

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	32MHz
Connectivity	I ² C, IrDA, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, I ² S, LVD, POR, PWM, WDT
Number of I/O	36
Program Memory Size	64KB (22K x 24)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	16К х 8
Voltage - Supply (Vcc/Vdd)	2V ~ 3.6V
Data Converters	A/D 14x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°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/pic24fj64ga704-i-pt

Email: info@E-XFL.COM

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

2.4 Voltage Regulator Pin (VCAP)

Note: This section applies only to PIC24FJ devices with an on-chip voltage regulator.

Refer to **Section 29.3** "**On-Chip Voltage Regulator**" for details on connecting and using the on-chip regulator.

A low-ESR (< 5 Ω) capacitor is required on the VCAP pin to stabilize the voltage regulator output voltage. The VCAP pin must not be connected to VDD and must use a capacitor of 10 μ F connected to ground. The type can be ceramic or tantalum. Suitable examples of capacitors are shown in Table 2-1. Capacitors with equivalent specifications can be used.

Designers may use Figure 2-3 to evaluate the ESR equivalence of candidate devices.

The placement of this capacitor should be close to VCAP. It is recommended that the trace length not exceed 0.25 inch (6 mm). Refer to **Section 32.0** "**Electrical Characteristics**" for additional information.



TABLE 2-1: SUITABLE CAPACITOR EQUIVALENTS (0805 CASE SIZE)

Make	Part #	Nominal Capacitance	Base Tolerance	Rated Voltage
TDK	C2012X5R1E106K085AC	10 µF	±10%	25V
TDK	C2012X5R1C106K085AC	10 µF	±10%	16V
Kemet	C0805C106M4PACTU	10 µF	±10%	16V
Murata	GRM21BR61E106KA3L	10 µF	±10%	25V
Murata	GRM21BR61C106KE15	10 µF	±10%	16V

3.2 CPU Control Registers

REGISTER 3-1: SR: ALU STATUS REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
_	—	—		_	—	—	DC
bit 15							bit 8
R/W-0 ⁽¹⁾	R/W-0 ⁽¹⁾	R/W-0 ⁽¹⁾	R-0	R/W-0	R/W-0	R/W-0	R/W-0
IPL2 ⁽²⁾	IPL1 ⁽²⁾	IPL0 ⁽²⁾	RA	N	OV	Z	С
bit 7							bit 0
Legend:							
R = Readal	ole bit	W = Writable b	bit	U = Unimplem	nented bit, read	l as '0'	
-n = Value a	at POR	'1' = Bit is set		'0' = Bit is clea	ared	x = Bit is unkn	iown
bit 15-9	Unimplement	ed: Read as '0					
bit 8	DC: ALU Half	Carry/Borrow t	Dit avv. and an bit (f:		ata) ar O th Iaur a	and an bit /fam.u.a	
	⊥ = A carry of of the res	ult occurred	ow-order bit (in	or byte-sized da	ata) or 8 th low-0	brder bit (for wo	ord-sized data)
	0 = No carry	out from the 4 th	or 8 th low-ord	ler bit of the res	sult has occurre	ed	
bit 7-5	IPL<2:0>: CP	U Interrupt Pric	ority Level Stat	us bits ^(1,2)			
	111 = CPU In	terrupt Priority	Level is 7 (15)	; user interrupt	s are disabled		
	110 = CPU In	terrupt Priority	Level is 6 (14)				
	101 = CPU In 100 = CPU In	terrupt Priority	Level is 5 (13) Level is 4 (12)				
	011 = CPU In	terrupt Priority	Level is 3 (11)				
	010 = CPU In	terrupt Priority	Level is 2 (10)	1			
	001 = CPU In	terrupt Priority	Level is 1 (9)				
hit 4							
	1 = REPEAT	oop is in progre	ss				
	0 = REPEAT IC	op is not in pro	gress				
bit 3	N: ALU Negat	ive bit					
	1 = Result wa	s negative					
	0 = Result wa	s not negative (zero or positiv	ve)			
bit 2	OV: ALU Over	flow bit					
	1 = Overflow d 0 = No overflow	occurred for sig	ned (2's comp 1	lement) arithm	etic in this arith	metic operation	1
bit 1	Z: ALU Zero b	oit					
-	1 = An operati	ion, which affec	ts the Z bit, ha	as set it at some	e time in the pa	ist	
	0 = The most	recent operatio	n, which affec	ts the Z bit, has	cleared it (i.e.	, a non-zero res	sult)
bit 0	C: ALU Carry/	Borrow bit					
	1 = A carry ou	t from the Mos	t Significant bit	t (MSb) of the r	esult occurred		
	0 = No carry c	out from the Mo	st Significant t	bit of the result	occurred		
Note 1:	The IPLx Status b	its are read-onl	y when NSTD	IS (INTCON1<	15>) = 1.		

2: The IPLx Status bits are concatenated with the IPL3 Status bit (CORCON<3>) to form the CPU Interrupt Priority Level (IPL). The value in parentheses indicates the IPL when IPL3 = 1.

4.2.5 EXTENDED DATA SPACE (EDS)

The Extended Data Space (EDS) allows PIC24F devices to address a much larger range of data than would otherwise be possible with a 16-bit address range. EDS includes any additional internal data memory not directly accessible by the lower 32-Kbyte data address space and any external memory through EPMP.

In addition, EDS also allows read access to the program memory space. This feature is called Program Space Visibility (PSV) and is discussed in detail in Section 4.3.3 "Reading Data from Program Memory Using EDS".

Figure 4-3 displays the entire EDS space. The EDS is organized as pages, called EDS pages, with one page equal to the size of the EDS window (32 Kbytes). A particular EDS page is selected through the Data Space Read Page register (DSRPAG) or the Data Space Write Page register (DSWPAG). For PSV, only the DSRPAG register is used. The combination of the DSRPAG register value and the 16-bit wide data address forms a 24-bit Effective Address (EA).

The data addressing range of the PIC24FJ256GA705 family devices depends on the version of the Enhanced Parallel Master Port implemented on a particular device; this is, in turn, a function of device pin count. Table 4-12 lists the total memory accessible by each of the devices in this family. For more details on accessing external memory using EPMP, refer to the "dsPIC33/PIC24 Family Reference Manual', "Enhanced Parallel Master Port (EPMP)" (DS39730).

TABLE 4-12:	TOTAL ACCESSIBLE DATA
	MEMORY

Family	Internal RAM	External RAM Access Using EPMP
PIC24FJXXXGA70X	16K	1K

Note: Accessing Page 0 in the EDS window will generate an address error trap as Page 0 is the base data memory (data locations, 0800h to 7FFFh, in the lower Data Space).



FIGURE 4-3: **EXTENDED DATA SPACE**

10.2.2 IDLE MODE

Idle mode has these features:

- The CPU will stop executing instructions.
- The WDT is automatically cleared.
- The system clock source remains active. By default, all peripheral modules continue to operate normally from the system clock source, but can also be selectively disabled (see Section 10.4 "Selective Peripheral Module Control").
- If the WDT or FSCM is enabled, the LPRC will also remain active.

The device will wake from Idle mode on any of these events:

- Any interrupt that is individually enabled.
- · Any device Reset.
- · A WDT time-out.

On wake-up from Idle, the clock is reapplied to the CPU and instruction execution begins immediately, starting with the instruction following the PWRSAV instruction or the first instruction in the ISR.

10.2.3 INTERRUPTS COINCIDENT WITH POWER SAVE INSTRUCTIONS

Any interrupt that coincides with the execution of a PWRSAV instruction will be held off until entry into Sleep or Idle mode has completed. The device will then wake-up from Sleep or Idle mode.

10.2.4 LOW-VOLTAGE RETENTION REGULATOR

PIC24FJ256GA705 family devices incorporate a second on-chip voltage regulator, designed to provide power to select microcontroller features at 1.2V nominal. This regulator allows features, such as data RAM and the WDT, to be maintained in power-saving modes where they would otherwise be inactive, or maintain them at a lower power than would otherwise be the case.

Retention Sleep uses less power than standard Sleep mode, but takes more time to recover and begin execution. An additional 10-15 μ S (typical) is required to charge VCAP from 1.2V to 1.8V and start to execute instructions when exiting Retention Sleep.

The VREGS bit allows control of speed to exit from the Sleep modes (regular and Retention) at the cost of more power. The regulator band gaps are enabled, which increases the current but reduces time to recover from Sleep by ~10 μ s.

The low-voltage retention regulator is only available when Sleep mode is invoked. It is controlled by the LPCFG Configuration bit (FPOR<2>) and in firmware by the RETEN bit (RCON<12>). LPCFG must be programmed (= 0) and the RETEN bit must be set (= 1) for the regulator to be enabled.

10.2.5 EXITING FROM LOW-VOLTAGE RETENTION SLEEP

All of the methods for exiting from standard Sleep also apply to Retention Sleep (MCLR, INT0, etc.). However, in order to allow the regulator to switch from 1.8V (operating) to Retention mode (1.2V), there is a hardware 'lockout timer' from the execution of Retention Sleep until Retention Sleep can be exited.

During the 'lockout time', the only method to exit Retention Sleep is a POR or MCLR. Interrupts that are asserted (such as INT0) during the 'lockout time' are masked. The lockout timer then sets a minimum interval from when the part enters Retention Sleep until it can exit from Retention Sleep. Interrupts are not 'held pending' during lockout; they are masked and in order to exit after the lockout expires, the exiting source must assert after the lockout time.

The lockout timer is derived from the LPRC clock, which has a wide (untrimmed) frequency tolerance.

The lockout time will be one of the following two cases:

- If the LPRC was not running at the time of Retention Sleep, the lockout time is 2 LPRC periods + LPRC wake-up time
- If the LPRC was running at the time of Retention Sleep, the lockout time is 1 LPRC period

Refer to Table 32-20 and Table 32-21 in the AC Electrical Specifications for the LPRC timing.

10.2.6 SUMMARY OF LOW-POWER SLEEP MODES

The RETEN bit and the VREGS bit (RCON<12,8>) allow for four different Sleep modes, which will vary by wake-up time and power consumption. Refer to Table 10-1 for a summary of these modes. Specific information about the current consumption and wake times can be found in **Section 32.0 "Electrical Characteristics"**.

TABLE 10-1: LOW-POWER SLEEP MODES

RETEN	VREGS	MODE	Relative Power (1 = Lowest)
0	0	Sleep	3
0	1	Fast Wake-up	4
1	0	Retention Sleep	1
1	1	Fast Retention	2

11.5.3.3 Mapping Limitations

The control schema of the Peripheral Pin Select is extremely flexible. Other than systematic blocks that prevent signal contention, caused by two physical pins being configured as the same functional input or two functional outputs configured as the same pin, there are no hardware enforced lockouts. The flexibility extends to the point of allowing a single input to drive multiple peripherals or a single functional output to drive multiple output pins.

11.5.3.4 Mapping Exceptions for Family Devices

The differences in available remappable pins are summarized in Table 11-8.

When developing applications that use remappable pins, users should also keep these things in mind:

- For the RPINRx registers, bit combinations corresponding to an unimplemented pin for a particular device are treated as invalid; the corresponding module will not have an input mapped to it.
- For RPORx registers, the bit fields corresponding to an unimplemented pin will also be unimplemented; writing to these fields will have no effect.

11.5.4 CONTROLLING CONFIGURATION CHANGES

Because peripheral remapping can be changed during run time, some restrictions on peripheral remapping are needed to prevent accidental configuration changes. PIC24F devices include three features to prevent alterations to the peripheral map:

- · Control register lock sequence
- · Continuous state monitoring
- Configuration bit remapping lock

11.5.4.1 Control Register Lock

Under normal operation, writes to the RPINRx and RPORx registers are not allowed. Attempted writes will appear to execute normally, but the contents of the

18

registers will remain unchanged. To change these registers, they must be unlocked in hardware. The register lock is controlled by the IOLOCK bit (OSCCON<6>). Setting IOLOCK prevents writes to the control registers; clearing IOLOCK allows writes.

To set or clear IOLOCK, a specific command sequence must be executed:

- 1. Write 46h to OSCCON<7:0>.
- 2. Write 57h to OSCCON<7:0>.
- 3. Clear (or set) IOLOCK as a single operation.

Unlike the similar sequence with the oscillator's LOCK bit, IOLOCK remains in one state until changed. This allows all of the Peripheral Pin Selects to be configured with a single unlock sequence, followed by an update to all control registers, then locked with a second lock sequence.

11.5.4.2 Continuous State Monitoring

In addition to being protected from direct writes, the contents of the RPINRx and RPORx registers are constantly monitored in hardware by shadow registers. If an unexpected change in any of the registers occurs (such as cell disturbances caused by ESD or other external events), a Configuration Mismatch Reset will be triggered.

11.5.4.3 Configuration Bit Pin Select Lock

As an additional level of safety, the device can be configured to prevent more than one write session to the RPINRx and RPORx registers. The IOL1WAY (FOSC<5>) Configuration bit blocks the IOLOCK bit from being cleared after it has been set once. If IOLOCK remains set, the register unlock procedure will not execute and the Peripheral Pin Select Control registers cannot be written to. The only way to clear the bit and re-enable peripheral remapping is to perform a device Reset.

In the default (unprogrammed) state, IOL1WAY is set, restricting users to one write session. Programming IOL1WAY allows users unlimited access (with the proper use of the unlock sequence) to the Peripheral Pin Select registers.

0

Device		RPn Pins (I/O)	RPIn Pins		
Device	Total	Unimplemented	ented Total Unimplemented		
PIC24FJXXXGA705	29	_	4	_	
PIC24FJXXXGA704	29	_	0	RPI29-32	

RP16-25

TABLE 11-8: REMAPPABLE PIN EXCEPTIONS FOR PIC24FJ256GA705 FAMILY DEVICES

PIC24FJXXXGA702

RPI29-32

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
_	_	U1CTSR5	U1CTSR4	U1CTSR3	U1CTSR2	U1CTSR1	U1CTSR0
bit 15							bit 8
U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1

REGISTER 11-23: RPINR18: PERIPHERAL PIN SELECT INPUT REGISTER 18

	 U1RXR5	U1RXR4	U1RXR3	U1RXR2	U1RXR1	U1RXR0
bit 7						bit 0
Legend:						

R = Readable bit	W = Writable bit	U = Unimplemented bit, read	d as '0'
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-14	Unimplemented: Read as '0'
bit 13-8	U1CTSR<5:0>: Assign UART1 Clear-to-Send (U1CTS) to Corresponding RPn or RPIn Pin bits
bit 7-6	Unimplemented: Read as '0'
bit 5-0	U1RXR<5:0>: Assign UART1 Receive (U1RX) to Corresponding RPn or RPIn Pin bits

REGISTER 11-24: RPINR19: PERIPHERAL PIN SELECT INPUT REGISTER 19

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	U2CTSR5	U2CTSR4	U2CTSR3	U2CTSR2	U2CTSR1	U2CTSR0
bit 15							bit 8

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	U2RXR5	U2RXR4	U2RXR3	U2RXR2	U2RXR1	U2RXR0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read	d as '0'
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-14 Unimplemented: Read as '0'

U2CTSR<5:0>: Assign UART2 Clear-to-Send (U2CTS) to Corresponding RPn or RPIn Pin bits bit 13-8

bit 7-6 Unimplemented: Read as '0'

bit 5-0 U2RXR<5:0>: Assign UART2 Receive (U2RX) to Corresponding RPn or RPIn Pin bits For 32-bit cascaded operation, these steps are also necessary:

- 1. Set the OC32 bits for both registers (OCyCON2<8> and OCxCON2<8>). Enable the even numbered module first to ensure the modules will start functioning in unison.
- Clear the OCTRIG bit of the even module (OCyCON2<7>), so the module will run in Synchronous mode.
- 3. Configure the desired output and Fault settings for OCy.
- 4. Force the output pin for OCx to the output state by clearing the OCTRIS bit.
- If Trigger mode operation is required, configure the Trigger options in OCx by using the OCTRIG (OCxCON2<7>), TRIGMODE (OCxCON1<3>) and SYNCSEL<4:0> (OCxCON2<4:0>) bits.
- Configure the desired Compare or PWM mode of operation (OCM<2:0>) for OCy first, then for OCx.

Depending on the output mode selected, the module holds the OCx pin in its default state and forces a transition to the opposite state when OCxR matches the timer. In Double Compare modes, OCx is forced back to its default state when a match with OCxRS occurs. The OCxIF interrupt flag is set after an OCxR match in Single Compare modes and after each OCxRS match in Double Compare modes.

Single-Shot pulse events only occur once, but may be repeated by simply rewriting the value of the OCxCON1 register. Continuous pulse events continue indefinitely until terminated.

15.3 Pulse-Width Modulation (PWM) Mode

In PWM mode, the output compare module can be configured for edge-aligned or center-aligned pulse waveform generation. All PWM operations are doublebuffered (buffer registers are internal to the module and are not mapped into SFR space).

To configure the output compare module for PWM operation:

- 1. Configure the OCx output for one of the available Peripheral Pin Select pins if available on the OC module you are using. Otherwise, configure the dedicated OCx output pins.
- 2. Calculate the desired duty cycles and load them into the OCxR register.
- 3. Calculate the desired period and load it into the OCxRS register.
- Select the current OCx as the synchronization source by writing '0x1F' to the SYNCSEL<4:0> bits (OCxCON2<4:0>) and '0' to the OCTRIG bit (OCxCON2<7>).
- 5. Select a clock source by writing to the OCTSEL<2:0> bits (OCxCON1<12:10>).
- 6. Enable interrupts, if required, for the timer and output compare modules. The output compare interrupt is required for PWM Fault pin utilization.
- Select the desired PWM mode in the OCM<2:0> bits (OCxCON1<2:0>).
- Appropriate Fault inputs may be enabled by using the ENFLT<2:0> bits as described in Register 15-1.
- 9. If a timer is selected as a clock source, set the selected timer prescale value. The selected timer's prescaler output is used as the clock input for the OCx timer and not the selected timer output.

Note: This peripheral contains input and output functions that may need to be configured by the Peripheral Pin Select. See Section 11.5 "Peripheral Pin Select (PPS)" for more information.

15.3.2 PWM DUTY CYCLE

The PWM duty cycle is specified by writing to the OCxRS and OCxR registers. The OCxRS and OCxR registers can be written to at any time, but the duty cycle value is not latched until a match between PRy and TMRy occurs (i.e., the period is complete). This provides a double buffer for the PWM duty cycle and is essential for glitchless PWM operation.

Some important boundary parameters of the PWM duty cycle include:

- If OCxR, OCxRS and PRy are all loaded with 0000h, the OCx pin will remain low (0% duty cycle).
- If OCxRS is greater than PRy, the pin will remain high (100% duty cycle).

See Example 15-1 for PWM mode timing details. Table 15-1 and Table 15-2 show example PWM frequencies and resolutions for a device operating at 4 MIPS and 10 MIPS, respectively.

EQUATION 15-2: CALCULATION FOR MAXIMUM PWM RESOLUTION⁽¹⁾

Maximum PWM Resolution (bits) = $\frac{\log_{10} \left(\frac{F_{CY}}{F_{PWM} \cdot (T_{imer} Prescale Value)} \right)}{\log_{10} (2)}$ bits

Note 1: Based on Fcy = Fosc/2; Doze mode and PLL are disabled.

EXAMPLE 15-1: PWM PERIOD AND DUTY CYCLE CALCULATIONS⁽¹⁾

- Find the Timer Period register value for a desired PWM frequency of 52.08 kHz, where Fosc = 32 MHz with PLL (32 MHz device clock rate) and a Timer2 prescaler setting of 1:1.
 TCY = 2 Tosc = 62.5 ns
 PWM Period = 1/PWM Frequency = 1/52.08 kHz = 19.2 μS
 PWM Period = (PR2 + 1) TCY (Timer2 Prescale Value)
 19.2 μS = (PR2 + 1) 62.5 ns 1
 PR2 = 306

 Find the maximum resolution of the duty cycle that can be used with a 52.08 kHz frequency and a 32 MHz device clock rate:
 PWM Resolution = log₁₀(FCY/FPWM)/log₁₀2) bits
 = (log₁₀(16 MHz/52.08 kHz)/log₁₀2) bits
 = 8.3 bits
- Note 1: Based on TCY = 2 * TOSC; Doze mode and PLL are disabled.

TABLE 15-1:	EXAMPLE PWM FREQUENCIES AND RESOLUTIONS AT 4 MIPS ($FCY = 4 MHz)^{(1)}$)
-		-	

PWM Frequency	7.6 Hz	61 Hz	122 Hz	977 Hz	3.9 kHz	31.3 kHz	125 kHz
Timer Prescaler Ratio	8	1	1	1	1	1	1
Period Register Value	FFFFh	FFFFh	7FFFh	0FFFh	03FFh	007Fh	001Fh
Resolution (bits)	16	16	15	12	10	7	5

Note 1: Based on Fcy = Fosc/2; Doze mode and PLL are disabled.

TABLE 15-2: EXAMPLE PWM FREQUENCIES AND RESOLUTIONS AT 16 MIPS (Fcy = 16 MHz)⁽¹⁾

PWM Frequency	30.5 Hz	244 Hz	488 Hz	3.9 kHz	15.6 kHz	125 kHz	500 kHz
Timer Prescaler Ratio	8	1	1	1	1	1	1
Period Register Value	FFFFh	FFFFh	7FFFh	0FFFh	03FFh	007Fh	001Fh
Resolution (bits)	16	16	15	12	10	7	5

Note 1: Based on FCY = FOSC/2; Doze mode and PLL are disabled.

R/M/_0	11_0	R/M/-0	R/M/ 0	R///_0	R/\//_0	R/M_0	R/\\/_1
	<u> </u>						
bit 15		OCILIN	OCLEN	OCDEN	OCCEN	OCDEN	bit 8
							bit 0
R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ICGSM1	ICGSM0		AUXOUT1	AUXOUT0	ICS2	ICS1	ICS0
bit 7							bit 0
Legend:							
R = Readable	e bit	W = Writable	bit	U = Unimplem	nented bit, read	d as '0'	
-n = Value at	POR	'1' = Bit is set		'0' = Bit is clea	ared	x = Bit is unkr	nown
bit 15	OENSYNC: C	Output Enable S	Synchronizatio	n bit			
	1 = Update b	y output enable	e bits occurs or	n the next Time	Base Reset o	r rollover	
	0 = Update b	y output enable	e bits occurs in	nmediately			
bit 14	Unimplement	ted: Read as ')' . .				
bit 13-8	OCXEN: Outp	out Enable/Stee	ering Control bi	ts			
	1 = OCMx pir	n is controlled t	by the CCPX m led by the CC	odule and prod	nin is availab	t compare or P	www.signal
	periphera	al multiplexed o	n the pin				
bit 7-6	ICGSM<1:0>:	Input Capture	Gating Source	Mode Control	bits		
	11 = Reserve	d	-				
	10 = One-Sho	ot mode: Falling	g edge from ga	ting source dis	ables future ca	pture events (I	CDIS = 1)
	01 = One-Sho	ot mode: Rising	l edge from ga	ting source ena	ibles future ca	pture events (IC	DIS = 0
	level will	disable future	capture events	S			events, a low
bit 5	Unimplement	ted: Read as ')'				
bit 4-3	AUXOUT<1:0)>: Auxiliary Οι	utput Signal on	Event Selectio	n bits		
	11 = Input ca	pture or output	compare even	it; no signal in T	imer mode		
	10 = Signal o	utput is defined	I by module op	erating mode (see Table 16-4)	
	01 = 1 ime bas	se rollover eve 1	nt (all modes)				
bit 2-0		• out Capture So	urce Select bits	3			
5112 0	111 = Reserv	ved		5			
	110 = Reserv	ved					
	101 = CLC2 (output					
	100 = CLC1	output					
	011 = Compa	arator 2 output					
	001 = Compa	arator 1 output					
	000 = Input C	Capture x (ICM)	() I/O pin				

REGISTER 16-4: CCPxCON2H: CCPx CONTROL 2 HIGH REGISTERS

20.1 Memory Addressable in Different Modes

The memory space addressable by the device depends on the address/data multiplexing selection; it varies from 1K to 2 MB. Refer to Table 20-1 for different Memory-Addressable modes.

20.2 PMDOUT1 and PMDOUT2 Registers

The EPMP Data Output 1 and Data Output 2 registers are used only in Slave mode. These registers act as a buffer for outgoing data.

20.3 PMDIN1 and PMDIN2 Registers

The EPMP Data Input 1 and Data Input 2 registers are used in Slave modes to buffer incoming data. These registers hold data that is asynchronously clocked in. In Master mode, PMDIN1 is the holding register for incoming data.

Data Port Size	PMA<9:8>	PMA<7:0>	PMD<7:4>	PMD<3:0>	Accessible Memory
	Demulti	plexed Address	(ADRMUX<1:0>	= 00)	
8-Bit (PTSZ<1:0> = 00)	Addr<9:8>	Addr<7:0>	Da	ata	1K
4-Bit (PTSZ<1:0> = 01)	Addr<9:8>	Addr<7:0>	—	Data	1K
	1 Ad	dress Phase (AD	DRMUX<1:0> = 0	1)	
8-Bit (PTSZ<1:0> = 00)	—	PMALL	Addr<7	:0> Data	1K
4-Bit (PTSZ<1:0> = 01)	Addr<9:8>	PMALL	Addr<7:4>	Addr<3:0>	1K
			—	Data (1)	7
	2 Add	Iress Phases (Al	DRMUX<1:0> = 1	L0)	-
8-Bit (PTSZ<1:0> = 00)	—	PMALL	Addr	<7:0>	64K
		PMALH	Addr<	:15:8>	
		—	Da	ata	
4-Bit (PTSZ<1:0> = 01)	Addr<9:8>	PMALL	Addr	<3:0>	1K
		PMALH	Addr	<7:4>	
		—	Da	ata	1
	3 Add	Iress Phases (Al	DRMUX<1:0> = 1	1)	-
8-Bit (PTSZ<1:0> = 00)	—	PMALL	Addr	<7:0>	2 Mbytes
		PMALH	Addr<	:15:8>	7
		PMALU	Addr<	22:16>	
		_	Da	ata	
4-Bit (PTSZ<1:0> = 01)	Addr<13:12>	PMALL	Addr	<3:0>	16K
		PMALH	Addr	<7:4>	
		PMALU	Addr<	<11:8>]
		—	Da	ata	1

TABLE 20-1: EPMP FEATURE DIFFERENCES BY DEVICE PIN COUNT

21.0 REAL-TIME CLOCK AND CALENDAR (RTCC) WITH TIMESTAMP

Note: This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information on the Real-Time Clock and Calendar, refer to the *"dsPIC33/PIC24 Family Reference Manual"*, **"RTCC with Timestamp"** (DS70005193), which is available from the Microchip web site (www.microchip.com). The information in this data sheet supersedes the information in the FRM.

The RTCC provides the user with a Real-Time Clock and Calendar (RTCC) function that can be calibrated.

Key features of the RTCC module are:

- Selectable Clock Source
- Provides Hours, Minutes and Seconds Using 24-Hour Format
- · Visibility of One Half Second Period
- Provides Calendar Weekday, Date, Month and Year
- Alarm-Configurable for Half a Second, 1 Second, 10 Seconds, 1 Minute, 10 Minutes, 1 Hour, 1 Day, 1 Week, 1 Month or 1 Year
- Alarm Repeat with Decrementing Counter
- · Alarm with Indefinite Repeat Chime
- · Year 2000 to 2099 Leap Year Correction
- BCD Format for Smaller Software Overhead
- Optimized for Long-Term Battery Operation
- User Calibration of the 32.768 kHz Clock Crystal/ 32K INTRC Frequency with Periodic Auto-Adjust
- Fractional Second Synchronization
- Calibration to within ±2.64 Seconds Error per Month
- · Calibrates up to 260 ppm of Crystal Error
- Ability to Periodically Wake-up External Devices without CPU Intervention (external power control)
- Power Control Output for External Circuit Control
- Calibration takes Effect Every 15 Seconds
- Timestamp Capture register for Time and Date
- Programmable Prescaler and Clock Divider Circuit allows Operation with Any Clock Source up to 32 MHz, Including 32.768 kHz Crystal, 50/60 Hz Powerline Clock, External Real-Time Clock (RTC) or 31.25 kHz LPRC Clock

21.1 RTCC Source Clock

The RTCC clock divider block converts the incoming oscillator source into accurate 1/2 and 1 second clocks for the RTCC. The clock divider is optimized to work with three different oscillator sources:

- · 32.768 kHz crystal oscillator
- 31 kHz Low-Power RC Oscillator (LPRC)
- External 50 Hz or 60 Hz powerline frequency

An asynchronous prescaler, PS<1:0> (RTCCON2L<5:4>), is provided that allows the RTCC to work with higher speed clock sources, such as the system clock. Divide ratios of 1:16, 1:64 or 1:256 may be selected, allowing sources up to 32 MHz to clock the RTCC.

21.1.1 COARSE FREQUENCY DIVISION

The clock divider block has a 16-bit counter used to divide the input clock frequency. The divide ratio is set by the DIV<15:0> register bits (RTCCON2H<15:0>). The DIV<15:0> bits should be programmed with a value to produce a nominal 1/2 second clock divider count period.

21.1.2 FINE FREQUENCY DIVISION

The fine frequency division is set using the FDIV<4:0> (RTCCON2L<15:11>) bits. Increasing the FDIVx value will lengthen the overall clock divider period.

If FDIV<4:0> = 00000, the fine frequency division circuit is effectively disabled. Otherwise, it will optionally remove a clock pulse from the input of the clock divider every 1/2 second. This functionality will allow the user to remove up to 31 pulses over a fixed period of 16 seconds, depending on the value of FDIVx.

The value for DIV<15:0> is calculated as shown in Equation 21-1. The fractional remainder of the DIV<15:0> calculation result can be used to calculate the value for FDIV<4:0>.

EQUATION 21-1: RTCC CLOCK DIVIDER OUTPUT FREQUENCY



The DIV<15:0> value is the integer part of this calculation:

$$DIV < 15:0 > = \frac{FIN}{2 \cdot (PS < 1:0 > Prescaler)} - 1$$

The FDIV<4:0> value is the fractional part of the DIV<15:0> calculation, multiplied by 32.

x = Bit is unknown

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	DAYTEN1	DAYTEN0	DAYONE3	DAYONE2	DAYONE1	DAYONE0
bit 15							bit 8
U-0	U-0	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x
—	—	_	—	—	WDAY2	WDAY1	WDAY0
bit 7							bit 0
Legend:							
R = Readable	e bit	W = Writable	bit	U = Unimplem	nented bit, read	l as '0'	

'0' = Bit is cleared

REGISTER 21-9: DATEL: RTCC DATE REGISTER (LOW)

'1' = Bit is set

bit 15-14	Unimplemented: Read as '0'
bit 13-12	DAYTEN<1:0>: Binary Coded Decimal Value of Days '10' Digit bits
	Contains a value from 0 to 3.
bit 11-8	DAYONE<3:0>: Binary Coded Decimal Value of Days '1' Digit bits
	Contains a value from 0 to 9.
bit 7-3	Unimplemented: Read as '0'
bit 2-0	WDAY<2:0>: Binary Coded Decimal Value of Weekdays '1' Digit bits

Contains a value from 0 to 6.

-n = Value at POR

REGISTER 21-10: DATEH: RTCC DATE REGISTER (HIGH)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-x	R/W-x	R/W-x	R/W-x
YRTEN3	YRTEN2	YRTEN1	YRTEN0	YRONE3	YRONE2	YRONE1	YRONE0
bit 15							bit 8

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	MTHTEN	MTHONE3	MTHONE2	MTHONE1	MTHONE0
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	YRTEN<3:0>: Binary Coded Decimal Value of Years '10' Digit bits
-----------	---

bit 11-8 **YRONE<3:0>:** Binary Coded Decimal Value of Years '1' Digit bits

bit 7-5 Unimplemented: Read as '0'

bit 4	MTHTEN: Binary Coded Decimal Value of Months '10' Digit bit
	Contains a value from 0 to 1.

bit 3-0 MTHONE<3:0>: Binary Coded Decimal Value of Months '1' Digit bits Contains a value from 0 to 9.

22.1 User Interface

22.1.1 POLYNOMIAL INTERFACE

The CRC module can be programmed for CRC polynomials of up to the 32^{nd} order, using up to 32 bits.

Polynomial length, which reflects the highest exponent in the equation, is selected by the PLEN<4:0> bits (CRCCON2<4:0>).

The CRCXORL and CRCXORH registers control which exponent terms are included in the equation. Setting a particular bit includes that exponent term in the equation. Functionally, this includes an XOR operation on the corresponding bit in the CRC engine. Clearing the bit disables the XOR.

For example, consider two CRC polynomials, one a 16-bit and the other a 32-bit equation.

EQUATION 22-1: 16-BIT, 32-BIT CRC POLYNOMIALS

X16 + X12 + X5 + 1

and

 $\begin{array}{c} X32 + X26 + X23 + X22 + X16 + X12 + X11 + X10 + \\ X8 + X7 + X5 + X4 + X2 + X + 1 \end{array}$

To program these polynomials into the CRC generator, set the register bits, as shown in Table 22-1.

Note that the appropriate positions are set to '1' to indicate that they are used in the equation (for example, X26 and X23). The '0' bit required by the equation is always XORed; thus, X0 is a don't care. For a polynomial of length 32, it is assumed that the 32^{nd} bit will be used. Therefore, the X<31:1> bits do not have the 32^{nd} bit.

22.1.2 DATA INTERFACE

The module incorporates a FIFO that works with a variable data width. Input data width can be configured to any value between 1 and 32 bits using the DWIDTH<4:0> bits (CRCCON2<12:8>). When the data width is greater than 15, the FIFO is 4 words deep. When the DWIDTHx bits are between 15 and 8, the FIFO is 8 words deep. When the DWIDTHx bits are less than 8, the FIFO is 16 words deep.

The data for which the CRC is to be calculated must first be written into the FIFO. Even if the data width is less than 8, the smallest data element that can be written into the FIFO is 1 byte. For example, if the DWIDTHx bits are 5, then the size of the data is DWIDTH<4:0> + 1 or 6. The data is written as a whole byte; the two unused upper bits are ignored by the module.

Once data is written into the MSb of the CRCDAT registers (that is, the MSb as defined by the data width), the value of the VWORD<4:0> bits (CRCCON1<12:8>) increments by one. For example, if the DWIDTHx bits are 24, the VWORDx bits will increment when bit 7 of CRCDATH is written. Therefore, CRCDATL must always be written to before CRCDATH.

The CRC engine starts shifting data when the CRCGO bit (CRCCON1<4>) is set and the value of the VWORDx bits is greater than zero.

Each word is copied out of the FIFO into a buffer register, which decrements the VWORDx bits. The data is then shifted out of the buffer. The CRC engine continues shifting at a rate of two bits per instruction cycle, until the VWORDx bits reach zero. This means that for a given data width, it takes half that number of instructions for each word to complete the calculation. For example, it takes 16 cycles to calculate the CRC for a single word of 32-bit data.

When the VWORDx bits reach the maximum value for the configured value of the DWIDTHx bits (4, 8 or 16), the CRCFUL bit (CRCCON1<7>) becomes set. When the VWORDx bits reach zero, the CRCMPT bit (CRCCON1<6>) becomes set. The FIFO is emptied and the VWORD<4:0> bits are set to '00000' whenever CRCEN is '0'.

At least one instruction cycle must pass after a write to CRCWDAT before a read of the VWORDx bits is done.

TABLE 22-1: CRC SETUP EXAMPLES FOR 16 AND 32-BIT POLYNOMIALS

CBC Control Bito	Bit Values				
CRC Control Bits	16-Bit Polynomial	32-Bit Polynomial			
PLEN<4:0>	01111	11111			
X<31:16>	0000 0000 0000 0001	0000 0100 1100 0001			
X<15:1>	0001 0000 0010 000	0001 1101 1011 011			

R/W-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADRC ⁽¹	I) EXTSAM	PUMPEN ⁽²⁾	SAMC4	SAMC3	SAMC2	SAMC1	SAMC0
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	7 ADCS6	ADCS5	ADCS4	ADCS3	ADCS2	ADCS1	ADCS0
bit 7							bit 0
Legend:							
R = Read	able bit	W = Writable b	oit	U = Unimplen	nented bit, read	l as '0'	
-n = Value	e at POR	'1' = Bit is set		'0' = Bit is clea	ared	x = Bit is unkr	nown
bit 15	ADRC: A/D C	Conversion Cloc	k Source bit ⁽¹⁾)			
	1 = Dedicated	d ADC RC clock	generator (4	MHz nominal).			
	0 = Clock der	rived from syste	m clock				
bit 14	EXTSAM: Ex	tended Samplin	ig Time bit				
	1 = A/D is stil 0 = A/D is fini	I sampling after ished sampling	SAMP = 0				
bit 13	PUMPEN: Cr	narge Pump Ena	able bit ⁽²⁾				
	1 = Charge p	ump for switche	s is enabled				
	0 = Charge p	ump for switche	s is disabled				
bit 12-8	SAMC<4:0>:	Auto-Sample T	ime Select bit	s			
	11111 = 31 T	F ad					
	•••						
	00001 = 1 IA	ND					
hit 7 0			- Clock Soloot	hito			
DIL 7-0	ADC3<7:0>:			DILS			
	•••	250 • ICY - IAL)				
	00000001 =	2 • TCY = TAD					
	00000000 =	Tcy = Tad					
Note 1	Selecting the inte	ernal ADC RC cl	ock requires t	hat ADCSx be	1 or greater. Se	tting ADCSx =	0 when
	ADRC = 1 will vic	plate the TAD (m	in) specificatio	n.			

REGISTER 24-3: AD1CON3: A/D CONTROL REGISTER 3

2: The user should enable the charge pump if AVDD is < 2.7 V. Longer sample times are required due to the increase of the internal resistance of the MUX if the charge pump is disabled.

REGISTER 24-4: AD1CON4: A/D CONTROL REGISTER 4

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
—	—	—	—	—		DMABL<2:0>(1)
bit 7							bit 0
Legend:							
R = Readable	e 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			nown

bit 15-3 Unimplemented: Read as '0'

bit 2-0

- DMABL<2:0>: DMA Buffer Size Select bits⁽¹⁾
 - 111 = Allocates 128 words of buffer to each analog input
 - 110 = Allocates 64 words of buffer to each analog input
 - 101 = Allocates 32 words of buffer to each analog input
 - 100 = Allocates 16 words of buffer to each analog input
 - 011 = Allocates 8 words of buffer to each analog input
 - 010 = Allocates 4 words of buffer to each analog input
 - 001 = Allocates 2 words of buffer to each analog input
 - 000 = Allocates 1 word of buffer to each analog input
- **Note 1:** The DMABL<2:0> bits are only used when AD1CON1<11> = 1 and AD1CON1<12> = 0; otherwise, their value is ignored.

U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	
_		CTMEN<30:28>		—	—	CTMEN	<25:24>	
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	
			CTMEN	<23:16> (1)				
bit 7							bit 0	
Legend:								
R = Readable bit W = Writable		W = Writable bit		U = Unimplem	nented bit, read	d as '0'		
-n = Value a	at POR	'1' = Bit is set	'1' = Bit is set		'0' = Bit is cleared		x = Bit is unknown	
bit 15	Unimpleme	nted: Read as '0'						
bit 14-12	CTMEN<30	:28>: CTMU Enable	ed During C	Conversion bits				
	1 = CTMU is 0 = CTMU is	s enabled and conr s not connected to t	nected to the this channe	e selected chanı I	nel during conv	version		
bit 11-10	Unimpleme	nted: Read as '0'						
bit 9-0	CTMEN<25	:16>: CTMU Enable	ed During C	Conversion bits ⁽¹)			
	1 = CTMU is 0 = CTMU is	s enabled and conr s not connected to	nected to the this channe	e selected chani	nel during conv	version		

REGISTER 24-11: AD1CTMENH: A/D CTMU ENABLE REGISTER (HIGH WORD)

Note 1: CTMEN<23:16> bits are not available on 64-pin parts.

REGISTER 24-12: AD1CTMENL: A/D CTMU ENABLE REGISTER (LOW WORD)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
			CTM	EN<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
			CTM	/IEN<7:0>			
bit 7							bit 0
Legend:							
R = Readable	bit	W = Writable	bit	U = Unimpleme	ented bit, read	as '0'	
-n = Value at	POR	'1' = Bit is set		'0' = Bit is clea	red	x = Bit is unkr	nown
-							

bit 15-0 **CTMEN<15:0>:** CTMU Enabled During Conversion bits 1 = CTMU is enabled and connected to the selected channel during conversion 0 = CTMU is not connected to this channel

25.0 TRIPLE COMPARATOR MODULE

Note: This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the "dsPIC33/PIC24 Family Reference Manual", "Scalable Comparator Module" (DS39734), which is available from the Microchip web site (www.microchip.com). The information in this data sheet supersedes the information in the FRM.

The triple comparator module provides three dual input comparators. The inputs to the comparator can be configured to use any one of five external analog inputs (CxINA, CxINB, CxINC, CxIND and CVREF+) and a

voltage reference input from one of the internal band gap references or the comparator voltage reference generator (VBG and CVREF).

The comparator outputs may be directly connected to the CxOUT pins. When the respective COE bit equals '1', the I/O pad logic makes the unsynchronized output of the comparator available on the pin.

A simplified block diagram of the module in shown in Figure 25-1. Diagrams of the possible individual comparator configurations are shown in Figure 25-2 through Figure 25-4.

Each comparator has its own control register, CMxCON (Register 25-1), for enabling and configuring its operation. The output and event status of all three comparators is provided in the CMSTAT register (Register 25-2).



FIGURE 25-1: TRIPLE COMPARATOR MODULE BLOCK DIAGRAM

REGISTER 29-6: FWD	CONFIGURATION REGISTER
--------------------	------------------------

U-1	U-1	U-1	U-1	U-1	U-1	U-1	U-1
—	—	—	—	—	—	—	—
bit 23							bit 16

U-1	R/PO-1	R/PO-1	U-1	R/PO-1	U-1	R/PO-1	R/PO-1
—	WDTCLK1	WDTCLK0	—	WDTCMX	—	WDTWIN1	WDTWIN0
bit 15							bit 8

R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1
WINDIS	FWDTEN1	FWDTEN0	FWPSA	WDTPS3	WDTPS2	WDTPS1	WDTPS0
bit 7							bit 0

Legend:	PO = Program Once 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 23-15	Unimplemented: Read as '1'				
bit 14-13	WDTCLK<1:0>: Watchdog Timer Clock Select bits (when WDTCMX = 1)				
	11 = Always uses LPRC				
	10 = Uses FRC when WINDIS = 0, system clock is not LPRC and device is not in Sleep; otherwise, uses LPRC				
	01 = Always uses SOSC				
	00 = Uses peripheral clock when system clock is not LPRC and device is not in Sleep; otherwise, uses LPRC				
bit 12	Unimplemented: Read as '1'				
bit 11	WDTCMX: WDT Clock MUX Control bit				
	 1 = Enables WDT clock MUX, WDT clock is selected by WDTCLK<1:0> 0 = WDT clock is LPRC 				
bit 10	Unimplemented: Read as '1'				
bit 9-8	WDTWIN<1:0>: Watchdog Timer Window Width bits				
	11 = WDT window is 25% of the WDT period				
	10 = WDT window is 37.5% of the WDT period				
	01 = WDT window is 50% of the WDT period				
hit 7	WINDIS: Window is 75% of the WDT period				
	1 - Windowed WDT is disabled				
	0 = Windowed WDT is enabled				
bit 6-5	FWDTEN<1:0>: Watchdog Timer Enable bits				
	11 = WDT is enabled				
	10 = WDT is disabled (control is placed on the SWDTEN bit)				
	 01 = WDT is enabled only while device is active and disabled in Sleep; SWDTEN bit is disabled 00 = WDT and SWDTEN are disabled 				
bit 4	FWPSA: Watchdog Timer Prescaler bit				
	1 = WDT prescaler ratio of 1:128				
	0 = WDT prescaler ratio of 1:32				

30.2 MPLAB XC Compilers

The MPLAB XC Compilers are complete ANSI C compilers for all of Microchip's 8, 16, and 32-bit MCU and DSC devices. These compilers provide powerful integration capabilities, superior code optimization and ease of use. MPLAB XC Compilers run on Windows, Linux or MAC OS X.

For easy source level debugging, the compilers provide debug information that is optimized to the MPLAB X IDE.

The free MPLAB XC Compiler editions support all devices and commands, with no time or memory restrictions, and offer sufficient code optimization for most applications.

MPLAB XC Compilers include an assembler, linker and utilities. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. MPLAB XC Compiler uses the assembler to produce its object file. Notable features of the assembler include:

- · Support for the entire device instruction set
- Support for fixed-point and floating-point data
- Command-line interface
- · Rich directive set
- Flexible macro language
- MPLAB X IDE compatibility

30.3 MPASM Assembler

The MPASM Assembler is a full-featured, universal macro assembler for PIC10/12/16/18 MCUs.

The MPASM Assembler generates relocatable object files for the MPLINK Object Linker, Intel[®] standard HEX files, MAP files to detail memory usage and symbol reference, absolute LST files that contain source lines and generated machine code, and COFF files for debugging.

The MPASM Assembler features include:

- Integration into MPLAB X IDE projects
- User-defined macros to streamline assembly code
- Conditional assembly for multipurpose source files
- Directives that allow complete control over the assembly process

30.4 MPLINK Object Linker/ MPLIB Object Librarian

The MPLINK Object Linker combines relocatable objects created by the MPASM Assembler. It can link relocatable objects from precompiled libraries, using directives from a linker script.

The MPLIB Object Librarian manages the creation and modification of library files of precompiled code. When a routine from a library is called from a source file, only the modules that contain that routine will be linked in with the application. This allows large libraries to be used efficiently in many different applications.

The object linker/library features include:

- Efficient linking of single libraries instead of many smaller files
- Enhanced code maintainability by grouping related modules together
- Flexible creation of libraries with easy module listing, replacement, deletion and extraction

30.5 MPLAB Assembler, Linker and Librarian for Various Device Families

MPLAB Assembler produces relocatable machine code from symbolic assembly language for PIC24, PIC32 and dsPIC DSC devices. MPLAB XC Compiler uses the assembler to produce its object file. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. Notable features of the assembler include:

- · Support for the entire device instruction set
- · Support for fixed-point and floating-point data
- Command-line interface
- · Rich directive set
- Flexible macro language
- · MPLAB X IDE compatibility

Assembly Mnemonic		Assembly Syntax	Description	# of Words	# of Cycles	Status Flags Affected
ADD	ADD	f	f = f + WREG	1	1	C, DC, N, OV, Z
	ADD	f,WREG	WREG = f + WREG	1	1	C, DC, N, OV, Z
	ADD	#lit10,Wn	Wd = lit10 + Wd	1	1	C, DC, N, OV, Z
	ADD	Wb,Ws,Wd	Wd = Wb + Ws	1	1	C, DC, N, OV, Z
	ADD	Wb,#lit5,Wd	Wd = Wb + lit5	1	1	C, DC, N, OV, Z
ADDC	ADDC	f	f = f + WREG + (C)	1	1	C, DC, N, OV, Z
	ADDC	f,WREG	WREG = f + WREG + (C)	1	1	C, DC, N, OV, Z
	ADDC	#lit10,Wn	Wd = lit10 + Wd + (C)	1	1	C, DC, N, OV, Z
	ADDC	Wb,Ws,Wd	Wd = Wb + Ws + (C)	1	1	C, DC, N, OV, Z
	ADDC	Wb,#lit5,Wd	Wd = Wb + lit5 + (C)	1	1	C, DC, N, OV, Z
AND	AND	f	f = f .AND. WREG	1	1	N, Z
	AND	f,WREG	WREG = f .AND. WREG	1	1	N, Z
	AND	#lit10,Wn	Wd = lit10 .AND. Wd	1	1	N, Z
	AND	Wb,Ws,Wd	Wd = Wb .AND. Ws	1	1	N, Z
	AND	Wb,#lit5,Wd	Wd = Wb .AND. lit5	1	1	N, Z
ASR	ASR	f	f = Arithmetic Right Shift f	1	1	C, N, OV, Z
	ASR	f,WREG	WREG = Arithmetic Right Shift f	1	1	C, N, OV, Z
	ASR	Ws,Wd	Wd = Arithmetic Right Shift Ws	1	1	C, N, OV, Z
	ASR	Wb,Wns,Wnd	Wnd = Arithmetic Right Shift Wb by Wns	1	1	N, Z
	ASR	Wb,#lit5,Wnd	Wnd = Arithmetic Right Shift Wb by lit5	1	1	N, Z
BCLR	BCLR	f,#bit4	Bit Clear f	1	1	None
	BCLR	Ws,#bit4	Bit Clear Ws	1	1	None
BRA	BRA	C,Expr	Branch if Carry	1	1 (2)	None
	BRA	GE,Expr	Branch if Greater than or Equal	1	1 (2)	None
	BRA	GEU, Expr	Branch if Unsigned Greater than or Equal	1	1 (2)	None
	BRA	GT,Expr	Branch if Greater than	1	1 (2)	None
	BRA	GTU, Expr	Branch if Unsigned Greater than	1	1 (2)	None
	BRA	LE,Expr	Branch if Less than or Equal	1	1 (2)	None
	BRA	LEU,Expr	Branch if Unsigned Less than or Equal	1	1 (2)	None
	BRA	LT,Expr	Branch if Less than	1	1 (2)	None
	BRA	LTU, Expr	Branch if Unsigned Less than	1	1 (2)	None
	BRA	N,Expr	Branch if Negative	1	1 (2)	None
	BRA	NC, Expr	Branch if Not Carry	1	1 (2)	None
	BRA	NN,Expr	Branch if Not Negative	1	1 (2)	None
	BRA	NOV, Expr	Branch if Not Overflow	1	1 (2)	None
	BRA	NZ,Expr	Branch if Not Zero	1	1 (2)	None
	BRA	OV,Expr	Branch if Overflow	1	1 (2)	None
	BRA	Expr	Branch Unconditionally	1	2	None
	BRA	Z,Expr	Branch if Zero	1	1 (2)	None
	BRA	Wn	Computed Branch	1	2	None
BSET	BSET	f,#bit4	Bit Set f	1	1	None
	BSET	Ws,#bit4	Bit Set Ws	1	1	None
BSW	BSW.C	Ws,Wb	Write C bit to Ws <wb></wb>	1	1	None
	BSW.Z	Ws,Wb	Write Z bit to Ws <wb></wb>	1	1	None
BTG	BTG	f,#bit4	Bit Toggle f	1	1	None
	BTG	Ws,#bit4	Bit Toggle Ws	1	1	None
BTSC	BTSC	f,#bit4	Bit Test f, Skip if Clear	1	1 (2 or 3)	None
	BTSC	Ws,#bit4	Bit Test Ws, Skip if Clear	1	1 (2 or 3)	None

TABLE 31-2: INSTRUCTION SET OVERVIEW