



Welcome to [E-XFL.COM](https://www.e-xfl.com)

### What is "[Embedded - Microcontrollers](#)"?

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

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

#### Details

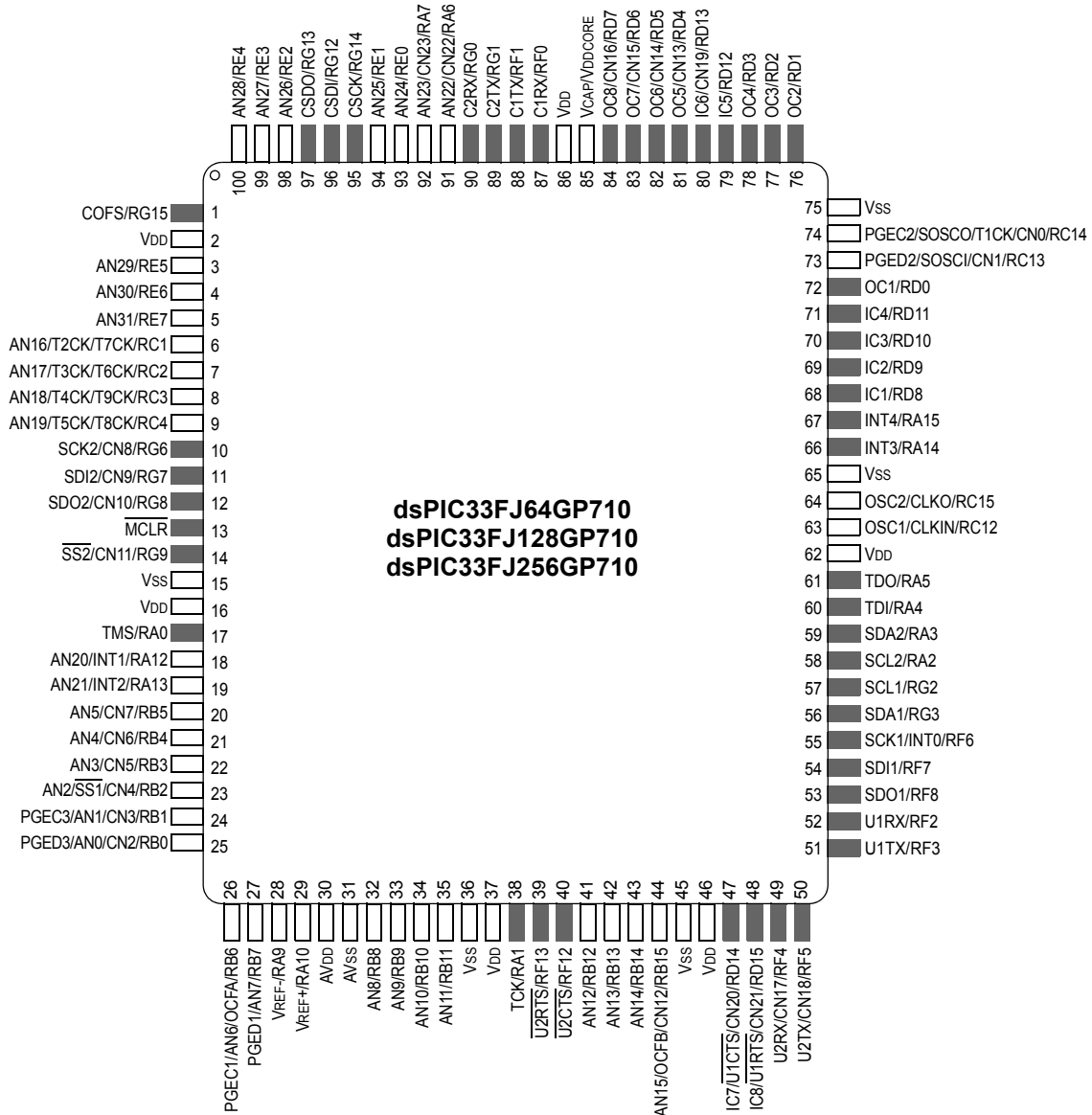
Product Status	Active
Core Processor	dsPIC
Core Size	16-Bit
Speed	40 MIPS
Connectivity	CANbus, I <sup>2</sup> C, IrDA, LINbus, SPI, UART/USART
Peripherals	AC'97, Brown-out Detect/Reset, DMA, I <sup>2</sup> S, POR, PWM, WDT
Number of I/O	85
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	16K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 32x10b/12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	100-TQFP
Supplier Device Package	100-TQFP (14x14)
Purchase URL	<a href="https://www.e-xfl.com/product-detail/microchip-technology/dspic33fj128gp710-i-pf">https://www.e-xfl.com/product-detail/microchip-technology/dspic33fj128gp710-i-pf</a>

# dsPIC33FJXXXGPX06/X08/X10

## Pin Diagrams (Continued)

### 100-Pin TQFP

■ = Pins are up to 5V tolerant



## 3.6.1 MULTIPLIER

The 17-bit x 17-bit multiplier is capable of signed or unsigned operation and can multiplex its output using a scaler to support either 1.31 fractional (Q31) or 32-bit integer results. Unsigned operands are zero-extended into the 17th bit of the multiplier input value. Signed operands are sign-extended into the 17th bit of the multiplier input value. The output of the 17-bit x 17-bit multiplier/scaler is a 33-bit value which is sign-extended to 40 bits. Integer data is inherently represented as a signed two's complement value, where the Most Significant bit (MSb) is defined as a sign bit. Generally speaking, the range of an N-bit two's complement integer is  $-2^{N-1}$  to  $2^{N-1} - 1$ . For a 16-bit integer, the data range is -32768 (0x8000) to 32767 (0x7FFF) including 0. For a 32-bit integer, the data range is -2,147,483,648 (0x8000 0000) to 2,147,483,647 (0x7FFF FFFF).

When the multiplier is configured for fractional multiplication, the data is represented as a two's complement fraction, where the MSb is defined as a sign bit and the radix point is implied to lie just after the sign bit (QX format). The range of an N-bit two's complement fraction with this implied radix point is -1.0 to  $(1 - 2^{1-N})$ . For a 16-bit fraction, the Q15 data range is -1.0 (0x8000) to 0.999969482 (0x7FFF) including 0 and has a precision of  $3.01518 \times 10^{-5}$ . In Fractional mode, the 16 x 16 multiply operation generates a 1.31 product which has a precision of  $4.65661 \times 10^{-10}$ .

The same multiplier is used to support the MCU multiply instructions which include integer 16-bit signed, unsigned and mixed sign multiplies.

The MUL instruction may be directed to use byte or word sized operands. Byte operands will direct a 16-bit result, and word operands will direct a 32-bit result to the specified register(s) in the W array.

## 3.6.2 DATA ACCUMULATORS AND ADDER/SUBTRACTER

The data accumulator consists of a 40-bit adder/subtractor with automatic sign extension logic. It can select one of two accumulators (A or B) as its pre-accumulation source and post-accumulation destination. For the ADD and LAC instructions, the data to be accumulated or loaded can be optionally scaled via the barrel shifter prior to accumulation.

## 3.6.2.1 Adder/Subtractor, Overflow and Saturation

The adder/subtractor is a 40-bit adder with an optional zero input into one side, and either true, or complement data into the other input. In the case of addition, the Carry/Borrow input is active-high and the other input is true data (not complemented), whereas in the case of subtraction, the Carry/Borrow input is active-low and the other input is complemented. The adder/subtractor generates Overflow Status bits, SA/SB and OA/OB, which are latched and reflected in the STATUS register:

- Overflow from bit 39: this is a catastrophic overflow in which the sign of the accumulator is destroyed.
- Overflow into guard bits 32 through 39: this is a recoverable overflow. This bit is set whenever all the guard bits are not identical to each other.

The adder has an additional saturation block which controls accumulator data saturation, if selected. It uses the result of the adder, the Overflow Status bits described above and the SAT<A:B> (CORCON<7:6>) and ACCSAT (CORCON<4>) mode control bits to determine when and to what value to saturate.

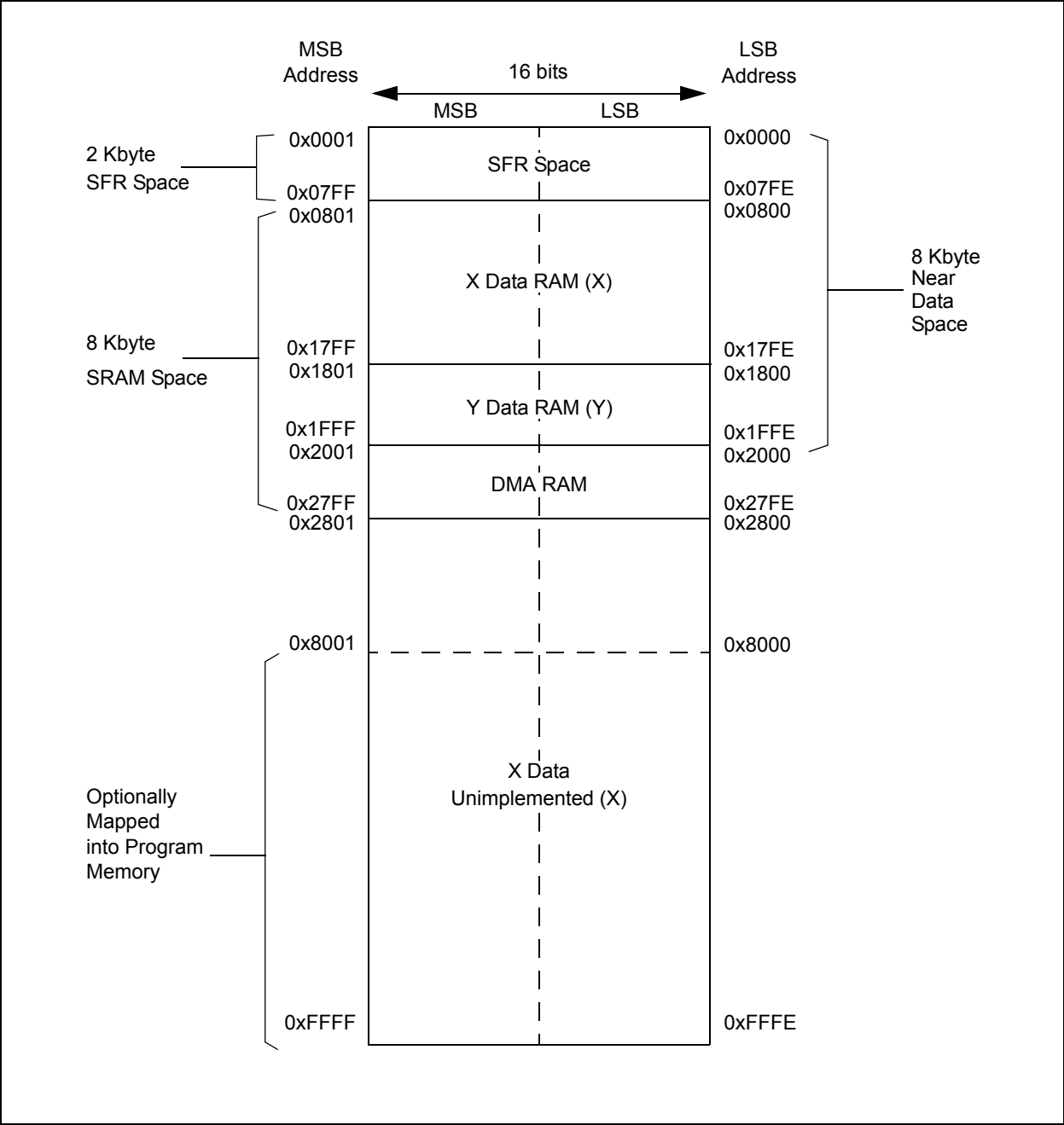
Six STATUS register bits have been provided to support saturation and overflow; they are:

1. OA:  
AccA overflowed into guard bits
2. OB:  
AccB overflowed into guard bits
3. SA:  
AccA saturated (bit 31 overflow and saturation)  
or  
AccA overflowed into guard bits and saturated (bit 39 overflow and saturation)
4. SB:  
AccB saturated (bit 31 overflow and saturation)  
or  
AccB overflowed into guard bits and saturated (bit 39 overflow and saturation)
5. OAB:  
Logical OR of OA and OB
6. SAB:  
Logical OR of SA and SB

The OA and OB bits are modified each time data passes through the adder/subtractor. When set, they indicate that the most recent operation has overflowed into the accumulator guard bits (bits 32 through 39). The OA and OB bits can also optionally generate an arithmetic warning trap when set and the corresponding Overflow Trap Flag Enable bits (OVATE, OVBT) in the INTCON1 register (refer to **Section 7.0 "Interrupt Controller"**) are set. This allows the user to take immediate action, for example, to correct system gain.

# dsPIC33FJXXXGPX06/X08/X10

FIGURE 4-3: DATA MEMORY MAP FOR dsPIC33FJXXXGPX06/X08/X10 DEVICES WITH 8 KBS RAM



**TABLE 4-6: TIMER REGISTER MAP**

SFR Name	SFR 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 Register 1																FFFF
T1CON	0104	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>	—	TSYNC	TCS	—	0000	
TMR2	0106	Timer2 Register																xxxx
TMR3HLD	0108	Timer3 Holding Register (for 32-bit timer operations only)																xxxx
TMR3	010A	Timer3 Register																xxxx
PR2	010C	Period Register 2																FFFF
PR3	010E	Period Register 3																FFFF
T2CON	0110	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>	T32	—	TCS	—	0000	
T3CON	0112	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>	—	—	TCS	—	0000	
TMR4	0114	Timer4 Register																xxxx
TMR5HLD	0116	Timer5 Holding Register (for 32-bit operations only)																xxxx
TMR5	0118	Timer5 Register																xxxx
PR4	011A	Period Register 4																FFFF
PR5	011C	Period Register 5																FFFF
T4CON	011E	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>	T32	—	TCS	—	0000	
T5CON	0120	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>	—	—	TCS	—	0000	
TMR6	0122	Timer6 Register																xxxx
TMR7HLD	0124	Timer7 Holding Register (for 32-bit operations only)																xxxx
TMR7	0126	Timer7 Register																xxxx
PR6	0128	Period Register 6																FFFF
PR7	012A	Period Register 7																FFFF
T6CON	012C	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>	T32	—	TCS	—	0000	
T7CON	012E	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>	—	—	TCS	—	0000	
TMR8	0130	Timer8 Register																xxxx
TMR9HLD	0132	Timer9 Holding Register (for 32-bit operations only)																xxxx
TMR9	0134	Timer9 Register																xxxx
PR8	0136	Period Register 8																FFFF
PR9	0138	Period Register 9																FFFF
T8CON	013A	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>	T32	—	TCS	—	0000	
T9CON	013C	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>	—	—	TCS	—	0000	

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

TABLE 4-17: DMA 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	
DMA0CON	0380	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—	MODE<1:0>		0000	
DMA0REQ	0382	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>								0000
DMA0STA	0384	STA<15:0>																	0000
DMA0STB	0386	STB<15:0>																	0000
DMA0PAD	0388	PAD<15:0>																	0000
DMA0CNT	038A	—	—	—	—	—	—	CNT<9:0>											0000
DMA1CON	038C	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—	MODE<1:0>		0000	
DMA1REQ	038E	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>								0000
DMA1STA	0390	STA<15:0>																	0000
DMA1STB	0392	STB<15:0>																	0000
DMA1PAD	0394	PAD<15:0>																	0000
DMA1CNT	0396	—	—	—	—	—	—	CNT<9:0>											0000
DMA2CON	0398	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—	MODE<1:0>		0000	
DMA2REQ	039A	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>								0000
DMA2STA	039C	STA<15:0>																	0000
DMA2STB	039E	STB<15:0>																	0000
DMA2PAD	03A0	PAD<15:0>																	0000
DMA2CNT	03A2	—	—	—	—	—	—	CNT<9:0>											0000
DMA3CON	03A4	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—	MODE<1:0>		0000	
DMA3REQ	03A6	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>								0000
DMA3STA	03A8	STA<15:0>																	0000
DMA3STB	03AA	STB<15:0>																	0000
DMA3PAD	03AC	PAD<15:0>																	0000
DMA3CNT	03AE	—	—	—	—	—	—	CNT<9:0>											0000
DMA4CON	03B0	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—	MODE<1:0>		0000	
DMA4REQ	03B2	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>								0000
DMA4STA	03B4	STA<15:0>																	0000
DMA4STB	03B6	STB<15:0>																	0000
DMA4PAD	03B8	PAD<15:0>																	0000
DMA4CNT	03BA	—	—	—	—	—	—	CNT<9:0>											0000
DMA5CON	03BC	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—	MODE<1:0>		0000	
DMA5REQ	03BE	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>								0000
DMA5STA	03C0	STA<15:0>																	0000
DMA5STB	03C2	STB<15:0>																	0000
DMA5PAD	03C4	PAD<15:0>																	0000

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

**TABLE 4-20: ECAN1 REGISTER MAP WHEN C1CTRL1.WIN = 1 FOR dsPIC33FJXXXGP506/510/706/708/710 DEVICES ONLY (CONTINUED)**

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets	
C1RXF11SID	046C	SID<10:3>								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx	
C1RXF11EID	046E	EID<15:8>								EID<7:0>									xxxx
C1RXF12SID	0470	SID<10:3>								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx	
C1RXF12EID	0472	EID<15:8>								EID<7:0>									xxxx
C1RXF13SID	0474	SID<10:3>								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx	
C1RXF13EID	0476	EID<15:8>								EID<7:0>									xxxx
C1RXF14SID	0478	SID<10:3>								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx	
C1RXF14EID	047A	EID<15:8>								EID<7:0>									xxxx
C1RXF15SID	047C	SID<10:3>								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx	
C1RXF15EID	047E	EID<15:8>								EID<7:0>									xxxx

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

# dsPIC33FJXXXGPX06/X08/X10

## 4.6.2 DATA ACCESS FROM PROGRAM MEMORY USING TABLE INSTRUCTIONS

The **TBLRDL** and **TBLWTL** instructions offer a direct method of reading or writing the lower word of any address within the program space without going through data space. The **TBLRDH** and **TBLWTH** instructions are the only method to read or write the upper 8 bits of a program space word as data.

The PC is incremented by two for each successive 24-bit program word. This allows program memory addresses to directly map to data space addresses. Program memory can thus be regarded as two 16-bit word wide address spaces, residing side by side, each with the same address range. **TBLRDL** and **TBLWTL** access the space which contains the least significant data word and **TBLRDH** and **TBLWTH** access the space which contains the upper data byte.

Two table instructions are provided to move byte or word sized (16-bit) data to and from program space. Both function as either byte or word operations.

1. **TBLRDL** (Table Read Low): In Word mode, it maps the lower word of the program space location ( $P<15:0>$ ) to a data address ( $D<15:0>$ ).

In Byte mode, either the upper or lower byte of the lower program word is mapped to the lower byte of a data address. The upper byte is selected when Byte Select is '1'; the lower byte is selected when it is '0'.

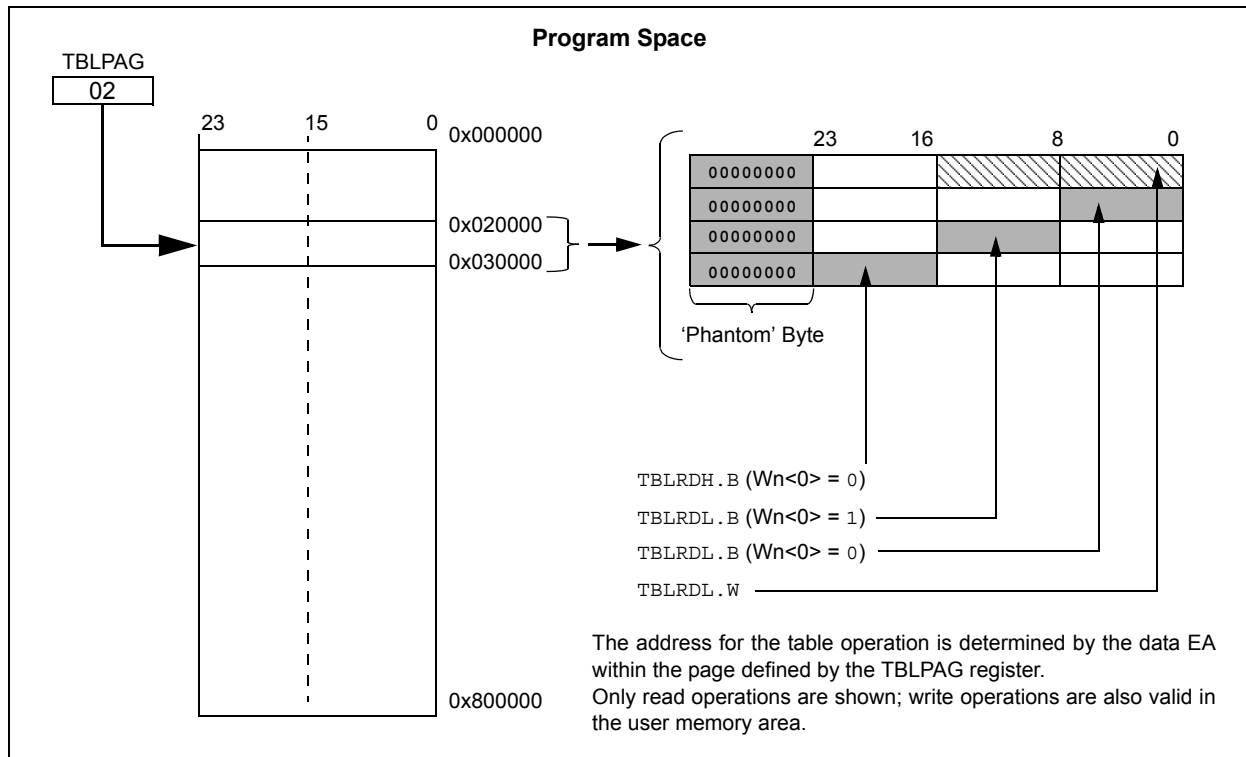
2. **TBLRDH** (Table Read High): In Word mode, it maps the entire upper word of a program address ( $P<23:16>$ ) to a data address. Note that  $D<15:8>$ , the 'phantom byte', will always be '0'.

In Byte mode, it maps the upper or lower byte of the program word to  $D<7:0>$  of the data address, as above. Note that the data will always be '0' when the upper 'phantom' byte is selected (Byte Select = 1).

In a similar fashion, two table instructions, **TBLWTH** and **TBLWTL**, are used to write individual bytes or words to a program space address. The details of their operation are explained in **Section 5.0 "Flash Program Memory"**.

For all table operations, the area of program memory space to be accessed is determined by the Table Page register (**TBLPAG**). **TBLPAG** covers the entire program memory space of the device, including user and configuration spaces. When  $TBLPAG<7> = 0$ , the table page is located in the user memory space. When  $TBLPAG<7> = 1$ , the page is located in configuration space.

**FIGURE 4-10: ACCESSING PROGRAM MEMORY WITH TABLE INSTRUCTIONS**





# dsPIC33FJXXXGPX06/X08/X10

## REGISTER 7-4: INTCON2: INTERRUPT CONTROL REGISTER 2

R/W-0	R-0	U-0	U-0	U-0	U-0	U-0	U-0
ALTIVT	DISI	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	INT4EP	INT3EP	INT2EP	INT1EP	INT0EP
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      **ALTIVT:** Enable Alternate Interrupt Vector Table bit  
                  1 = Use alternate vector table  
                  0 = Use standard (default) vector table
- bit 14      **DISI:** DISI Instruction Status bit  
                  1 = DISI instruction is active  
                  0 = DISI instruction is not active
- bit 13-5    **Unimplemented:** Read as '0'
- bit 4        **INT4EP:** External Interrupt 4 Edge Detect Polarity Select bit  
                  1 = Interrupt on negative edge  
                  0 = Interrupt on positive edge
- bit 3        **INT3EP:** External Interrupt 3 Edge Detect Polarity Select bit  
                  1 = Interrupt on negative edge  
                  0 = Interrupt on positive edge
- bit 2        **INT2EP:** External Interrupt 2 Edge Detect Polarity Select bit  
                  1 = Interrupt on negative edge  
                  0 = Interrupt on positive edge
- bit 1        **INT1EP:** External Interrupt 1 Edge Detect Polarity Select bit  
                  1 = Interrupt on negative edge  
                  0 = Interrupt on positive edge
- bit 0        **INT0EP:** External Interrupt 0 Edge Detect Polarity Select bit  
                  1 = Interrupt on negative edge  
                  0 = Interrupt on positive edge

## REGISTER 7-11: IEC1: INTERRUPT ENABLE CONTROL REGISTER 1 (CONTINUED)

bit 3	<b>CNIE:</b> Input Change Notification Interrupt Enable bit 1 = Interrupt request enabled 0 = Interrupt request not enabled
bit 2	<b>Unimplemented:</b> Read as '0'
bit 1	<b>MI2C1IE:</b> I2C1 Master Events Interrupt Enable bit 1 = Interrupt request enabled 0 = Interrupt request not enabled
bit 0	<b>SI2C1IE:</b> I2C1 Slave Events Interrupt Enable bit 1 = Interrupt request enabled 0 = Interrupt request not enabled

# dsPIC33FJXXXGPX06/X08/X10

## REGISTER 7-33: INTTREG: INTERRUPT CONTROL AND STATUS REGISTER

U-0	U-0	U-0	U-0	R-0	R-0	R-0	R-0
—	—	—	—	ILR<3:0>			
bit 15				bit 8			
U-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
—	VECNUM<6:0>						
bit 7				bit 0			

<b>Legend:</b>			
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

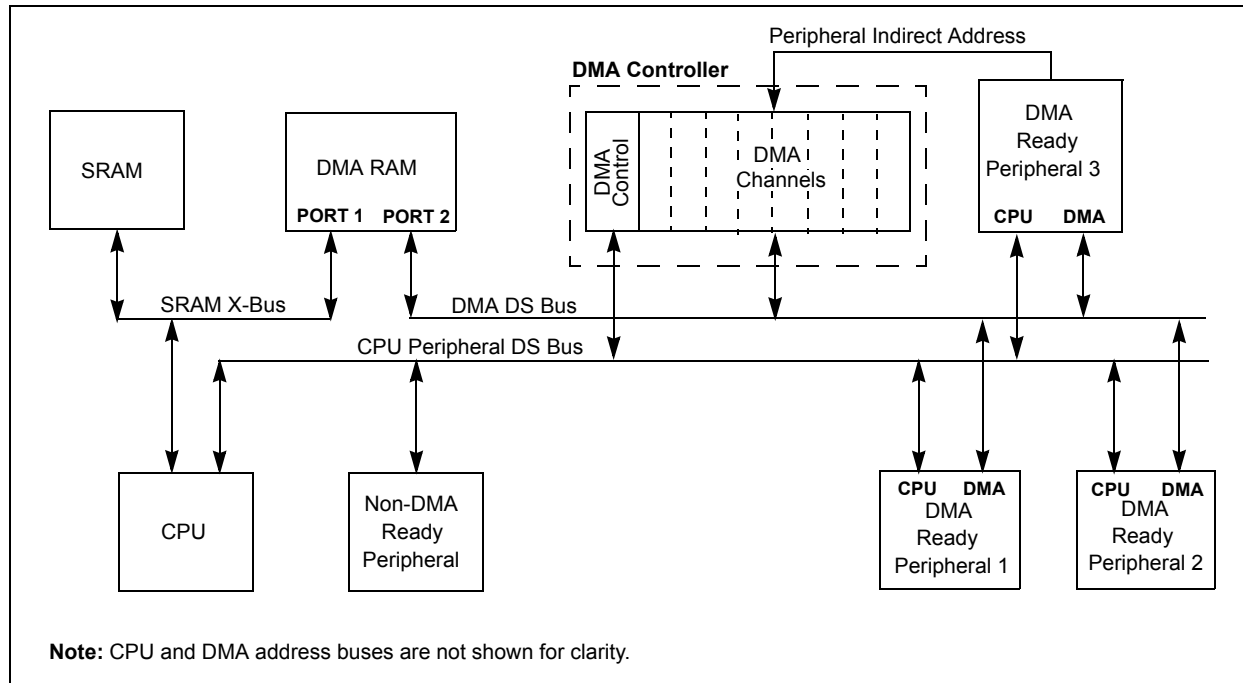
**ILR<3:0>:** New CPU Interrupt Priority Level bits  
1111 = CPU Interrupt Priority Level is 15  
•  
•  
•  
0001 = CPU Interrupt Priority Level is 1  
0000 = CPU Interrupt Priority Level is 0
- bit 7

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

**VECNUM<6:0>:** Vector Number of Pending Interrupt bits  
0111111 = Interrupt Vector pending is number 135  
•  
•  
•  
0000001 = Interrupt Vector pending is number 9  
0000000 = Interrupt Vector pending is number 8

# dsPIC33FJXXXGPX06/X08/X10

**FIGURE 8-1: TOP LEVEL SYSTEM ARCHITECTURE USING A DEDICATED TRANSACTION BUS**



## 8.1 DMAC Registers

Each DMAC Channel  $x$  ( $x = 0, 1, 2, 3, 4, 5, 6$  or  $7$ ) contains the following registers:

- A 16-bit DMA Channel Control register (DMAxCON)
- A 16-bit DMA Channel IRQ Select register (DMAxREQ)
- A 16-bit DMA RAM Primary Start Address Offset register (DMAxSTA)
- A 16-bit DMA RAM Secondary Start Address Offset register (DMAxSTB)
- A 16-bit DMA Peripheral Address register (DMAxPAD)
- A 10-bit DMA Transfer Count register (DMAxCNT)

An additional pair of status registers, DMACS0 and DMACS1, are common to all DMAC channels.

# dsPIC33FJXXXGPX06/X08/X10

## 15.1 Output Compare Modes

Configure the Output Compare modes by setting the appropriate Output Compare Mode (OCM<2:0>) bits in the Output Compare Control (OCxCON<2:0>) register. Table 15-1 lists the different bit settings for the Output Compare modes. Figure 15-2 illustrates the output compare operation for various modes. The user

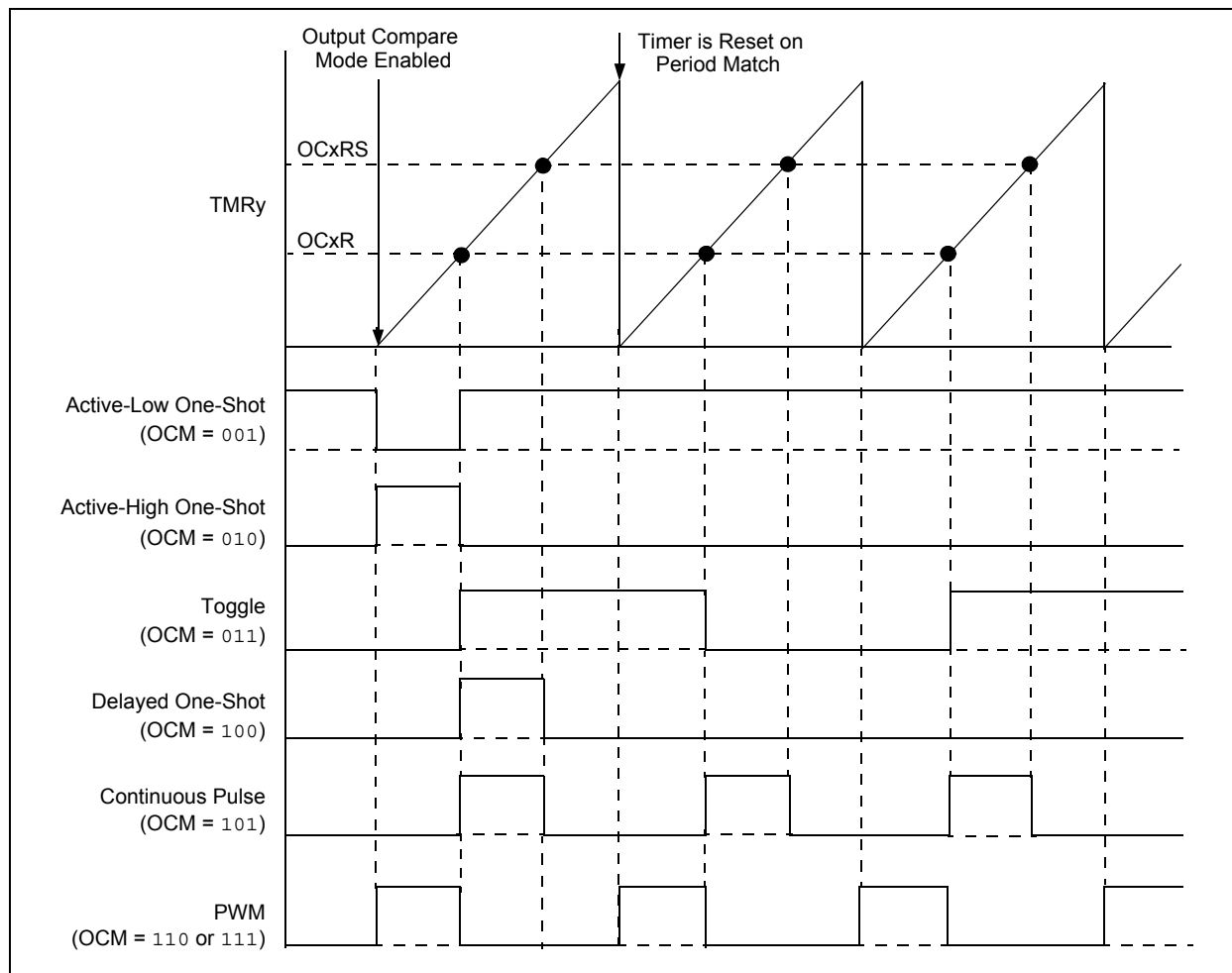
application must disable the associated timer when writing to the Output Compare Control registers to avoid malfunctions.

**Note:** See **Section 13. “Output Compare”** (DS70209) in the “dsPIC33F Family Reference Manual” for OCxR and OCxRS register restrictions.

**TABLE 15-1: OUTPUT COMPARE MODES**

OCM<2:0>	Mode	OCx Pin Initial State	OCx Interrupt Generation
000	Module Disabled	Controlled by GPIO register	—
001	Active-Low One-Shot	0	OCx rising edge
010	Active-High One-Shot	1	OCx falling edge
011	Toggle	Current output is maintained	OCx rising and falling edge
100	Delayed One-Shot	0	OCx falling edge
101	Continuous Pulse	0	OCx falling edge
110	PWM without Fault Protection	'0', if OCxR is zero '1', if OCxR is non-zero	No interrupt
111	PWM with Fault Protection	'0', if OCxR is zero '1', if OCxR is non-zero	OCFA falling edge for OC1 to OC4

**FIGURE 15-2: OUTPUT COMPARE OPERATION**



# dsPIC33FJXXXGPX06/X08/X10

## REGISTER 18-2: UxSTA: UARTx STATUS AND CONTROL REGISTER

R/W-0	R/W-0	R/W-0	U-0	R/W-0 HC	R/W-0	R-0	R-1
UTXISEL1	UTXINV	UTXISEL0	—	UTXBRK	UTXEN <sup>(1)</sup>	UTXBF	TRMT
bit 15						bit 8	

R/W-0	R/W-0	R/W-0	R-1	R-0	R-0	R/C-0	R-0
URXISEL<1:0>		ADDEN	RIDL	PERR	FERR	OERR	URXDA
bit 7							bit 0

<b>Legend:</b>	HC = Hardware cleared		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

- bit 15,13 **UTXISEL<1:0>**: Transmission Interrupt Mode Selection bits  
 11 = Reserved; do not use  
 10 = Interrupt when a character is transferred to the Transmit Shift Register, and as a result, the transmit buffer becomes empty  
 01 = Interrupt when the last character is shifted out of the Transmit Shift Register; all transmit operations are completed  
 00 = Interrupt when a character is transferred to the Transmit Shift Register (this implies there is at least one character open in the transmit buffer)
- bit 14 **UTXINV**: Transmit Polarity Inversion bit  
 If IREN = 0:  
 1 = UxTX Idle state is '0'  
 0 = UxTX Idle state is '1'  
 If IREN = 1:  
 1 = IrDA<sup>®</sup> encoded UxTX Idle state is '1'  
 0 = IrDA<sup>®</sup> encoded UxTX Idle state is '0'
- bit 12 **Unimplemented**: Read as '0'
- bit 11 **UTXBRK**: Transmit Break bit  
 1 = Send Sync Break on next transmission - Start bit, followed by twelve '0' bits, followed by Stop bit; cleared by hardware upon completion  
 0 = Sync Break transmission disabled or completed
- bit 10 **UTXEN**: Transmit Enable bit<sup>(1)</sup>  
 1 = Transmit enabled, UxTX pin controlled by UARTx  
 0 = Transmit disabled, any pending transmission is aborted and buffer is reset. UxTX pin controlled by port
- bit 9 **UTXBF**: Transmit Buffer Full Status bit (read-only)  
 1 = Transmit buffer is full  
 0 = Transmit buffer is not full, at least one more character can be written
- bit 8 **TRMT**: Transmit Shift Register Empty bit (read-only)  
 1 = Transmit Shift Register is empty and transmit buffer is empty (the last transmission has completed)  
 0 = Transmit Shift Register is not empty, a transmission is in progress or queued
- bit 7-6 **URXISEL<1:0>**: Receive Interrupt Mode Selection bits  
 11 = Interrupt is set on UxRSR transfer making the receive buffer full (i.e., has 4 data characters)  
 10 = Interrupt is set on UxRSR transfer making the receive buffer 3/4 full (i.e., has 3 data characters)  
 0x = Interrupt is set when any character is received and transferred from the UxRSR to the receive buffer. Receive buffer has one or more characters

**Note 1:** Refer to **Section 17. "UART"** (DS70188) in the *"dsPIC33F Family Reference Manual"* for information on enabling the UART module for transmit operation.

# dsPIC33FJXXXGPX06/X08/X10

## REGISTER 19-3: CIVEC: ECAN™ INTERRUPT CODE REGISTER

U-0	U-0	U-0	R-0	R-0	R-0	R-0	R-0
—	—	—	FILHIT<4:0>				
bit 15							bit 8

U-0	R-1	R-0	R-0	R-0	R-0	R-0	R-0	
—	ICODE<6:0>							
bit 7								bit 0

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

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

bit 12-8 **FILHIT<4:0>:** Filter Hit Number bits

10000-11111 = Reserved

01111 = Filter 15

•

•

•

00001 = Filter 1

00000 = Filter 0

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

bit 6-0 **ICODE<6:0>:** Interrupt Flag Code bits

1000101-1111111 = Reserved

1000100 = FIFO almost full interrupt

1000011 = Receiver overflow interrupt

1000010 = Wake-up interrupt

1000001 = Error interrupt

1000000 = No interrupt

0010000-0111111 = Reserved

0001111 = RB15 buffer Interrupt

•

•

•

0001001 = RB9 buffer interrupt

0001000 = RB8 buffer interrupt

0000111 = TRB7 buffer interrupt

0000110 = TRB6 buffer interrupt

0000101 = TRB5 buffer interrupt

0000100 = TRB4 buffer interrupt

0000011 = TRB3 buffer interrupt

0000010 = TRB2 buffer interrupt

0000001 = TRB1 buffer interrupt

0000000 = TRB0 Buffer interrupt

# dsPIC33FJXXXGPX06/X08/X10

## REGISTER 19-8: CIEC: ECAN™ TRANSMIT/RECEIVE ERROR COUNT REGISTER

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
TERRCNT<7:0>							
bit 15				bit 8			

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
RERRCNT<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 15-8      **TERRCNT<7:0>**: Transmit Error Count bits

bit 7-0      **RERRCNT<7:0>**: Receive Error Count bits



# dsPIC33FJXXXGPX06/X08/X10

REGISTER 19-15: CiBUFPNT4: ECAN™ FILTER 12-15 BUFFER POINTER REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F15BP<3:0>				F14BP<3:0>			
bit 15				bit 8			

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F13BP<3:0>				F12BP<3:0>			
bit 7				bit 0			

<b>Legend:</b>			
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      **F15BP<3:0>**: RX Buffer Written when Filter 15 Hits bits
- bit 11-8      **F14BP<3:0>**: RX Buffer Written when Filter 14 Hits bits
- bit 7-4        **F13BP<3:0>**: RX Buffer Written when Filter 13 Hits bits
- bit 3-0        **F12BP<3:0>**: RX Buffer Written when Filter 12 Hits bits

# dsPIC33FJXXXGPX06/X08/X10

## REGISTER 20-1: DCICON1: DCI CONTROL REGISTER 1

R/W-0	U-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
DCIEN	—	DCISIDL	—	DLOOP	CCKD	CCKE	COFSD
bit 15							bit 8

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0
UNFM	CSDOM	DJST	—	—	—	COFSM<1:0>	
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

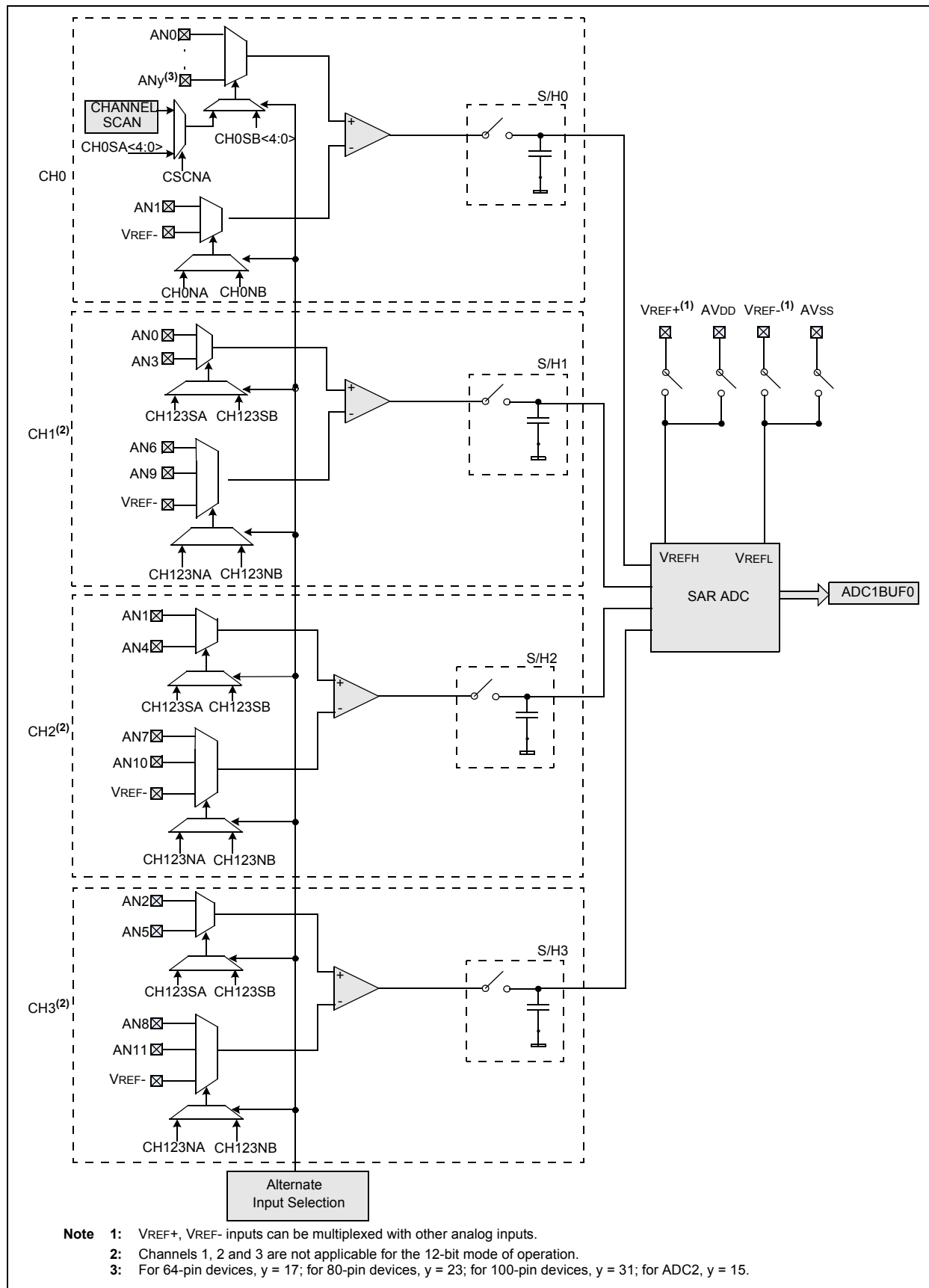
'0' = Bit is cleared

x = Bit is unknown

- bit 15 **DCIEN:** DCI Module Enable bit  
1 = Module is enabled  
0 = Module is disabled
- bit 14 **Unimplemented:** Read as '0'
- bit 13 **DCISIDL:** DCI Stop in Idle Control bit  
1 = Module will halt in CPU Idle mode  
0 = Module will continue to operate in CPU Idle mode
- bit 12 **Unimplemented:** Read as '0'
- bit 11 **DLOOP:** Digital Loopback Mode Control bit  
1 = Digital Loopback mode is enabled. CSDI and CSDO pins internally connected  
0 = Digital Loopback mode is disabled
- bit 10 **CCKD:** Sample Clock Direction Control bit  
1 = CCK pin is an input when DCI module is enabled  
0 = CCK pin is an output when DCI module is enabled
- bit 9 **CCKE:** Sample Clock Edge Control bit  
1 = Data changes on serial clock falling edge, sampled on serial clock rising edge  
0 = Data changes on serial clock rising edge, sampled on serial clock falling edge
- bit 8 **COFSD:** Frame Synchronization Direction Control bit  
1 = COFS pin is an input when DCI module is enabled  
0 = COFS pin is an output when DCI module is enabled
- bit 7 **UNFM:** Underflow Mode bit  
1 = Transmit last value written to the transmit registers on a transmit underflow  
0 = Transmit '0's on a transmit underflow
- bit 6 **CSDOM:** Serial Data Output Mode bit  
1 = CSDO pin will be tri-stated during disabled transmit time slots  
0 = CSDO pin drives '0's during disabled transmit time slots
- bit 5 **DJST:** DCI Data Justification Control bit  
1 = Data transmission/reception is begun during the same serial clock cycle as the frame synchronization pulse  
0 = Data transmission/reception is begun one serial clock cycle after frame synchronization pulse
- bit 4-2 **Unimplemented:** Read as '0'
- bit 1-0 **COFSM<1:0>:** Frame Sync Mode bits  
11 = 20-bit AC-Link mode  
10 = 16-bit AC-Link mode  
01 = I<sup>2</sup>S Frame Sync mode  
00 = Multi-Channel Frame Sync mode

# dsPIC33FJXXXGPX06/X08/X10

**FIGURE 21-1: ADCx MODULE BLOCK DIAGRAM**



**TABLE 23-1: SYMBOLS USED IN OPCODE DESCRIPTIONS (CONTINUED)**

Field	Description
Wm*Wm	Multiplicand and Multiplier working register pair for Square instructions $\in \{W4 * W4, W5 * W5, W6 * W6, W7 * W7\}$
Wm*Wn	Multiplicand and Multiplier working register pair for DSP instructions $\in \{W4 * W5, W4 * W6, W4 * W7, W5 * W6, W5 * W7, W6 * W7\}$
Wn	One of 16 working registers $\in \{W0..W15\}$
Wnd	One of 16 destination working registers $\in \{W0..W15\}$
Wns	One of 16 source working registers $\in \{W0..W15\}$
WREG	W0 (working register used in file register instructions)
Ws	Source W register $\in \{Ws, [Ws], [Ws++] , [Ws--], [++Ws], [--Ws] \}$
Wso	Source W register $\in \{Wns, [Wns], [Wns++] , [Wns--], [++Wns], [--Wns], [Wns+Wb] \}$
Wx	X data space prefetch address register for DSP instructions $\in \{[W8]+ = 6, [W8]+ = 4, [W8]+ = 2, [W8], [W8]- = 6, [W8]- = 4, [W8]- = 2, [W9]+ = 6, [W9]+ = 4, [W9]+ = 2, [W9], [W9]- = 6, [W9]- = 4, [W9]- = 2, [W9 + W12], \text{none}\}$
Wxd	X data space prefetch destination register for DSP instructions $\in \{W4..W7\}$
Wy	Y data space prefetch address register for DSP instructions $\in \{[W10]+ = 6, [W10]+ = 4, [W10]+ = 2, [W10], [W10]- = 6, [W10]- = 4, [W10]- = 2, [W11]+ = 6, [W11]+ = 4, [W11]+ = 2, [W11], [W11]- = 6, [W11]- = 4, [W11]- = 2, [W11 + W12], \text{none}\}$
Wyd	Y data space prefetch destination register for DSP instructions $\in \{W4..W7\}$

# dsPIC33FJXXXGPX06/X08/X10

ECAN2 Register Map (C2CTRL1.WIN = 0 or 1) .....	55
ECAN2 Register Map (C2CTRL1.WIN = 0) .....	55, 56
Frame Types .....	191
Modes of Operation .....	193
Overview .....	191
ECAN Registers	
Filter 15-8 Mask Selection Register (CiFMSKSEL2) .....	209
Electrical Characteristics .....	257
AC .....	266
Enhanced CAN Module .....	191
Equations	
Device Operating Frequency .....	138
Errata .....	12
<b>F</b>	
Flash Program Memory .....	71
Control Registers .....	72
Operations .....	72
Programming Algorithm .....	75
RTSP Operation .....	72
Table Instructions .....	71
Flexible Configuration .....	237
FSCM	
Delay for Crystal and PLL Clock Sources .....	80
Device Resets .....	80
<b>I</b>	
I/O Ports .....	155
Parallel I/O (PIO) .....	155
Write/Read Timing .....	156
I <sup>2</sup> C	
Operating Modes .....	177
Registers .....	177
I <sup>2</sup> C Module	
I2C1 Register Map .....	47
I2C2 Register Map .....	47
In-Circuit Debugger .....	243
In-Circuit Emulation .....	237
In-Circuit Serial Programming (ICSP) .....	237, 243
Input Capture	
Registers .....	166
Input Change Notification Module .....	156
Instruction Addressing Modes .....	61
File Register Instructions .....	61
Fundamental Modes Supported .....	62
MAC Instructions .....	62
MCU Instructions .....	61
Move and Accumulator Instructions .....	62
Other Instructions .....	62
Instruction Set	
Overview .....	248
Summary .....	245
Instruction-Based Power-Saving Modes .....	147
Idle .....	148
Sleep .....	147
Internal RC Oscillator	
Use with WDT .....	242
Internet Address .....	317
Interrupt Control and Status Registers .....	85
IECx .....	85
IFSx .....	85
INTCON1 .....	85
INTCON2 .....	85
IPCx .....	85
Interrupt Setup Procedures .....	125
Initialization .....	125
Interrupt Disable .....	125
Interrupt Service Routine .....	125
Trap Service Routine .....	125
Interrupt Vector Table (IVT) .....	81
Interrupts Coincident with Power Save Instructions .....	148
<b>J</b>	
JTAG Boundary Scan Interface .....	237
<b>M</b>	
Memory Organization .....	33
Microchip Internet Web Site .....	317
Modes of Operation	
Disable .....	193
Initialization .....	193
Listen All Messages .....	193
Listen Only .....	193
Loopback .....	193
Normal Operation .....	193
Modulo Addressing .....	62
Applicability .....	64
Operation Example .....	63
Start and End Address .....	63
W Address Register Selection .....	63
MPLAB ASM30 Assembler, Linker, Librarian .....	254
MPLAB ICD 2 In-Circuit Debugger .....	255
MPLAB ICE 2000 High-Performance Universal In-Circuit Emulator .....	255
MPLAB Integrated Development Environment Software .....	253
MPLAB PM3 Device Programmer .....	255
MPLAB REAL ICE In-Circuit Emulator System .....	255
MPLINK Object Linker/MPLIB Object Librarian .....	254
<b>N</b>	
NVM Module	
Register Map .....	60
<b>O</b>	
Open-Drain Configuration .....	156
Output Compare .....	167
<b>P</b>	
Packaging .....	297
Details .....	298
Marking .....	297
Peripheral Module Disable (PMD) .....	148
PICSTART Plus Development Programmer .....	256
Pinout I/O Descriptions (table) .....	15
PMD Module	
Register Map .....	60
POR and Long Oscillator Start-up Times .....	80
PORTA	
Register Map .....	58
PORTB	
Register Map .....	58
PORTC	
Register Map .....	59
PORTD	
Register Map .....	59
PORTE	
Register Map .....	59
PORTF	
Register Map .....	59
PORTG	
Register Map .....	60
Power-Saving Features .....	147