:
 - ADC
 - PWM
 - Output Compare
 - Input Capture
 - Op Amp/Comparator
 - INT2
- Able to trigger or synchronize to these peripherals:
 - Watchdog Timer
 - Output Compare
 - Input Capture
 - ADC
 - PWM
- Op Amp/Comparator

REGISTER 24-10: PTGADJ: PTG ADJUST REGISTER⁽¹⁾

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
			PTGA	DJ<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
			PTGA	DJ<7:0>			
bit 7							bit 0
Legend:							
R = Readable I	bit	W = Writable I	bit	U = Unimpler	nented bit, rea	d as '0'	
-n = Value at P	OR	'1' = Bit is set		'0' = Bit is cle	ared	x = Bit is unkr	nown

bit 15-0 **PTGADJ<15:0>:** PTG Adjust Register bits This register holds user-supplied data to be added to the PTGTxLIM, PTGCxLIM, PTGSDLIM or PTGL0 registers with the PTGADD command.

REGISTER 24-11: PTGL0: PTG LITERAL 0 REGISTER⁽¹⁾

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
			PTGL0	<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
			PTGL	0<7:0>			
bit 7							bit 0

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 15-0 PTGL0<15:0>: PTG Literal 0 Register bits

This register holds the 16-bit value to be written to the AD1CHS0 register with the ${\tt PTGCTRL}$ Step command.

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

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

dsPIC33EPXXXGP50X, dsPIC33EPXXXMC20X/50X AND PIC24EPXXXGP/MC20X

REGISTER 2	5-3: CM4C	ON: COMPA	RATOR 4 CO	ONTROL RE	GISTER		
R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0
CON	COE	CPOL	_			CEVT	COUT
bit 15							bit 8
R/W-0	R/W-0	U-0	R/W-0	U-0	U-0	R/W-0	R/W-0
EVPOL1	EVPOL0		CREF ⁽¹⁾			CCH1 ⁽¹⁾	CCH0 ⁽¹⁾
bit 7	•		1			-	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	CON: Comparator Enable bit						
	1 = Comparator is enabled						
	0 = Comparator is disabled						
bit 14	COE: Comparator Output Enable bit						
	1 = Comparator output is present on the CxOUT pin 0 = Comparator output is internal only						
bit 12	CPOL - Comparator Output Is Internal Only						
DIL 13	1 = Comparator output is inverted						
	0 = Comparator output is not inverted						
bit 12-10	Unimplemented: Read as '0'						
bit 9	CEVT: Comparator Event bit						
	1 = Comparator event according to EVPOL<1:0> settings occurred; disables future triggers and						
	interrupts until the bit is cleared						
hit 0	COLLE: Comparator Output bit						
DILO	When $CPOI = 0$ (non-inverted polarity):						
	1 = VIN + > VIN-						
	$0 = VIN + \langle VIN - VIN $						
	When CPOL = 1 (inverted polarity):						
	1 = VIN + < VIN						
bit 7.6		 Triagor/Eyopt 		arity Soloct bits	-		
bit 7-0	11 = Trigger/event/interrupt generated on any change of the comparator output (while CEV/T = 0)						
	 10 = Trigger/event/interrupt generated only on high-to-low transition of the polarity selected comparator output (while CEVT = 0) 						
	If CPOL = 1 (inverted polarity): Low-to-high transition of the comparator output.						
	If CPOL = 0 (non-inverted polarity): High-to-low transition of the comparator output.						
	01 = Trigger/event/interrupt generated only on low-to-high transition of the polarity selected comparator output (while CEVT = 0)						
	If CPOL = 1 (inverted polarity): High-to-low transition of the comparator output.						
	If CPOL = 0 (non-inverted polarity): Low-to-high transition of the comparator output.						
	00 = Trigger/event/interrupt generation is disabled						
				1	() (() -)		

Note 1: Inputs that are selected and not available will be tied to Vss. See the "Pin Diagrams" section for available inputs for each package.

NOTES: