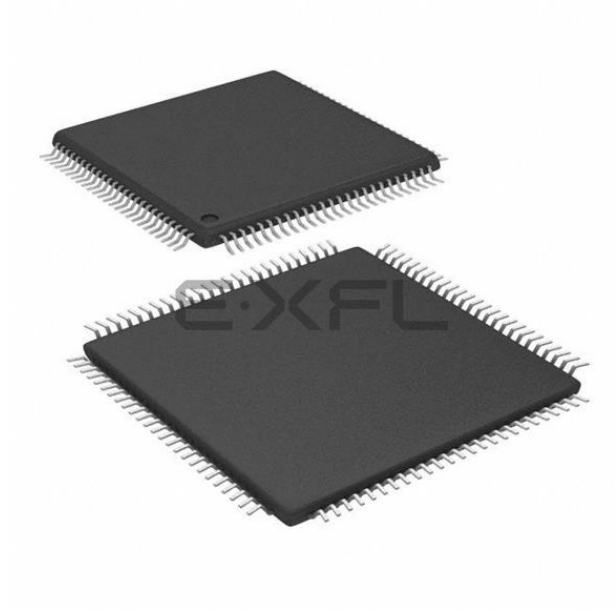
Microchip Technology - PIC24FJ128GB210-I/PT Datasheet



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 "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	PIC
Core Size	16-Bit
Speed	32MHz
Connectivity	I²C, IrDA, SPI, UART/USART, USB OTG
Peripherals	Brown-out Detect/Reset, LVD, POR, PWM, WDT
Number of I/O	84
Program Memory Size	128KB (43K x 24)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	96K x 8
Voltage - Supply (Vcc/Vdd)	2.2V ~ 3.6V
Data Converters	A/D 24x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	100-TQFP
Supplier Device Package	100-TQFP (12x12)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic24fj128gb210-i-pt

TABLE 2: COMPLETE PIN FUNCTION DESCRIPTIONS FOR 100-PIN DEVICES

Pin	Function	Pin	Function
1	CN82/RG15	41	AN12/PMA11/CTEDG2/CN30/RB12
2	VDD	42	AN13/PMA10/CTEDG1/CN31/RB13
3	PMD5/CN63/RE5	43	AN14/CTPLS/**RP14**/PMA1/CN32/RB14
4	SCL3/PMD6/CN64/RE6	44	AN15/REFO/**RP29**/PMA0/CN12/RB15
5	SDA3/PMD7/CN65/RE7	45	VSS
6	**RPI38**/CN45/RC1	46	VDD
7	**RPI39**/CN46/RC2	47	**RPI43**/CN20/RD14
8	**RPI40**/CN47/RC3	48	**RP5**/CN21/RD15
9	AN16/**RPI41**/PMCS2/PMA22(2)/CN48/RC4	49	**RP10**/PMA9/CN17/RF4
10	AN17/C1IND/**RP21**/PMA5/PMA18(2)/CN8/RG6	50	**RP17**/PMA8/CN18/RF5
11	AN18/C1INC/**RP26**/PMA4/PMA20(2)/CN9/RG7	51	**RP16**/USBID/CN71/RF3
12	AN19/C2IND/**RP19**/PMA3/PMA21(2)/CN10/RG8	52	**RP30**/CN70/RF2
13	$\overline{\text{MCLR}}$	53	**RP15**/CN74/RF8
14	AN20/C2INC/**RP27**/PMA2/CN11/RG9	54	VBUS/CN73/RF7
15	VSS	55	VUSB
16	VDD	56	D-/CN84/RG3
17	TMS/CN33/RA0	57	D+/CN83/RG2
18	**RPI33**/PMCS1/CN66/RE8	58	SCL2/CN35/RA2
19	AN21/**RPI34**/PMA19/CN67/RE9	59	SDA2/PMA20/PMA4(2)/CN36/RA3
20	PGEC3/AN5/C1INA/VBUSON/**RP18**/CN7/RB5	60	TDI/PMA21/PMA3(2)/CN37/RA4
21	PGED3/AN4/C1INB/USBOEN/**RP28**/CN6/RB4	61	TDO/CN38/RA5
22	AN3/C2INA/VPIO/CN5/RB3	62	VDD
23	AN2/C2INB/VMIO/**RP13**/CN4/RB2	63	OSCI/CLKI/CN23/RC12
24	PGEC1/AN1/VREF-(1)/**RP1**/CN3/RB1	64	OSCO/CLKO/CN22/RC15
25	PGED1/AN0/VREF+(1)/**RP0**/CN2/RB0	65	VSS
26	PGEC2/AN6/**RP6**/CN24/RB6	66	SCL1/**RPI36**/PMA22/PMCS2(2)/CN43/RA14
27	PGED2/AN7/**RP7**/RCV/CN25/RB7	67	SDA1/**RPI35**/PMBE1/CN44/RA15
28	VREF-/PMA7/CN41/RA9	68	DMLN/RTCC/**RP2**/CN53/RD8
29	VREF+/PMA6/CN42/RA10	69	DPLN/**RP4**/PMACK2/CN54/RD9
30	AVDD	70	**RP3**/PMA15/PMCS2(3)/CN55/RD10
31	AVSS	71	**RP12**/PMA14/PMCS1(3)/CN56/RD11
32	AN8/**RP8**/CN26/RB8	72	DMH/**RP11**/INT0/CN49/RD0
33	AN9/**RP9**/CN27/RB9	73	SOSCI/C3IND/CN1/RC13
34	AN10/CVREF/PMA13/CN28/RB10	74	SOSCO/SCLKI/T1CK/C3INC/**RPI37**/CN0/RC14
35	AN11/PMA12/CN29/RB11	75	VSS
36	VSS	76	VCPCON/**RP24**/VBUSCHG/CN50/RD1
37	VDD	77	DPH/**RP23**/PMACK1/CN51/RD2
38	TCK/CN34/RA1	78	**RP22**/PMBE0/CN52/RD3
39	**RP31**/CN76/RF13	79	**RPI42**/PMD12/CN57/RD12
40	**RPI32**/PMA18/PMA5(2)/CN75/RF12	80	PMD13/CN19/RD13

Legend: **RPn** and **RPIn** represent remappable pins for Peripheral Pin Select (PPS) functions.

Note 1: Alternate pin assignments for VREF+ and VREF- when the $\overline{\text{ALTVREF}}$ Configuration bit is programmed.

2: Alternate pin assignments for EPMP when the $\overline{\text{ALTPMP}}$ Configuration bit is programmed (only in 100-pin devices).

3: Pin assignment for PMCSx when CSF<1:0> is not equal to '00'.

TABLE 1-3: PIC24FJ256GB210 FAMILY PINOUT DESCRIPTIONS (CONTINUED)

Function	Pin Number			I/O	Input Buffer	Description
	64-Pin TQFP/QFN	100-Pin TQFP	121-Pin BGA			
RP20	53	82	B8	I/O	ST	Remappable Peripheral (input or output).
RP21	4	10	E3	I/O	ST	
RP22	51	78	B9	I/O	ST	
RP23	50	77	A10	I/O	ST	
RP24	49	76	A11	I/O	ST	
RP25	52	81	C8	I/O	ST	
RP26	5	11	F4	I/O	ST	
RP27	8	14	F3	I/O	ST	
RP28	12	21	H2	I/O	ST	
RP29	30	44	L8	I/O	ST	
RP30	—	52	K11	I/O	ST	
RP31	—	39	L6	I/O	ST	
RPI32	—	40	K6	I	ST	Remappable Peripheral (input only).
RPI33	—	18	G1	I	ST	
RPI34	—	19	G2	I	ST	
RPI35	—	67	E8	I	ST	
RPI36	—	66	E11	I	ST	
RPI37	48	74	B11	I	ST	
RPI38	—	6	D1	I	ST	
RPI39	—	7	E4	I	ST	
RPI40	—	8	E2	I	ST	
RPI41	—	9	E1	I	ST	
RPI42	—	79	A9	I	ST	
RPI43	—	47	L9	I	ST	
RTCC	42	68	E9	O	—	Real-Time Clock Alarm/Seconds Pulse Output.
SCL1	44	66	E11	I/O	I^2C™	I2C1 Synchronous Serial Clock Input/Output.
SCL2	32	58	H11	I/O	I^2C	I2C2 Synchronous Serial Clock Input/Output.
SCL3	2	4	C1	I/O	I^2C	I2C3 Synchronous Serial Clock Input/Output.
SCLKI	48	74	B11	O	ANA	Secondary Clock Input.
SDA1	43	67	E8	I/O	I^2C	I2C1 Data Input/Output.
SDA2	31	59	G10	I/O	I^2C	I2C2 Data Input/Output.
SDA3	3	5	D2	I/O	I^2C	I2C3 Data Input/Output.
SESSEND	55	84	C7	I	ST	USB VBUS Boost Generator, Comparator Input 3.
SESSVLD	59	88	A6	I	ST	USB VBUS Boost Generator, Comparator Input 2.
SOSCI	47	73	C10	I	ANA	Secondary Oscillator/Timer1 Clock Input.
SOSCO	48	74	B11	O	ANA	Secondary Oscillator/Timer1 Clock Output.
T1CK	48	74	B11	I	ST	Timer1 Clock.

Legend: TTL = TTL input buffer; ST = Schmitt Trigger input buffer; ANA = Analog level input/output; I^2C™ = I^2C/SMBus input buffer

Note 1: The alternate EPMP pins are selected when the $\overline{\text{ALTPMP}}$ (CW3<12>) bit is programmed to '0'.

2: The PMSC2 signal will replace the PMA15 signal on the 15-pin PMA when CSF<1:0> = 01 or 10.

3: The PMCS1 signal will replace the PMA14 signal on the 14-pin PMA when CSF<1:0> = 10.

4: The alternate VREF pins selected when the $\overline{\text{ALTVREF}}$ (CW1<5>) bit is programmed to '0'.

TABLE 4-28: COMPARATORS 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
CMSTAT	0630	CMIDL	—	—	—		C3EVT	C2EVT	C1EVT	—	—	—	—	—	C3OUT	C2OUT	C1OUT	0000
CVRCON	0632	—	—	—	—	—	CVREFP	CVREFM1	CVREFM0	CVREN	CVROE	CVRR	CVRSS	CVR3	CVR2	CVR1	CVR0	0000
CM1CON	0634	CON	COE	CPOL	—	—	—	CEVT	COUT	EVPOL1	EVPOL0	—	CREF	—	—	CCH1	CCH0	0000
CM2CON	0636	CON	COE	CPOL	—	—	—	CEVT	COUT	EVPOL1	EVPOL0	—	CREF	—	—	CCH1	CCH0	0000
CM3CON	0638	CON	COE	CPOL	—	—	—	CEVT	COUT	EVPOL1	EVPOL0	—	CREF	—	—	CCH1	CCH0	0000

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

TABLE 4-29: 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	VWORD4	VWORD3	VWORD2	VWORD1	VWORD0	CRCFUL	CRCMPT	CRCISEL	CRCGO	LENDIAN	—	—	—	0040
CRCCON2	0642	—	—	—	DWIDTH4	DWIDTH3	DWIDTH2	DWIDTH1	DWIDTH0	—	—	—	PLEN4	PLEN3	PLEN2	PLEN1	PLEN0	0000
CRCXORL	0644	X15	X14	X13	X12	X11	X10	X9	X8	X7	X6	X5	X4	X3	X2	X1	—	0000
CRCXORH	0646	X31	X30	X29	X28	X27	X26	X25	X24	X23	X22	X21	X20	X19	X18	X17	X16	0000
CRCDATL	0648	CRC Data Input Register Low																0000
CRCDATH	064A	CRC Data Input Register High																0000
CRCWDATL	064C	CRC Result Register Low																0000
CRCWDATH	064E	CRC Result Register High																0000

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

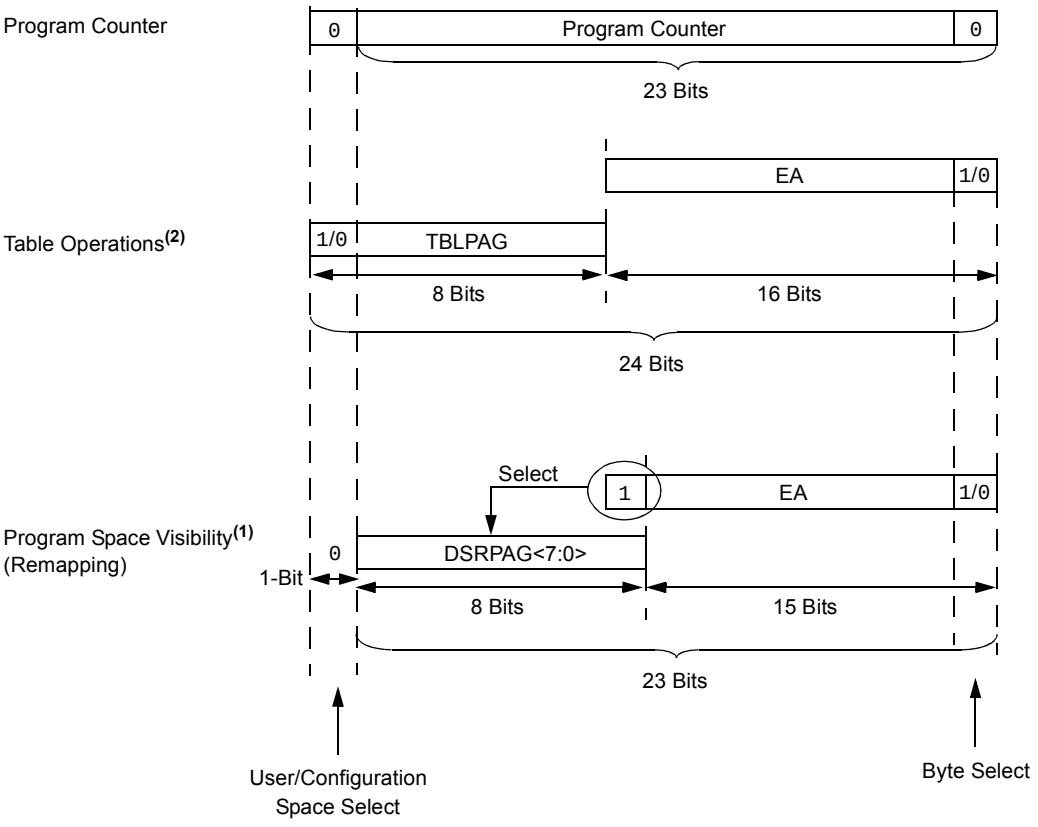
TABLE 4-35: PROGRAM SPACE ADDRESS CONSTRUCTION

Access Type	Access Space	Program Space Address				
		<23>	<22:16>	<15>	<14:1>	<0>
Instruction Access (Code Execution)	User	0	PC<22:1>			0
		0xx xxxx xxxx xxxx xxxx xxx0				
TBLRD/TBLWT (Byte/Word Read/Write)	User	TBLPAG<7:0>		Data EA<15:0>		
		0xxx xxxx		xxxx xxxx xxxx xxxx		
	Configuration	TBLPAG<7:0>		Data EA<15:0>		
		1xxx xxxx		xxxx xxxx xxxx xxxx		
Program Space Visibility (Block Remap/Read)	User	0	DSRPAG<7:0>(2)		Data EA<14:0>(1)	
		0	xxxx xxxx		xxx xxxx xxxx xxxx	

Note 1: Data EA<15> is always '1' in this case, but is not used in calculating the program space address. Bit 15 of the address is DSRPAG<0>.

2: DSRPAG<9> is always '1' in this case. DSRPAG<8> decides whether the lower word or higher word of program memory is read. When DSRPAG<8> is '0', the lower word is read and when it is '1', the higher word is read.

FIGURE 4-8: DATA ACCESS FROM PROGRAM SPACE ADDRESS GENERATION



Note 1: DSRPAG<8> acts as word select. DSRPAG<9> should always be '1' to map program memory to data memory.

2: The instructions, TBLRDH/TBLWTH/TBLRDL/TBLWTL, decide if the higher or lower word of program memory is accessed. TBLRDH/TBLWTH instructions access the higher word and TBLRDL/TBLWTL instructions access the lower word. Table read operations are permitted in the configuration memory space.

REGISTER 6-1: RCON: RESET CONTROL REGISTER(1)

R/W-0, HS	R/W-0, HS	U-0	U-0	U-0	U-0	R/W-0, HS	R/W-0
TRAPR	IOPUWR	—	—	—	—	CM	VREGS(3)
bit 15							bit 8

R/W-0, HS	R/W-0, HS	R/W-0, HS	R/W-0, HS	R/W-0, HS	R/W-0, HS	R/W-1, HS	R/W-1, HS
EXTR	SWR	SWDTEN(2)	WDTO	SLEEP	IDLE	BOR	POR
bit 7							bit 0

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

bit 15 **TRAPR:** Trap Reset Flag bit
1 = A Trap Conflict Reset has occurred
0 = A Trap Conflict Reset has not occurred

bit 14 **IOPUWR:** Illegal Opcode or Uninitialized W Access Reset Flag bit
1 = An illegal opcode detection, an illegal address mode or uninitialized W register is used as an Address Pointer and caused a Reset
0 = An illegal opcode or uninitialized W Reset has not occurred

bit 13-10 **Unimplemented:** Read as '0'

bit 9 **CM:** Configuration Word Mismatch Reset Flag bit
1 = A Configuration Word Mismatch Reset has occurred
0 = A Configuration Word Mismatch Reset has not occurred

bit 8 **VREGS:** Voltage Regulator Standby Enable bit(3)
1 = Program memory and regulator remain active during Sleep/Idle
0 = Program memory power is removed and regulator goes to standby during Seep/Idle

bit 7 **EXTR:** External Reset ($\overline{\text{MCLR}}$) Pin bit
1 = A Master Clear (pin) Reset has occurred
0 = A Master Clear (pin) Reset has not occurred

bit 6 **SWR:** Software Reset (Instruction) Flag bit
1 = A RESET instruction has been executed
0 = A RESET instruction has not been executed

bit 5 **SWDTEN:** Software Enable/Disable of WDT bit(2)
1 = WDT is enabled
0 = WDT is disabled

bit 4 **WDTO:** Watchdog Timer Time-out Flag bit
1 = WDT time-out has occurred
0 = WDT time-out has not occurred

bit 3 **SLEEP:** Wake From Sleep Flag bit
1 = Device has been in Sleep mode
0 = Device has not been in Sleep mode

Note 1: All of the Reset status bits may be set or cleared in software. Setting one of these bits in software does not cause a device Reset.

2: If the FWDTEN Configuration bit is '1' (unprogrammed), the WDT is always enabled**,** regardless of the SWDTEN bit setting.

3: Re-enabling the regulator after it enters Standby mode will add a delay, T_{VREG}, when waking up from Sleep. Applications that do not use the voltage regulator should set this bit to prevent this delay from occurring.

REGISTER 6-1: RCON: RESET CONTROL REGISTER(1) (CONTINUED)

bit 2 **IDLE:** Wake-up From Idle Flag bit
1 = Device has been in Idle mode
0 = Device has not been in Idle mode

bit 1 **BOR:** Brown-out Reset Flag bit
1 = A Brown-out Reset has occurred
Note that BOR is also set after a Power-on Reset.
0 = A Brown-out Reset has not occurred

bit 0 **POR:** Power-on Reset Flag bit
1 = A Power-on Reset has occurred
0 = A Power-on Reset has not occurred

Note 1: All of the Reset status bits may be set or cleared in software. Setting one of these bits in software does not cause a device Reset.

2: If the FWDTEN Configuration bit is '1' (unprogrammed), the WDT is always enabled, regardless of the SWDTEN bit setting.

3: Re-enabling the regulator after it enters Standby mode will add a delay, TVREG, when waking up from Sleep. Applications that do not use the voltage regulator should set this bit to prevent this delay from occurring.

TABLE 6-1: RESET FLAG BIT OPERATION

Flag Bit	Setting Event	Clearing Event
TRAPR (RCON<15>)	Trap Conflict Event	POR
IOPUWR (RCON<14>)	Illegal Opcode or Uninitialized W Register Access	POR
CM (RCON<9>)	Configuration Mismatch Reset	POR
EXTR (RCON<7>)	$\overline{MCLR}$ Reset	POR
SWR (RCON<6>)	RESET Instruction	POR
WDTO (RCON<4>)	WDT Time-out	CLRWDT, PWRSAV Instruction, POR
SLEEP (RCON<3>)	PWRSAV #0 Instruction	POR
IDLE (RCON<2>)	PWRSAV #1 Instruction	POR
BOR (RCON<1>)	POR, BOR	—
POR (RCON<0>)	POR	—

Note: All Reset flag bits may be set or cleared by the user software.

TABLE 6-3: RESET DELAY TIMES FOR VARIOUS DEVICE RESETS

Reset Type	Clock Source	$\overline{\text{SYSRST}}$ Delay	System Clock Delay	Notes
POR(7)	EC	TPOR + TSTARTUP + TRST	—	1, 2, 3
	ECPLL	TPOR + TSTARTUP + TRST	TLOCK	1, 2, 3, 5
	XT, HS, SOSC	TPOR + TSTARTUP + TRST	TOST	1, 2, 3, 4
	XTPLL, HSPLL	TPOR + TSTARTUP + TRST	TOST + TLOCK	1, 2, 3, 4, 5
	FRC, FRCDIV	TPOR + TSTARTUP + TRST	TFRC	1, 2, 3, 6, 7
	FRCPLL	TPOR + TSTARTUP + TRST	TFRC + TLOCK	1, 2, 3, 5, 6
	LPRC	TPOR + TSTARTUP + TRST	TLPRC	1, 2, 3, 6
BOR	EC	TSTARTUP + TRST	—	2, 3
	ECPLL	TSTARTUP + TRST	TLOCK	2, 3, 5
	XT, HS, SOSC	TSTARTUP + TRST	TOST	2, 3, 4
	XTPLL, HSPLL	TSTARTUP + TRST	TOST + TLOCK	2, 3, 4, 5
	FRC, FRCDIV	TSTARTUP + TRST	TFRC	2, 3, 6, 7
	FRCPLL	TSTARTUP + TRST	TFRC + TLOCK	2, 3, 5, 6
	LPRC	TSTARTUP + TRST	TLPRC	2, 3, 6
$\overline{\text{MCLR}}$	Any Clock	TRST	—	3
WDT	Any Clock	TRST	—	3
Software	Any clock	TRST	—	3
Illegal Opcode	Any Clock	TRST	—	3
Uninitialized W	Any Clock	TRST	—	3
Trap Conflict	Any Clock	TRST	—	3

Note 1: TPOR = Power-on Reset delay (10 μs nominal).

2: TSTARTUP = TVREG (10 μs nominal when VREGS = 1 and when VREGS = 0; depends upon WUTSEL<1:0> bits setting).

3: TRST = Internal State Reset time (32 μs nominal).

4: TOST = Oscillator Start-up Timer. A 10-bit counter counts 1024 oscillator periods before releasing the oscillator clock to the system.

5: TLOCK = PLL lock time.

6: TFRC and TLPRC = RC Oscillator start-up times.

7: If Two-speed Start-up is enabled, regardless of the primary oscillator selected, the device starts with FRC so the system clock delay is just TFRC, and in such cases, FRC start-up time is valid. It switches to the primary oscillator after its respective clock delay.

REGISTER 7-20: IPC3: INTERRUPT PRIORITY CONTROL REGISTER 3

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

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	AD1IP2	AD1IP1	AD1IP0	—	U1TXIP2	U1TXIP1	U1TXIP0
bit 7							bit 0

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

bit 15-7 **Unimplemented:** Read as '0'

bit 6-4 **AD1IP<2:0>:** A/D Conversion Complete Interrupt Priority bits
111 = Interrupt is priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is priority 1
000 = Interrupt source is disabled

bit 3 **Unimplemented:** Read as '0'

bit 2-0 **U1TXIP<2:0>:** UART1 Transmitter Interrupt Priority bits
111 = Interrupt is priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is priority 1
000 = Interrupt source is disabled

REGISTER 7-31: IPC15: INTERRUPT PRIORITY CONTROL REGISTER 15

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	RTCIP2	RTCIP1	RTCIP0
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7							bit 0

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

bit 15-11 **Unimplemented:** Read as '0'

bit 10-8 **RTCIP<2:0>:** Real-Time Clock and Calendar Interrupt Priority bits

111 = Interrupt is priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is priority 1
000 = Interrupt source is disabled

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

REGISTER 7-39: INTTREG: INTERRUPT CONTROLLER TEST REGISTER

R-0, HSC	U-0	R/W-0	U-0	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC
CPUIRQ	—	VHOLD	—	ILR3	ILR2	ILR1	ILR0
bit 15							bit 8

U-0	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC
—	VECNUM6	VECNUM5	VECNUM4	VECNUM3	VECNUM2	VECNUM1	VECNUM0
bit 7							bit 0

Legend:	HSC = Hardware Settable/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 **CPUIRQ:** Interrupt Request from Interrupt Controller CPU bit
1 = An interrupt request has occurred but has not yet been Acknowledged by the CPU; this happens when the CPU priority is higher than the interrupt priority
0 = No interrupt request is unacknowledged

bit 14 **Unimplemented:** Read as '0'

bit 13 **VHOLD:** Vector Number Capture Configuration bit
1 = The VECNUM bits contain the value of the highest priority pending interrupt
0 = The VECNUM bits contain the value of the last Acknowledged interrupt (i.e., the last interrupt that has occurred with higher priority than the CPU, even if other interrupts are pending)

bit 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<5:0>:** Vector Number of Pending Interrupt or Last Acknowledged Interrupt bits
VHOLD = 1: The VECNUM bits indicate the vector number (from 0 to 118) of the last interrupt to occur
VHOLD = 0: The VECNUM bits indicate the vector number (from 0 to 118) of the interrupt request currently being handled

8.1 CPU Clocking Scheme

The system clock source can be provided by one of four sources:

- Primary Oscillator (POSC) on the OSCI and OSCO pins
- Secondary Oscillator (SOSC) on the SOSCI and SOSCO pins
- Fast Internal RC (FRC) Oscillator
- Low-Power Internal RC (LPRC) Oscillator

The primary oscillator and FRC sources have the option of using the internal 24x PLL block, which generates the USB module clock, and a separate system clock through the 96 MHZ PLL. Refer to **Section 8.5 "96 MHz PLL Block"** for additional information.

The internal FRC provides an 8 MHz clock source. It can optionally be reduced by the programmable clock divider to provide a range of system clock frequencies.

The selected clock source generates the processor and peripheral clock sources. The processor clock source is divided by two to produce the internal instruction cycle clock, FCY. In this document, the instruction cycle clock is also denoted by FOSC/2. The internal instruction cycle clock, FOSC/2, can be provided on the OSCO I/O pin for some operating modes of the primary oscillator.

8.2 Initial Configuration on POR

The oscillator source (and operating mode) that is used at a device Power-on Reset (POR) event is selected using Configuration bit settings. The oscillator Configuration bit settings are located in the Configuration registers in the program memory (refer to **Section 26.1 "Configuration Bits"** for further details). The Primary Oscillator Configuration bits, POSCMD<1:0> (Configuration Word 2<1:0>) and the Initial Oscillator Select Configuration bits, FNOSC<2:0> (Configuration Word 2<10:8>), select the oscillator source that is used at a POR. The FRC primary Oscillator with Postscaler (FRCDIV) is the default (unprogrammed) selection. The secondary oscillator, or one of the internal oscillators, may be chosen by programming these bit locations.

The Configuration bits allow users to choose between the various clock modes, shown in Table 8-1.

8.2.1 CLOCK SWITCHING MODE CONFIGURATION BITS

The FCKSM Configuration bits (Configuration Word 2<7:6>) are used to jointly configure device clock switching and the Fail-Safe Clock Monitor (FSCM). Clock switching is enabled only when FCKSM1 is programmed ('0'). The FSCM is enabled only when FCKSM<1:0> are both programmed ('00').

TABLE 8-1: CONFIGURATION BIT VALUES FOR CLOCK SELECTION

Oscillator Mode	Oscillator Source	POSCMD<1:0>	FNOSC<2:0>	Notes
Fast RC Oscillator with Postscaler (FRCDIV)	Internal	11	111	**1, 2**
FRC Oscillator/16 (500 KHz)	Internal	11	110	**1**
Low-Power RC Oscillator (LPRC)	Internal	11	101	**1**
Secondary (Timer1) Oscillator (SOSC)	Secondary	11	100	**1**
Primary Oscillator (XT) with PLL Module (XTPLL)	Primary	01	011	**—**
Primary Oscillator (EC) with PLL Module (ECPLL)	Primary	00	011	**1**
Primary Oscillator (HS)	Primary	10	010	**—**
Primary Oscillator (XT)	Primary	01	010	**—**
Primary Oscillator (EC)	Primary	00	010	**1**
Fast RC Oscillator with PLL Module (FRCPLL)	Internal	11	001	**1**
Fast RC Oscillator (FRC)	Internal	11	000	**1**

Note 1: OSCO pin function is determined by the OSCIOFCN Configuration bit.

2: This is the default oscillator mode for an unprogrammed (erased) device.

13.1.2 CASCADED (32-BIT) MODE

By default, each module operates independently with its own 16-bit timer. To increase resolution, adjacent even and odd modules can be configured to function as a single 32-bit module. (For example, Modules 1 and 2 are paired, as are Modules 3 and 4, and so on.) The odd numbered module (ICx) provides the Least Significant 16 bits of the 32-bit register pairs and the even module (ICy) provides the Most Significant 16 bits. Wrap-arounds of the ICx registers cause an increment of their corresponding ICy registers.

Cascaded operation is configured in hardware by setting the IC32 bits (ICxCON2<8>) for both modules.

13.2 Capture Operations

The input capture module can be configured to capture timer values and generate interrupts on rising edges on ICx or all transitions on ICx. Captures can be configured to occur on all rising edges or just some (every 4th or 16th). Interrupts can be independently configured to generate on each event or a subset of events.

To set up the module for capture operations:

1. Configure the ICx input for one of the available Peripheral Pin Select pins.
2. If Synchronous mode is to be used, disable the sync source before proceeding.
3. Make sure that any previous data has been removed from the FIFO by reading ICxBUF until the ICBNE bit (ICxCON1<3>) is cleared.
4. Set the SYNCSEL bits (ICxCON2<4:0>) to the desired sync/trigger source.
5. Set the ICTSEL bits (ICxCON1<12:10>) for the desired clock source.
6. Set the ICI bits (ICxCON1<6:5>) to the desired interrupt frequency
7. Select Synchronous or Trigger mode operation:
   a) Check that the SYNCSEL bits are not set to '00000'.
   b) For Synchronous mode, clear the ICTRIG bit (ICxCON2<7>).
   c) For Trigger mode, set ICTRIG, and clear the TRIGSTAT bit (ICxCON2<6>).
8. Set the ICM bits (ICxCON1<2:0>) to the desired operational mode.
9. Enable the selected sync/trigger source.

For 32-bit cascaded operations, the setup procedure is slightly different:

1. Set the IC32 bits for both modules (ICyCON2<8>) and (ICxCON2<8>), enabling the even numbered module first. This ensures the modules will start functioning in unison.
2. Set the ICTSEL and SYNCSEL bits for both modules to select the same sync/trigger and time base source. Set the even module first, then the odd module. Both modules must use the same ICTSEL and SYNCSEL settings.
3. Clear the ICTRIG bit of the even module (ICyCON2<7>). This forces the module to run in Synchronous mode with the odd module, regardless of its trigger setting.
4. Use the odd module's ICI bits (ICxCON1<6:5>) to set the desired interrupt frequency.
5. Use the ICTRIG bit of the odd module (ICxCON2<7>) to configure Trigger or Synchronous mode operation.

Note:	For Synchronous mode operation, enable the sync source as the last step. Both input capture modules are held in Reset until the sync source is enabled.

6. Use the ICM bits of the odd module (ICxCON1<2:0>) to set the desired capture mode.

The module is ready to capture events when the time base and the sync/trigger source are enabled. When the ICBNE bit (ICxCON1<3>) becomes set, at least one capture value is available in the FIFO. Read input capture values from the FIFO until the ICBNE clears to '0'.

For 32-bit operation, read both the ICxBUF and ICyBUF for the full 32-bit timer value (ICxBUF for the lsw, ICyBUF for the msw). At least one capture value is available in the FIFO buffer when the odd module's ICBNE bit (ICxCON1<3>) becomes set. Continue to read the buffer registers until ICBNE is cleared (performed automatically by hardware).

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.
2. Clear the OCTRIG bit of the even module (OCyCON2) 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.
5. If Trigger mode operation is required, configure the trigger options in OCx by using the OCTRIG (OCxCON2<7>), TRIGMODE (OCxCON1<3>) and SYNCSEL (OCxCON2<4:0>) bits.
6. 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.

14.3 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 double-buffered (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.
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.
4. 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 (OCxCON<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.
7. Select the desired PWM mode in the OCM<2:0> bits (OCxCON1<2:0>).
8. Appropriate Fault inputs may be enabled by using the ENFLT<2:0> bits as described in Register 14-1.
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 10.4 "Peripheral Pin Select (PPS)"** for more information.

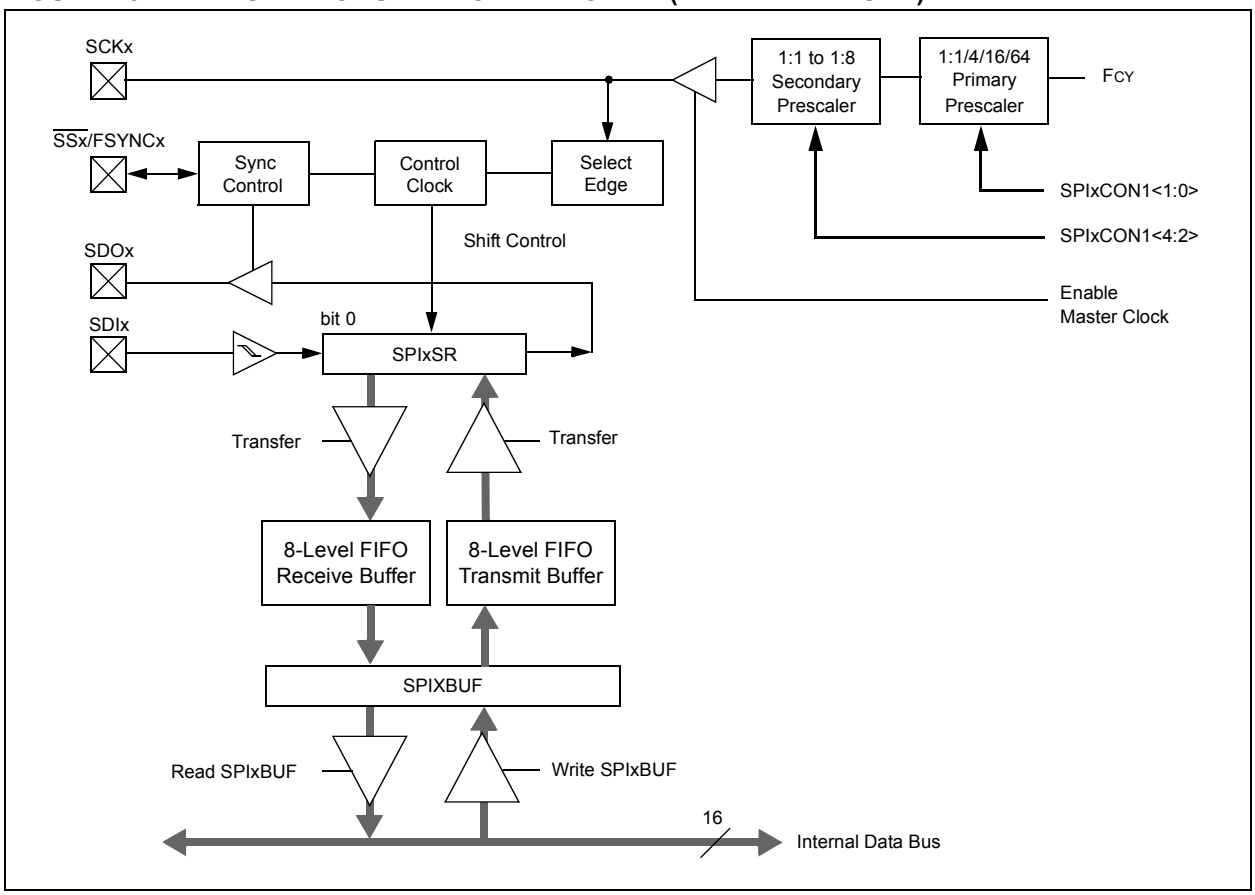
To set up the SPI module for the Enhanced Buffer Master mode of operation:

1. If using interrupts:
   a) Clear the SPIxIF bit in the respective IFS register.
   b) Set the SPIxIE bit in the respective IEC register.
   c) Write the SPIxIP bits in the respective IPC register.
2. Write the desired settings to the SPIxCON1 and SPIxCON2 registers with MSTEN (SPIxCON1<5>) = 1.
3. Clear the SPIROV bit (SPIxSTAT<6>).
4. Select Enhanced Buffer mode by setting the SPIBEN bit (SPIxCON2<0>).
5. Enable SPI operation by setting the SPIEN bit (SPIxSTAT<15>).
6. Write the data to be transmitted to the SPIxBUF register. Transmission (and reception) will start as soon as data is written to the SPIxBUF register.

To set up the SPI module for the Enhanced Buffer Slave mode of operation:

1. Clear the SPIxBUF register.
2. If using interrupts:
   a) Clear the SPIxIF bit in the respective IFS register.
   b) Set the SPIxIE bit in the respective IEC register.
   c) Write the SPIxIP bits in the respective IPC register to set the interrupt priority.
3. Write the desired settings to the SPIxCON1 and SPIxCON2 registers with MSTEN (SPIxCON1<5>) = 0.
4. Clear the SMP bit.
5. If the CKE bit is set, then the SSEN bit must be set, thus enabling the $\overline{SSx}$ pin.
6. Clear the SPIROV bit (SPIxSTAT<6>).
7. Select Enhanced Buffer mode by setting the SPIBEN bit (SPIxCON2<0>).
8. Enable SPI operation by setting the SPIEN bit (SPIxSTAT<15>).

FIGURE 15-2: SPIx MODULE BLOCK DIAGRAM (ENHANCED MODE)



REGISTER 15-2: SPIxCON1: SPIx CONTROL REGISTER 1

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	DISSCK(1)	DISSDO(2)	MODE16	SMP	CKE(3)
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
SSEN(4)	CKP	MSTEN	SPRE2	SPRE1	SPRE0	PPRE1	PPRE0
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 **DISSCK:** Disable SCKx Pin bit (SPI Master modes only)(1)
1 = Internal SPI clock is disabled; pin functions as I/O
0 = Internal SPI clock is enabled

bit 11 **DISSDO:** Disable SDOx Pin bit(2)
1 = SDOx pin is not used by the module; pin functions as I/O
0 = SDOx pin is controlled by the module

bit 10 **MODE16:** Word/Byte Communication Select bit
1 = Communication is word-wide (16 bits)
0 = Communication is byte-wide (8 bits)

bit 9 **SMP:** SPIx Data Input Sample Phase bit
Master mode:
1 = Input data is sampled at the end of data output time
0 = Input data is sampled at the middle of data output time
Slave mode:
SMP must be cleared when SPIx is used in Slave mode.

bit 8 **CKE:** SPIx Clock Edge Select bit(3)
1 = Serial output data changes on transition from active clock state to Idle clock state (see bit 6)
0 = Serial output data changes on transition from Idle clock state to active clock state (see bit 6)

bit 7 **SSEN:** Slave Select Enable (Slave mode) bit(4)
1 = $\overline{SSx}$ pin is used for Slave mode
0 = $\overline{SSx}$ pin is not used by the module; pin is controlled by the port function

bit 6 **CKP:** Clock Polarity Select bit
1 = Idle state for the clock is a high level; active state is a low level
0 = Idle state for the clock is a low level; active state is a high level

bit 5 **MSTEN:** Master Mode Enable bit
1 = Master mode
0 = Slave mode

Note 1: If DISSCK = 0, SCKx must be configured to an available RPn pin. See **Section 10.4 "Peripheral Pin Select (PPS)"** for more information.

2: If DISSDO = 0, SDOx must be configured to an available RPn pin. See **Section 10.4 "Peripheral Pin Select (PPS)"** for more information.

3: The CKE bit is not used in the Framed SPI modes. The user should program this bit to '0' for the Framed SPI modes (FRMEN = 1).

4: If SSEN = 1, $\overline{SSx}$ must be configured to an available RPn/PRIn pin. See **Section 10.4 "Peripheral Pin Select (PPS)"** for more information.

18.0 UNIVERSAL SERIAL BUS WITH ON-THE-GO SUPPORT (USB OTG)

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 *"PIC24F Family Reference Manual"*, **Section 27. "USB On-The-Go (OTG)"** (DS39721). The information in this data sheet supersedes the information in the FRM.

PIC24FJ256GB210 family devices contain a full-speed and low-speed compatible, On-The-Go (OTG) USB Serial Interface Engine (SIE). The OTG capability allows the device to act either as a USB peripheral device or as a USB embedded host with limited host capabilities. The OTG capability allows the device to dynamically switch from device to host operation using OTG's Host Negotiation Protocol (HNP).

For more details on OTG operation, refer to the "*On-The-Go Supplement"* to the "*USB 2.0 Specification*", published by the USB-IF. For more details on USB operation, refer to the "*Universal Serial Bus Specification*", v2.0.

The USB OTG module offers these features:

- USB functionality in Device and Host modes, and OTG capabilities for application-controlled mode switching
- Software-selectable module speeds of full speed (12 Mbps) or low speed (1.5 Mbps, available in Host mode only)
- Support for all four USB transfer types: control, interrupt, bulk and isochronous
- 16 bidirectional endpoints for a total of 32 unique endpoints
- DMA interface for data RAM access
- Queues up to sixteen unique endpoint transfers without servicing
- Integrated, on-chip USB transceiver with support for off-chip transceivers via a digital interface
- Integrated VBUS generation with on-chip comparators and boost generation, and support of external VBUS comparators and regulators through a digital interface
- Configurations for on-chip bus pull-up and pull-down resistors

A simplified block diagram of the USB OTG module is shown in Figure 18-1.

The USB OTG module can function as a USB peripheral device or as a USB host, and may dynamically switch between Device and Host modes under software control. In either mode, the same data paths and Buffer Descriptors (BDs) are used for the transmission and reception of data.

In discussing USB operation, this section will use a controller-centric nomenclature for describing the direction of the data transfer between the microcontroller and the USB. RX (Receive) will be used to describe transfers that move data from the USB to the microcontroller and TX (Transmit) will be used to describe transfers that move data from the microcontroller to the USB. Table 18-1 shows the relationship between data direction in this nomenclature and the USB tokens exchanged.

TABLE 18-1: CONTROLLER-CENTRIC DATA DIRECTION FOR USB HOST OR TARGET

USB Mode	Direction	
	RX	TX
Device	OUT or SETUP	IN
Host	IN	OUT or SETUP

This chapter presents the most basic operations needed to implement USB OTG functionality in an application. A complete and detailed discussion of the USB protocol and its OTG supplement are beyond the scope of this data sheet. It is assumed that the user already has a basic understanding of USB architecture and the latest version of the protocol.

Not all steps for proper USB operation (such as device enumeration) are presented here. It is recommended that application developers use an appropriate device driver to implement all of the necessary features. Microchip provides a number of application-specific resources, such as USB firmware and driver support. Refer to www.microchip.com/usb for the latest firmware and driver support.

18.6.2 HOST NEGOTIATION PROTOCOL (HNP)

In USB OTG applications, a Dual Role Device (DRD) is a device that is capable of being either a host or a peripheral. Any OTG DRD must support Host Negotiation Protocol (HNP).

HNP allows an OTG B-device to temporarily become the USB host. The A-device must first enable the B-device to follow HNP. Refer to the *"On-The-Go Supplement"* to the *"USB 2.0 Specification"* for more information regarding HNP. HNP may only be initiated at full speed.

After being enabled for HNP by the A-device, the B-device requests being the host any time that the USB link is in suspend state, by simply indicating a disconnect. This can be done in software by clearing DPPULUP and DMPULUP. When the A-device detects the disconnect condition (via the URSTIF (U1IR<0>) interrupt), the A-device may allow the B-device to take over as host. The A-device does this by signaling connect as a full-speed function. Software may accomplish this by setting DPPULUP.

If the A-device responds instead with resume signaling, the A-device remains as host. When the B-device detects the connect condition (via ATTACHIF (U1IR<6>), the B-device becomes host. The B-device drives Reset signaling prior to using the bus.

When the B-device has finished in its role as host, it stops all bus activity and turns on its D+ pull-up resistor by setting DPPULUP. When the A-device detects a suspend condition (Idle for 3 ms), the A-device turns off its D+ pull-up. The A-device may also power-down the VBUS supply to end the session. When the A-device detects the connect condition (via ATTACHIF), the A-device resumes host operation and drives Reset signaling.

18.6.3 EXTERNAL VBUS COMPARATORS

The external VBUS comparator option is enabled by setting the UVCMPDIS bit (U1CNFG2<1>). This disables the internal VBUS comparators, removing the need to attach VBUS to the microcontroller's VBUS pin.

The external comparator interface uses either the VCMPST1 and VCMPST2 pins, or the VBUSVLD, SESSVLD and SESSEND pins, based upon the setting of the UVCMPSEL bit (U1CNFG2<5>). These pins are digital inputs and should be set in the following patterns (see Table 18-3), based on the current level of the VBUS voltage.

TABLE 18-3: EXTERNAL VBUS COMPARATOR STATES

If UVCMPSEL = 0			
VCMPST1	VCMPST2	Bus Condition	
0	0	VBUS < VB_SESS_END	
1	0	VB_SESS_END < VBUS < VA_SESS_VLD	
0	1	VA_SESS_VLD < VBUS < VA_VBUS_VLD	
1	1	VBUS > VBUS_VLD	
If UVCMPSEL = 1			
VBUSVLD	SESSVLD	SESSEND	Bus Condition
0	0	1	VBUS < VB_SESS_END
0	0	0	VB_SESS_END < VBUS < VA_SESS_VLD
0	1	0	VA_SESS_VLD < VBUS < VA_VBUS_VLD
1	1	0	VBUS > VBUS_VLD

REGISTER 20-6: WKDYHR: WEEKDAY AND HOURS VALUE REGISTER(1)

U-0	U-0	U-0	U-0	U-0	R/W-x, HSC	R/W-x, HSC	R/W-x, HSC
—	—	—	—	—	WDAY2	WDAY1	WDAY0
bit 15							bit 8

U-0	U-0	R/W-x, HSC	R/W-x, HSC	R/W-x, HSC	R/W-x, HSC	R/W-x, HSC	R/W-x, HSC
—	—	HRTEN1	HRTEN0	HRONE3	HRONE2	HRONE1	HRONE0
bit 7							bit 0

Legend:	HSC = Hardware Settable/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 15-11 **Unimplemented:** Read as '0'

bit 10-8 **WDAY<2:0>:** Binary Coded Decimal Value of Weekday Digit bits
Contains a value from 0 to 6.

bit 7-6 **Unimplemented:** Read as '0'

bit 5-4 **HRTEN<1:0>:** Binary Coded Decimal Value of Hour's Tens Digit bits
Contains a value from 0 to 2.

bit 3-0 **HRONE<3:0>:** Binary Coded Decimal Value of Hour's Ones Digit bits
Contains a value from 0 to 9.

Note 1: A write to this register is only allowed when RTCWREN = 1.

REGISTER 20-7: MINSEC: MINUTES AND SECONDS VALUE REGISTER

U-0	R/W-x, HSC	R/W-x, HSC	R/W-x, HSC	R/W-x, HSC	R/W-x, HSC	R/W-x, HSC	R/W-x, HSC
—	MINTEN2	MINTEN1	MINTEN0	MINONE3	MINONE2	MINONE1	MINONE0
bit 15							bit 8

U-0	R/W-x, HSC	R/W-x, HSC	R/W-x, HSC	R/W-x, HSC	R/W-x, HSC	R/W-x, HSC	R/W-x, HSC
—	SECTEN2	SECTEN1	SECTEN0	SECONE3	SECONE2	SECONE1	SECONE0
bit 7							bit 0

Legend:	HSC = Hardware Settable/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 **Unimplemented:** Read as '0'

bit 14-12 **MINTEN<2:0>:** Binary Coded Decimal Value of Minute's Tens Digit bits
Contains a value from 0 to 5.

bit 11-8 **MINONE<3:0>:** Binary Coded Decimal Value of Minute's Ones Digit bits
Contains a value from 0 to 9.

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

bit 6-4 **SECTEN<2:0>:** Binary Coded Decimal Value of Second's Tens Digit bits
Contains a value from 0 to 5.

bit 3-0 **SECONE<3:0>:** Binary Coded Decimal Value of Second's Ones Digit bits
Contains a value from 0 to 9.

23.0 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 associated "*PIC24F Family Reference Manual*".

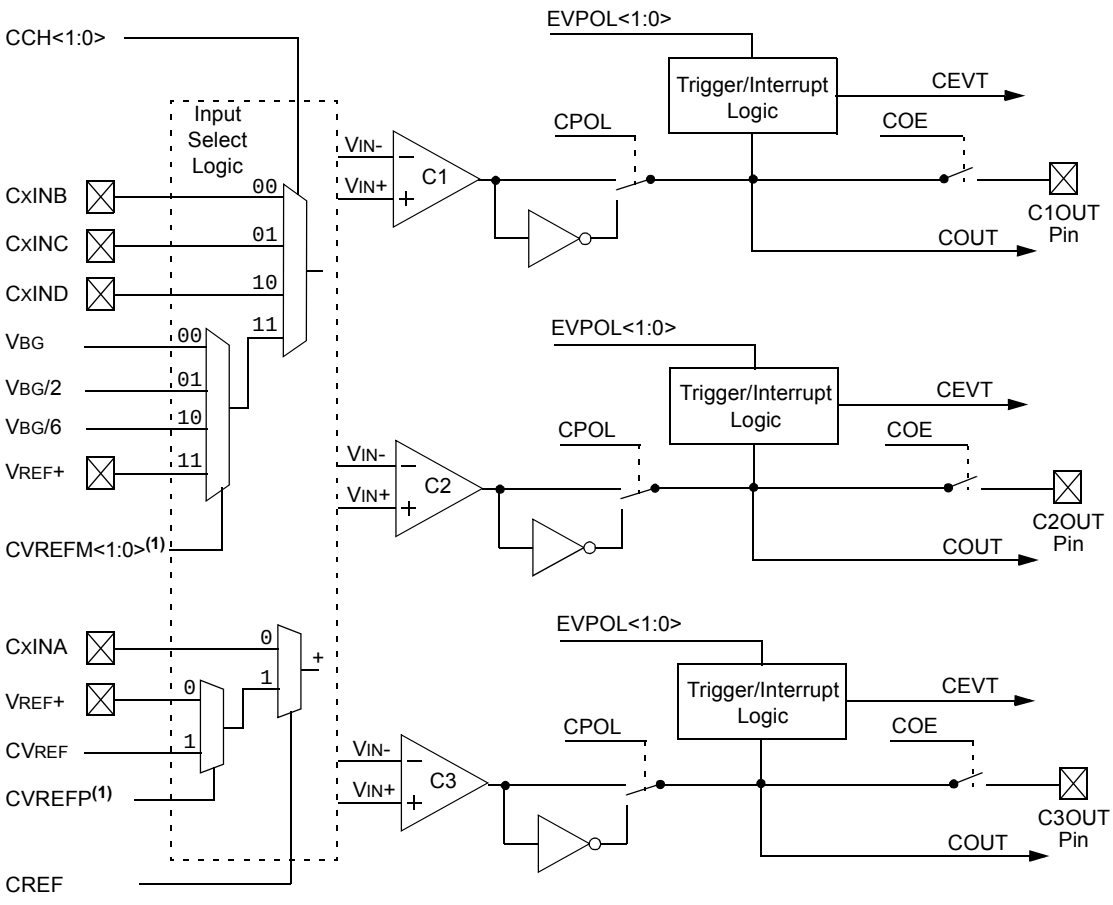
The triple comparator module provides three dual input comparators. The inputs to the comparator can be configured to use any one of five external analog inputs (CxINA, CxINB, CxINC, CxIND and VREF+) and a voltage reference input from one of the internal band gap references or the comparator voltage reference generator (VBG, VBG/2, VBG/6 and CVREF).

The comparator outputs may be directly connected to the CxOUT pins. When the respective COE 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 23-1. Diagrams of the possible individual comparator configurations are shown in Figure 23-2.

Each comparator has its own control register, CMxCON (Register 23-1), for enabling and configuring its operation. The output and event status of all three comparators is provided in the CMSTAT register (Register 23-2).

FIGURE 23-1: TRIPLE COMPARATOR MODULE BLOCK DIAGRAM



Note 1: Refer Register 24-1 for bit details.

NOTES: