PIC24FJ128GC010 FAMILY

16-Bit Flash Microcontrollers with 12-Bit Pipeline A/D, Sigma-Delta A/D, USB On-The-Go and XLP Technology

Advanced Analog Features

- 12-Bit, up to 50-Channel, High-Speed, Pipelined Analog-to-Digital (A/D) Converter:
  - Conversion rates up to 10 Msps
  - Compatibility features for low conversion rates
  - Flexible operating modes with auto-accumulate, Threshold Detect and channel scan using sample lists
  - Conversion available during Sleep and Idle
- 16-Bit Sigma-Delta Analog-to-Digital (A/D) Converter:
  - Programmable data rate with dithering option and adjustable oversampling ratios
  - Two differential channels
  - Configurable input gain stage
- Two 10-Bit Digital-to-Analog Converters (DAC):
  - Fast settling time supports 1 Msps update rate
- Two Rail-to-Rail, Input/Output, General Purpose Operational Amplifiers:
  - 2.5 MHz gain bandwidth product (typical)
  - Flexible input multiplexing options
  - Optional Comparator mode
- Three Rail-to-Rail, Enhanced Analog Comparators with Programmable Input/Output Configuration
- Three On-Chip Programmable Voltage References
- Charge Time Measurement Unit (CTMU):
  - Used for capacitive touch sensing, up to 50 channels
  - Time measurement down to 100 ps resolution
  - Operation in Sleep mode

Extreme Low-Power Features

- Multiple Power Management Options for Extreme Power Reduction:
  - VBAT allows for lowest power consumption on backup battery (with or without RTCC)
  - Deep Sleep allows near total power-down, with the ability to wake-up on internal or external triggers
  - Full RAM and state retention in select Deep Sleep and VBAT modes
  - Sleep and Idle modes selectively shut down peripherals and/or core for substantial power reduction and fast wake-up
  - Doze mode allows CPU to run at a lower clock speed than peripherals
- Alternate Clock modes Allow On-the-Fly Switching to a Lower Clock Speed for Selective Power Reduction
- Extreme Low-Power Current Consumption for Deep Sleep:
  - WDT: 270 nA @ 3.3V, typical
  - RTCC: 350 nA @ 32 kHz, 3.3V, typical
  - Deep Sleep current, 75 nA, 3.3V, typical

Device	Memory		Pins	Analog Peripherals						Digital Peripherals							LCD Controller (pixels)	USB OTG	Deep Sleep w/VBAT
	Program Flash (bytes)	Data RAM (bytes)		12-Bit HS A/D (ch)	16-Bit ΣΔ A/D (diff ch)	10-Bit DAC	Op Amps	Comparators	CTMU (ch)	Input Capture	Output Compare/PWM	I²C™	SPI	UART w/IrDA®	EPMP/PSP	16-Bit Timers			
PIC24FJ128GC010	128K	8K	100	50	2	2	2	3	50	9	9	2	2	4	Y	5	472	Y	Y
PIC24FJ128GC006	128K	8K	64	30	2	2	2	3	30	9	9	2	2	4	Y	5	248	Y	Y
PIC24FJ64GC010	64K	8K	100	50	2	2	2	3	50	9	9	2	2	4	Y	5	472	Y	Y
PIC24FJ64GC006	64K	8K	64	30	2	2	2	3	30	9	9	2	2	4	Y	5	248	Y	Y

Universal Serial Bus Features

- USB v2.0 On-The-Go (OTG) Compliant
- USB Device mode Operation from FRC Oscillator – No Crystal Oscillator Required
- Dual Role Capable – Can Act as Either Host or Peripheral
- Low-Speed (1.5 Mb/s) and Full-Speed (12 Mb/s) USB Operation in Host mode
- Full-Speed USB Operation in Device mode
- Low Jitter PLL for USB
- Supports up to 32 Endpoints (16 bidirectional):
  - USB module can use any RAM location on the device as USB endpoint buffers
- On-Chip USB Transceiver with Interface for Off-Chip USB Transceiver
- Supports Control, Interrupt, Isochronous and Bulk Transfers
- On-Chip Pull-up and Pull-Down Resistors

Peripheral Features

- LCD Display Controller:
  - Up to 59 segments by 8 commons
  - Internal charge pump and low-power, internal resistor biasing
  - Operation in Sleep mode
- Up to Five External Interrupt Sources
- Peripheral Pin Select (PPS); Allows Independent I/O Mapping of Many Peripherals
- Five 16-Bit Timers/Counters with Prescaler:
  - Can be paired as 32-bit timers/counters
- Six-Channel DMA Supports All Peripheral modules:
  - Minimizes CPU overhead, increases data throughput and lowers power consumption
- Nine Input Capture modules, Each with a Dedicated 16-Bit Timer
- Nine Output Compare/PWM modules, Each with a Dedicated 16-Bit Timer
- Enhanced Parallel Master/Slave Port (EPMP/EPSP)
- Hardware Real-Time Clock/Calendar (RTCC):
  - Run, Sleep, Deep Sleep and V_{BAT} modes
- Two 3-Wire/4-Wire SPI modules (support 4 Frame modes) with 8-Level FIFO Buffer
- Two I^2C™ modules Support Multi-Master/Slave mode and 7-Bit/10-Bit Addressing
- Four UART modules:
  - Support RS-485, RS-232 and LIN/J2602
  - On-chip hardware encoder/decoder for IrDA®
  - Auto-wake-up on Auto-Baud Detect (ABD)
  - 4-level deep FIFO buffer
- Programmable, 32-Bit Cyclic Redundancy Check (CRC) Generator
- Digital Signal Modulator (DSM) Provides On-Chip FSK and PSK Modulation for a Digital Signal Stream
- High-Current Sink/Source (18 mA/18 mA) on All I/O Pins
- Configurable Open-Drain Outputs on Digital I/O Pins
- 5.5V Tolerant Inputs on Select Pins

High-Performance CPU

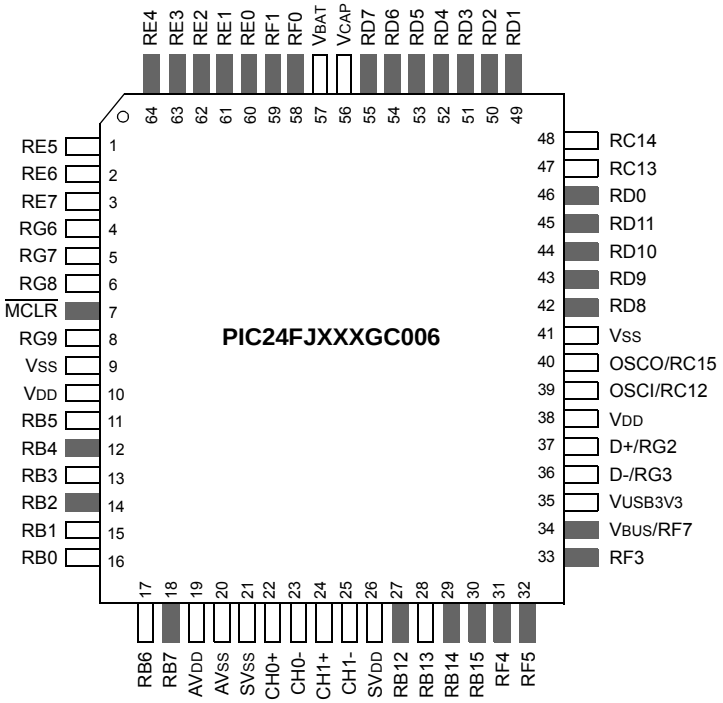
- Modified Harvard Architecture
- Up to 16 MIPS Operation @ 32 MHz
- C Compiler Optimized Instruction Set Architecture (ISA)
- 8 MHz Internal Oscillator:
  - 96 MHz PLL option for USB clocking
  - Multiple clock divide options
  - Run-time self-calibration capability for maintaining better than ±0.20% accuracy
  - Fast start-up
- 17-Bit x 17-Bit Single-Cycle Hardware Fractional/Integer Multiplier
- 32-Bit by 16-Bit Hardware Divider
- 16 x 16-Bit Working Register Array
- Two Address Generation Units (AGUs) for Separate Read and Write Addressing of Data Memory

Special Microcontroller Features

- Supply Voltage Range of 2.0V to 3.6V
- Two On-Chip Voltage Regulators (1.8V and 1.2V) for Regular and eXtreme Low-Power Operation
- 20,000 Erase/Write Cycle Endurance Flash Program Memory, Typical
- Flash Data Retention: 20 Years Minimum
- Self-Programmable under Software Control
- Programmable Reference Clock Output
- In-Circuit Serial Programming™ (ICSP™) and In-Circuit Emulation (ICE) via 2 Pins
- JTAG Boundary Scan Support
- Fail-Safe Clock Monitor Operation:
  - Detects clock failure and switches to on-chip, low-power RC oscillator
- Power-on Reset (POR), Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
- Separate Brown-out Reset (BOR) and Deep Sleep Brown-out Reset (DSBOR) Circuits
- Programmable High/Low-Voltage Detect (HLVD)
- Flexible Watchdog Timer (WDT) with its Own RC Oscillator for Reliable Operation
- Standard and Ultra Low-Power Watchdog Timers for Reliable Operation in Standard and Deep Sleep modes

Pin Diagrams

64-Pin TQFP (10 mm x 10 mm)
64-Pin QFN (9 mm x 9 mm)(1)



Legend: **Shaded pins** indicate pins tolerant to up to +5.5 VDC.
See Table 1 for a complete description of pin functions.
Note 1: It is recommended to connect the metal pad on the bottom of the 64-pin QFN package to VSS.

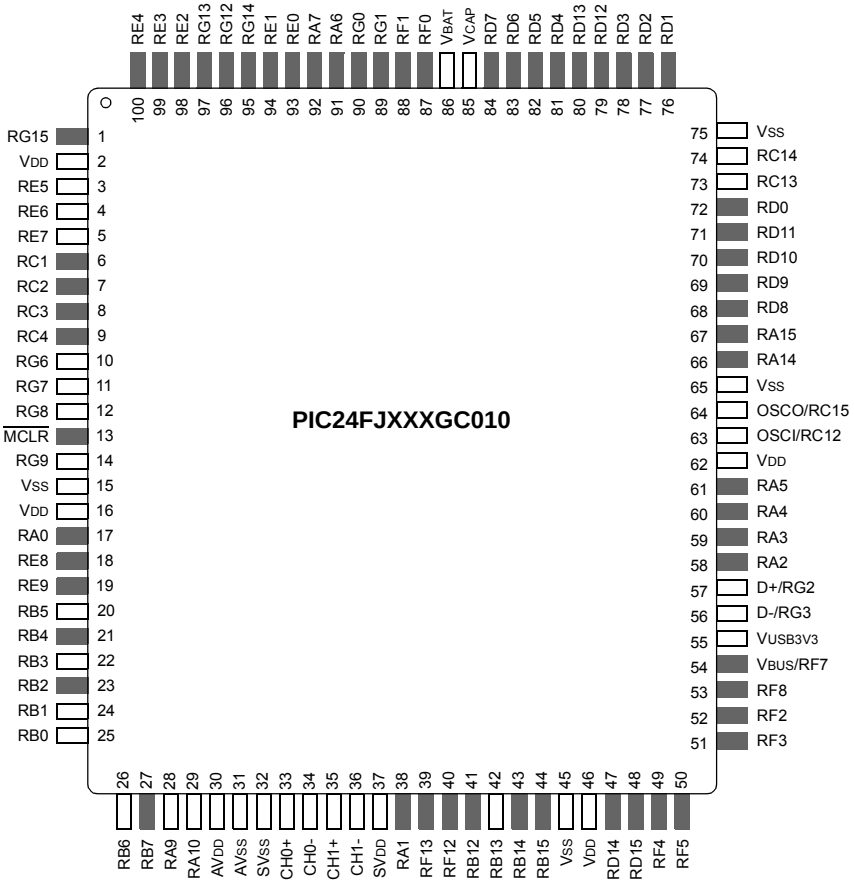
TABLE 1: COMPLETE PIN FUNCTION DESCRIPTIONS FOR 64-PIN DEVICES

Pin	Function	Pin	Function
1	CTED4/PMD5/LCDBIAS2/CN63/RE5	33	AN30/SEG12/**RP16**/USBID/PMA12/CN71/RF3
2	PMD6/LCDBIAS1/CN64/RE6	34	VBUS/CN83/RF7
3	PMD7/LCDBIAS0/CN65/RE7	35	VUSB3V3
4	BGBUF2/AN17/OA1PB/C1IND/SEG0/**RP21**/T5CK/PMA5/CN8/RG6	36	D-/CN73/RG3
5	VLCAP1/AN18/OA1NE/C1INC/**RP26**/PMA4/CN9/RG7	37	D+/CN72/RG2
6	VLCAP2/AN19/OA1ND/C2IND/**RP19**/PMA3/CN10/RG8	38	VDD
7	$\overline{\text{MCLR}}$	39	OSCI/CLKI/CN23/RC12
8	AN49/OA1PA/C2INC/SEG1/DAC1/**RP27**/PMA2/CN11/RG9	40	OSCO/CLKO/CN22/RC15
9	VSS	41	VSS
10	VDD	42	AN40/SEG13/**RP2**/RTCC/DMLN/OCTRIG1/PMA13/CN53/RD8
11	PGEC3/AN5/OA1OUT/C1INA/SEG2/**RP18**/CN7/RB5	43	AN24/SEG14/**RP4**/SDA1/DPLN/PMACK2/CN54/RD9
12	PGED3/AN4/OA2NC/C1INB/SEG3/**RP28**/USBOEN/CN6/RB4	44	AN41/C3IND/SEG15/SCL1/PMA15/CS2/CN55/RD10
13	AN3/OA2OUT/C2INA/SEG4/VPIO/CN5/RB3	45	TMS/AN42/OA2PA/C3INC/SEG16/**RP12**/PMA14/CS1/CN56/RD11
14	AN2/OA2NC/CTCMP/C2INB/SEG5/**RP13**/T4CK/VMIO/CTED13/PMA7/CN4/RB2	46	AN43/OA2NA/SEG17/**RP11**/VCMPST3/DMH/INT0/CN49/RD0
15	PGEC1/CVREF-/AVREF-/AN1/OA2PB/SEG6/**RP1**/CTED12/CN3/RB1	47	SOSCI/RC13
16	PGED1/CVREF+/AVREF+/DVREF+/BGBUF1/AN0/SEG7/**RP0**/PMA6/CN2/RB0	48	SOSCO/**RPI37**/SCLKI/RC14
17	PGEC2/AN6/OA1PD/**RP6**/LCDBIAS3/CN24/RB6	49	AN35/SEG20/**RP24**/CN50/RD1
18	PGED2/AN7/COM6/SEG30/**RP7**/CN25/RB7	50	AN25/OA2NB/SEG21/**RP23**/DPH/PMACK1/CN51/RD2
19	AVDD	51	AN44/OA2PE/SEG22/**RP22**/PMBE0/CN52/RD3
20	AVSS	52	AN47/OA1PE/SEG23/**RP25**/PMWR/CN13/RD4
21	SVSS	53	AN48/OA1NB/SEG24/**RP20**/PMRD/CN14/RD5
22	CH0+	54	AN34/OA1PC/C3INB/SEG25/CN15/RD6
23	CH0-	55	AN20/C3INA/SEG26/CN16/RD7
24	CH1+/SVREF+	56	VCAP
25	CH1-/CH1SE/SVREF-	57	VBAT
26	SVDD	58	COM7/SEG27/VCMPST1/CN68/RF0
27	TCK/AN12/COM5/SEG18/T1CK/CTED2/PMA11/CN30/RB12	59	COM4/SEG47/VCMPST2/CN69/RF1
28	TDI/AN13/OA2PD/SEG19/DAC2/CTED1/PMA10/CN31/RB13	60	COM3/PMD0/CN58/RE0
29	TDO/AN14/OA2NE/SEG8/**RP14**/CTED5/CTPLS/PMA1/CN32/RB14	61	COM2/PMD1/CN59/RE1
30	AN15/SEG9/**RP29**/T2CK/REFO/CTED6/PMA0/CN12/RB15	62	COM1/PMD2/CN60/RE2
31	AN11/OA2ND/SEG10/**RP10**/SDA2/T3CK/PMA9/CN17/RF4	63	COM0/CTED9/PMD3/CN61/RE3
32	CVREF/AN10/OA2PC/SEG11/**RP17**/SCL2/PMA8/CN18/RF5	64	HLVDIN/SEG62/CTED8/PMD4/CN62/RE4

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

Pin Diagrams (Continued)

100-Pin TQFP (12 mm x 12 mm)



Legend: **Shaded pins** indicate pins tolerant to up to +5.5 VDC.
See Table 2 for a complete description of pin functions.

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

Pin	Function	Pin	Function
1	AN33/SEG50/CTED3/CN82/RG15	41	AN12/COM5/SEG18/T1CK/CTED2/PMA11/CN30/RB12
2	VDD	42	AN13/OA2PD/SEG19/DAC2/CTED1/PMA10/CN31/RB13
3	CTED4/PMD5/LCDBIAS2/CN63/RE5	43	AN14/OA2NE/SEG8/**RP14**/CTED5/CTPLS/PMA1/CN32/RB14
4	PMD6/LCDBIAS1/CN64/RE6	44	AN15/SEG9/**RP29**/T2CK/REFO/CTED6/PMA0/CN12/RB15
5	PMD7/LCDBIAS0/CN65/RE7	45	VSS
6	AN8/OA1NB/SEG32/**RPI38**/CN45/RC1	46	VDD
7	SEG51/**RPI39**/CN46/RC2	47	AN28/SEG38/**RPI43**/CN20/RD14
8	AN9/SEG33/**RPI40**/CN47/RC3	48	AN29/SEG39/**RP5**/CN21/RD15
9	AN16/SEG52/**RPI41**/PMCS2/CN48/RC4	49	AN11/OA2ND/SEG10/**RP10**/SDA2(3)/T3CK/PMA9/CN17/RF4
10	BGBUF2/AN17/OA1PB/C1IND/SEG0/**RP21**/T5CK/PMA5/CN8/RG6	50	CVREF/AN10/OA2PC/SEG11/**RP17**/SCL2(3)/PMA8/CN18/RF5
11	VLCAP1/AN18/OA1NE/C1INC/**RP26**/PMA4/CN9/RG7	51	AN30/SEG12/**RP16**/USBID/PMA12/CN71/RF3
12	VLCAP2/AN19/OA1NC/C2IND/**RP19**/PMA3/CN10/RG8	52	AN31/SEG40/**RP30**/CN70/RF2
13	$\overline{\text{MCLR}}$	53	AN32/SEG41/**RP15**/CN74/RF8
14	AN49/OA1PA/C2INC/SEG1/DAC1/**RP27**/PMA2/CN11/RG9	54	VBUS/CN83/RF7
15	VSS	55	VUSB3V3
16	VDD	56	D-/CN73/RG3
17	TMS/SEG48/CTED0/CN33/RA0	57	D+/CN72/RG2
18	SEG34/**RPI33**/PMCS1/CN66/RE8	58	SEG55/SCL2/CN35/RA2
19	AN21/SEG35/**RPI34**/PMA19/CN67/RE9	59	SEG56/SDA2/PMA20/CN36/RA3
20	PGEC3/AN5/OA1OUT/C1INA/SEG2/**RP18**/CN7/RB5	60	TDI/AN36/SEG29/PMA21/CN37/RA4
21	PGED3/AN4/OA1NA/C1INB/SEG3/**RP28**/USBOEN/CN6/RB4	61	TDO/AN37/SEG28/CN38/RA5
22	AN3/OA2OUT/C2INA/SEG4/VPIO/CN5/RB3	62	VDD
23	AN2/OA2NC/CTCMP/C2INB/SEG5/**RP13**/T4CK/VMIO/CTED13/CN4/RB2	63	OSCI/CLKI/CN23/RC12
24	PGEC1/CVREF-/AVREF-/AN1/OA2PB/SEG6/**RP1**/CTED12/CN3/RB1	64	OSCO/CLKO/CN22/RC15
25	PGED1/CVREF+/AVREF+/DVREF+/BGBUF1/AN0/SEG7/**RP0**/CN2/RB0	65	VSS
26	PGEC2/AN6/OA1PD/**RP6**/LCDBIAS3/CN24/RB6	66	AN38/SEG42/**RPI36**/SCL1/OCTRIG2/PMA22/CN43/RA14
27	PGED2/AN7/COM6/SEG30/**RP7**/CN25/RB7	67	AN39/SEG43/**RPI35**/SDA1/PMBE1/CN44/RA15
28	CVREF-(1)/AVREF-(2)/SEG36/PMA7/CN41/RA9	68	AN40/SEG13/**RP2**/RTCC/DMLN/OCTRIG1/PMA13/CN53/RD8
29	CVREF+(1)/AVREF+(2)/SEG37/PMA6/CN42/RA10	69	AN24/SEG14/**RP4**/DPLN/PMACK2/CN54/RD9
30	AVDD	70	AN41/C3IND/SEG15/PMA15/CS2/CN55/RD10
31	AVSS	71	AN42/OA2PA/C3INC/SEG16/**RP12**/PMA14/CS1/CN56/RD11
32	SVSS	72	AN43/OA2NA/SEG17/**RP11**/VCMPST3/DMH/INT0/CN49/RD0
33	CH0+	73	SOSCI/RC13
34	CH0-	74	SOSCO/SCLKI/**RPI37**/RC14
35	CH1+/SVREF+	75	VSS
36	CH1-/CH1SE/SVREF-	76	AN35/SEG20/**RP24**/CN50/RD1
37	SVDD	77	AN25/OA2NB/SEG21/**RP23**/DPH/PMACK1/CN51/RD2
38	TCK/AN26/SEG31/CN34/RA1	78	AN44/OA2PE/SEG22/**RP22**/PMBE0/CN52/RD3
39	AN27/SEG53/**RP31**/CN76/RF13	79	AN45/SEG44/**RPI42**/PMD12/CN57/RD12
40	SEG54/**RPI32**/CTED7/PMA18/CN75/RF12	80	AN46/SEG45/PMD13/CN19/RD13

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

Note 1: Alternate pin assignments for the external comparator voltage references as determined by the ALTCVREF Configuration bit.

2: Alternate pin assignments for the external A/D voltage references as determined by the ALTADREF Configuration bit.

3: Alternate pin assignments for I2C2 as determined by the $\overline{\text{I2C2SEL}}$ Configuration bit.

TABLE 2: COMPLETE PIN FUNCTION DESCRIPTIONS FOR 100-PIN DEVICES (CONTINUED)

Pin	Function	Pin	Function
81	AN47/OA1PE/SEG23/**RP25**/PMWR/CN13/RD4	91	AN23/SEG57/CN39/RA6
82	AN48/OA1NB/SEG24/**RP20**/PMRD/CN14/RD5	92	AN22/SEG58/PMA17/CN40/RA7
83	AN34/OA1PC/C3INB/SEG25/PMD14/CN15/RD6	93	COM3/PMD0/CN58/RE0
84	AN20/C3INA/SEG26/PMD15/CN16/RD7	94	COM2/PMD1/CN59/RE1
85	VCAP	95	SEG59/CTED11/PMA16/CN81/RG14
86	VBAT	96	SEG60/CN79/RG12
87	COM7/SEG27/VCMPST1/PMD11/CN68/RF0	97	SEG61/CTED10/CN80/RG13
88	COM4/SEG47/VCMPST2/PMD10/CN69/RF1	98	COM1/PMD2/CN60/RE2
89	SEG46/PMD9/CN78/RG1	99	COM0/CTED9/PMD3/CN61/RE3
90	SEG49/PMD8/CN77/RG0	100	HLVDIN/SEG62/CTED8/PMD4/CN62/RE4

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

Note 1: Alternate pin assignments for the external comparator voltage references as determined by the ALTCVREF Configuration bit.

2: Alternate pin assignments for the external A/D voltage references as determined by the ALTADREF Configuration bit.

3: Alternate pin assignments for I2C2 as determined by the $\overline{\text{I2C2SEL}}$ Configuration bit.

Pin Diagrams (Continued)

121-Pin BGA (10 mm x 10 mm, Top View)

	1	2	3	4	5	6	7	8	9	10	11
A	RE4	RE3	RG13	RE0	RG0	RF1	VBAT	N/C	RD12	RD2	RD1
B	N/C	RG15	RE2	RE1	RA7	RF0	VCAP	RD5	RD3	VSS	RC14
C	RE6	VDD	RG12	RG14	RA6	N/C	RD7	RD4	N/C	RC13	RD11
D	RC1	RE7	RE5	N/C	N/C	N/C	RD6	RD13	RD0	N/C	RD10
E	RC4	RC3	RG6	RC2	N/C	RG1	N/C	RA15	RD8	RD9	RA14
F	$\overline{\text{MCLR}}$	RG8	RG9	RG7	VSS	N/C	N/C	VDD	OSCI/ RC12	VSS	OSCO/ RC15
G	RE8	RE9	RA0	N/C	VDD	VDD	VSS	N/C	RA5	RA3	RA4
H	RB5	RB4	N/C	N/C	CH0-	N/C	N/C	VBUS/ RF7	VUSB3V3	D+/RG2	RA2
J	RB3	RB2	RB7	AVDD	SVDD	RA1	RB12	N/C	N/C	RF8	D-/RG3
K	RB1	RB0	RA10	SVSS	CH1+	RF12	RB14	VDD	RD15	RF3	RF2
L	RB6	RA9	AVSS	CH0+	CH1-	RF13	RB13	RB15	RD14	RF4	RF5

Legend: **Shaded balls** indicate pins tolerant to up to +5.5 VDC.
See Table 3 for complete pinout descriptions.

TABLE 3: COMPLETE PIN FUNCTION DESCRIPTIONS FOR 121-PIN DEVICES

Pin	Function	Pin	Function
A1	HLVDIN/SEG62/CTED8/PMD4/CN62/RE4	E1	AN16/SEG52/**RPI41**/PMCS2/CN48/RC4
A2	COM0/CTED9/PMD3/CN61/RE3	E2	AN9/OA1NC/SEG33/**RPI40**/CN47/RC3
A3	SEG61/CTED10/CN80/RG13	E3	BGBUF2/AN17/OA1PB/C1IND/SEG0/**RP21**/T5CK/PMA5/CN8/RG6
A4	COM3/PMD0/CN58/RE0	E4	SEG51/**RPI39**/CN46/RC2
A5	SEG49/PMD8/CN77/RG0	E5	N/C
A6	COM4/SEG47/V_{CMPST2}/PMD10/CN69/RF1	E6	SEG46/PMD9/CN78/RG1
A7	V_{BAT}	E7	N/C
A8	N/C	E8	AN39/SEG43/**RPI35**/SDA1/PMBE1/CN44/RA15
A9	AN45/SEG44/**RPI42**/PMD12/CN57/RD12	E9	AN40/SEG13/**RP2**/RTCC/DMLN/OCTRIG1/PMA13/CN53/RD8
A10	AN25/OA2NB/SEG21/**RP23**/DPH/PMACK1/CN51/RD2	E10	AN24/SEG14/**RP4**/DPLN/PMACK2/CN54/RD9
A11	AN35/SEG20/**RP24**/CN50/RD1	E11	AN38/SEG42/**RPI36**/SCL1/OCTRIG2/PMA22/CN43/RA14
B1	N/C	F1	$\overline{MCLR}$
B2	AN33/SEG50/CTED3/CN82/RG15	F2	V_{LCAP2}/AN19/OA1ND/C2IND/**RP19**/PMA3/CN10/RG8
B3	COM1/PMD2/CN60/RE2	F3	AN49/C2INC/SEG1/DAC1/**RP27**/PMA2/CN11/RG9
B4	COM2/PMD1/CN59/RE1	F4	V_{LCAP1}/AN18/OA1NE/C1INC/**RP26**/PMA4/CN9/RG7
B5	AN22/SEG58/PMA17/CN40/RA7	F5	V_{SS}
B6	COM7/SEG27/V_{CMPST1}/PMD11/CN68/RF0	F6	N/C
B7	V_{CAP}	F7	N/C
B8	AN48/OA1NB/SEG24/**RP20**/PMRD/CN14/RD5	F8	V_{DD}
B9	AN44/OA2PE/SEG22/**RP22**/PMBE0/CN52/RD3	F9	OSCI/CLKI/CN23/RC12
B10	V_{SS}	F10	V_{SS}
B11	SOSCO/SCLKI/**RPI37**/RC14	F11	OSCO/CLKO/CN22/RC15
C1	PMD6/LCDBIAS1/CN64/RE6	G1	SEG34/**RPI33**/PMCS1/CN66/RE8
C2	V_{DD}	G2	AN21/SEG35/**RPI34**/PMA19/CN67/RE9
C3	SEG60/CN79/RG12	G3	TMS/SEG48/CTED0/CN33/RA0
C4	SEG59/CTED11/PMA16/CN81/RG14	G4	N/C
C5	AN23/SEG57/CN39/RA6	G5	V_{DD}
C6	N/C	G6	V_{DD}
C7	AN20/C3INA/SEG26/PMD15/CN16/RD7	G7	V_{SS}
C8	AN47/OA1PE/SEG23/**RP25**/PMWR/CN13/RD4	G8	N/C
C9	N/C	G9	TDO/AN37/SEG28/CN38/RA5
C10	SOSCI/RC13	G10	SEG56/SDA2/PMA20/CN36/RA3
C11	AN42/OA2PA/C3INC/SEG16/**RP12**/PMA14/CS1/CN56/RD11	G11	TDI/AN36/SEG29/PMA21/CN37/RA4
D1	AN8/OA1NB/SEG32/**RPI38**/CN45/RC1	H1	PGEC3/AN5/OA1OUT/C1INA/SEG2/**RP18**/CN7/RB5
D2	PMD7/LCDBIAS0/CN65/RE7	H2	PGED3/AN4/OA1NA/C1INB/SEG3/**RP28**/USBOEN/CN6/RB4
D3	CTED4/PMD5/LCDBIAS2/CN63/RE5	H3	N/C
D4	N/C	H4	N/C
D5	N/C	H5	CH0-
D6	N/C	H6	N/C
D7	AN34/OA1PC/C3INB/SEG25/PMD14/CN15/RD6	H7	N/C
D8	AN46/SEG45/PMD13/CN19/RD13	H8	V_{BUS}/CN83/RF7
D9	AN43/OA2NA/SEG17/**RP11**/V_{CMPST3}/DMH/INT0/CN49/RD0	H9	V_{USB3V3}
D10	N/C	H10	D+/CN72/RG2
D11	AN41/C3IND/SEG15/PMA15/CS2/CN55/RD10	H11	SEG55/SCL2/CN35/RA2

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

Note 1: Alternate pin assignments for the external comparator voltage references as determined by the ALTCVREF Configuration bit.

2: Alternate pin assignments for the external A/D voltage references as determined by the ALTADREF Configuration bit.

3: Alternate pin assignments for I2C2 as determined by the $\overline{I2C2SEL}$ Configuration bit.

TABLE 3: COMPLETE PIN FUNCTION DESCRIPTIONS FOR 121-PIN DEVICES (CONTINUED)

Pin	Function	Pin	Function
J1	AN3/OA2OUT/C2INA/SEG4/VPIO/CN5/RB3	K7	AN14/OA2NE/SEG8/**RP14**/CTED5/CTPLS/PMA1/CN32/RB14
J2	AN2/OA2NC/CTCMP/C2INB/SEG5/**RP13**/T4CK/VMIO/ CTED13/CN4/RB2	K8	VDD
J3	PGED2/AN7/COM6/SEG30/**RP7**/CN25/RB7	K9	AN29/SEG39/**RP5**/CN21/RD15
J4	AVDD	K10	AN30/SEG12/**RP16**/USBID/PMA12/CN71/RF3
J5	SVDD	K11	AN31/SEG40/**RP30**/CN70/RF2
J6	TCK/AN26/SEG31/CN34/RA1	L1	PGEC2/AN6/OA1PD/**RP6**/LCDBIAS3/CN24/RB6
J7	AN12/COM5/SEG18/T1CK/CTED2/PMA11/CN30/RB12	L2	CVREF-(1)/AVREF-(2)/SEG36/PMA7/CN41/RA9
J8	N/C	L3	AVSS
J9	N/C	L4	CH0+
J10	AN32/SEG41/**RP15**/CN74/RF8	L5	CH1-/CH1SE/SVREF-
J11	D-/CN73/RG3	L6	AN27/SEG53/**RP31**/CN76/RF13
K1	PGEC1/CVREF-/AVREF-/AN1/OA2PB/SEG6/**RP1**/CTED12/ CN3/RB1	L7	AN13/OA2PD/SEG19/DAC2/CTED1/PMA10/CN31/RB13
K2	PGED1/CVREF+/AVREF+/DVREF+/BGBUF1/AN0/SEG7/**RP0**/ CN2/RB0	L8	AN15/SEG9/**RP29**/T2CK/REFO/CTED6/PMA0/CN12/RB15
K3	CVREF+(1)/AVREF+(2)/SEG37/PMA6/CN42/RA10	L9	AN28/SEG38/**RPI43**/CN20/RD14
K4	SVSS	L10	AN11/OA2ND/SEG10/**RP10**/SDA2(3)/T3CK/PMA9/CN17/RF4
K5	CH1+/SVREF+	L11	CVREF/AN10/OA2PC/SEG11/**RP17**/SCL2(3)/PMA8/CN18/RF5
K6	SEG54/**RPI32**/CTED7/PMA18/CN75/RF12		

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

Note 1: Alternate pin assignments for the external comparator voltage references as determined by the ALTCVREF Configuration bit.

2: Alternate pin assignments for the external A/D voltage references as determined by the ALTADREF Configuration bit.

3: Alternate pin assignments for I2C2 as determined by the $\overline{\text{I2C2SEL}}$ Configuration bit.

Table of Contents

1.0 Device Overview 13
2.0 Guidelines for Getting Started with 16-Bit Microcontrollers 33
3.0 CPU 39
4.0 Memory Organization 45
5.0 Direct Memory Access Controller (DMA) 83
6.0 Flash Program Memory 91
7.0 Resets 97
8.0 Interrupt Controller 103
9.0 Oscillator Configuration 159
10.0 Power-Saving Features 171
11.0 I/O Ports 183
12.0 Timer1 215
13.0 Timer2/3 and Timer4/5 217
14.0 Input Capture with Dedicated Timers 223
15.0 Output Compare with Dedicated Timers 227
16.0 Serial Peripheral Interface (SPI) 237
17.0 Inter-Integrated Circuit™ (I^2C™) 249
18.0 Universal Asynchronous Receiver Transmitter (UART) 257
19.0 Universal Serial Bus with On-The-Go Support (USB OTG) 265
20.0 Data Signal Modulator 299
21.0 Enhanced Parallel Master Port (EPMP) 303
22.0 Liquid Crystal Display (LCD) Controller 315
23.0 Real-Time Clock and Calendar (RTCC) 323
24.0 32-Bit Programmable Cyclic Redundancy Check (CRC) Generator 335
25.0 Overview of Advanced Analog Features 341
26.0 12-Bit High-Speed, Pipeline A/D Converter 347
27.0 16-Bit Sigma-Delta Analog-to-Digital (A/D) Converter 369
28.0 10-Bit Digital-to-Analog Converter (DAC) 375
29.0 Dual Operational Amplifier Module 377
30.0 Triple Comparator Module 381
31.0 Comparator Voltage Reference 387
32.0 Charge Time Measurement Unit (CTMU) 389
33.0 High/Low-Voltage Detect (HLVD) 397
34.0 Special Features 399
35.0 Development Support 413
36.0 Instruction Set Summary 417
37.0 Electrical Characteristics 425
38.0 Packaging Information 447
Appendix A: Revision History 459
Index 461
The Microchip Web Site 467
Customer Change Notification Service 467
Customer Support 467
Reader Response 468
Product Identification System 469

TO OUR VALUED CUSTOMERS

It is our intention to provide our valued customers with the best documentation possible to ensure successful use of your Microchip products. To this end, we will continue to improve our publications to better suit your needs. Our publications will be refined and enhanced as new volumes and updates are introduced.

If you have any questions or comments regarding this publication, please contact the Marketing Communications Department via E-mail at **docerrors@microchip.com** or fax the **Reader Response Form** in the back of this data sheet to (480) 792-4150. We welcome your feedback.

Most Current Data Sheet

To obtain the most up-to-date version of this data sheet, please register at our Worldwide Web site at:

http://www.microchip.com

You can determine the version of a data sheet by examining its literature number found on the bottom outside corner of any page. The last character of the literature number is the version number, (e.g., DS30000A is version A of document DS30000).

Errata

An errata sheet, describing minor operational differences from the data sheet and recommended workarounds, may exist for current devices. As device/documentation issues become known to us, we will publish an errata sheet. The errata will specify the revision of silicon and revision of document to which it applies.

To determine if an errata sheet exists for a particular device, please check with one of the following:

- Microchip's Worldwide Web site; **http://www.microchip.com**
- Your local Microchip sales office (see last page)

When contacting a sales office, please specify which device, revision of silicon and data sheet (include literature number) you are using.

Customer Notification System

Register on our web site at **www.microchip.com** to receive the most current information on all of our products.

1.0 DEVICE OVERVIEW

This document contains device-specific information for the following devices:

- PIC24FJ64GC006
- PIC24FJ128GC006
- PIC24FJ64GC010
- PIC24FJ128GC010

The PIC24FJ128GC010 family expands the capabilities of the PIC24F family by adding a complete selection of advanced analog peripherals to its existing digital features. This combination, along with its ultra low-power features, Direct Memory Access (DMA) for peripherals, USB On-The-Go (OTG) and a built-in LCD controller and driver, makes this family the new standard for mixed-signal PIC® microcontrollers in one economical and power-saving package.

1.1 Core Features

1.1.1 16-BIT ARCHITECTURE

Central to all PIC24F devices is the 16-bit modified Harvard architecture, first introduced with Microchip's dsPIC® Digital Signal Controllers (DSCs). The PIC24F CPU core offers a wide range of enhancements, such as:

- 16-bit data and 24-bit address paths with the ability to move information between data and memory spaces
- Linear addressing of up to 12 Mbytes (program space) and 32 Kbytes (data)
- A 16-element working register array with built-in software stack support
- A 17 x 17 hardware multiplier with support for integer math
- Hardware support for 32 by 16-bit division
- An instruction set that supports multiple addressing modes and is optimized for high-level languages, such as 'C'
- Operational performance up to 16 MIPS

1.1.2 XLP POWER-SAVING TECHNOLOGY

The PIC24FJ128GC010 family of devices introduces a greatly expanded range of power-saving operating modes for the ultimate in power conservation. The new modes include:

- Retention Sleep, with essential circuits being powered from a separate low-voltage regulator
- Retention Deep Sleep, a lower power mode that maintains data RAM for fast start-up
- Deep Sleep without RTCC, for the lowest possible power consumption under software control
- V_{BAT} mode (with or without RTCC), to continue limited operation from a backup battery when V_{DD} is removed

Many of these new low-power modes also support the continuous operation of the low-power, on-chip Real-Time Clock/Calendar (RTCC), making it possible for an application to keep time while the device is otherwise asleep.

Aside from these new features, PIC24FJ128GC010 family devices also include all of the legacy power-saving features of previous PIC24F microcontrollers, such as:

- On-the-Fly Clock Switching, allowing the selection of a lower power clock during run time
- Doze Mode Operation, for maintaining peripheral clock speed while slowing the CPU clock
- Instruction-Based Power-Saving Modes, for quick invocation of Idle and the many Sleep modes.

1.1.3 OSCILLATOR OPTIONS AND FEATURES

All of the devices in the PIC24FJ128GC010 family offer five different oscillator options, allowing users a range of choices in developing application hardware. These include:

- Two Crystal modes
- Two External Clock modes
- A Phase Lock Loop (PLL) frequency multiplier, which allows clock speeds of up to 32 MHz
- A Fast Internal Oscillator (FRC) – nominal 8 MHz output, with multiple frequency divider options and automatic frequency self-calibration during run time
- A separate Low-Power Internal RC Oscillator (LPRC) – 31 kHz nominal, for low-power, timing-insensitive applications.

The internal oscillator block also provides a stable reference source for the Fail-Safe Clock Monitor (FSCM). This option constantly monitors the main clock source against a reference signal provided by the internal oscillator and enables the controller to switch to the internal oscillator, allowing for continued low-speed operation or a safe application shutdown.

1.1.4 EASY MIGRATION

Regardless of the memory size, all devices share the same rich set of peripherals, allowing for a smooth migration path as applications grow and evolve. This extends the ability of applications to grow from the relatively simple, to the powerful and complex, while still selecting a Microchip device.

1.2 Advanced Analog Features

The centerpiece of the PIC24FJ128GC010 family is the advanced analog block. This feature set provides application developers with all the tools they need for single chip applications that demand high analog performance. Included in the advanced analog block are:

- A new 12-bit, pipelined A/D Converter (A/D) module. A major departure from previous PIC24F A/D Converters, this module offers up to 50 single-ended input channels (or up to 25 differential channel pairs) and conversion rates of up to ten million samples per second. It also provides a wider range of new features that allow the converter to assess and make decisions on incoming data without CPU intervention.
- A dual differential channel, Sigma-Delta A/D Converter, for applications requiring high-precision conversions (up to 16-bit resolution). The Sigma-Delta Converter also offers programmable gain on each channel pair, and user-configurable data rate, between 976 samples per second and 62.5 ksps.
- Two independent, 10-bit Digital-to-Analog Converters (DACs), each capable of conversion rates up to one million samples per second.
- A comparator module with three analog comparators that are configurable for a wide range of operations. The comparators also have their own independent, configurable voltage reference.
- A dual operational amplifier module with multiple input options, selectable power modes, and rail-to-rail operation on the inputs and outputs. Each of the op amps can also be configured to function as a comparator, complete with interrupt generation.
- A dedicated, integrated band gap voltage reference for all analog modules, providing a range of on-chip reference voltages and two buffered reference outputs.
- Flexible multiplexing options for the entire analog block, allowing for the convenient sharing of signals between the analog modules.

1.3 DMA Controller

PIC24FJ128GC010 family devices also add a Direct Memory Access (DMA) controller to the existing PIC24F architecture. The DMA acts in concert with the CPU, allowing data to move between data memory and peripherals without the intervention of the CPU, increasing data throughput, and decreasing execution time overhead. Six independently programmable channels make it possible to service multiple peripherals at virtually the same time, with each channel peripheral performing a different operation. Many types of data transfer operations are supported.

1.4 USB On-The-Go (OTG)

USB On-The-Go provides on-chip functionality as a target device compatible with the USB 2.0 standard, as well as limited stand-alone functionality as a USB embedded host. By implementing USB Host Negotiation Protocol (HNP), the module can also dynamically switch between device and host operation, allowing for a much wider range of versatile USB-enabled applications on a microcontroller platform.

PIC24FJ128GC010 family devices also incorporate an integrated USB transceiver and precision oscillator, minimizing the required complexity of implementing a complete USB device, embedded host, dual role or On-The-Go application.

1.5 LCD Controller

With the PIC24FJ128GC010 family of devices, Microchip introduces its versatile Liquid Crystal Display (LCD) controller and driver to the PIC24F family. The on-chip LCD driver includes many features that make the integration of displays in low-power applications easier. These include an integrated voltage regulator with charge pump and an integrated internal resistor ladder that allows contrast control in software, and display operation above device V_{DD}.

1.6 Other Special Features

- **Peripheral Pin Select (PPS):** The Peripheral Pin Select feature allows most digital peripherals to be mapped over a fixed set of digital I/O pins. Users may independently map the input and/or output of any one of the many digital peripherals to any one of the I/O pins.
- **Communications:** The PIC24FJ128GC010 family incorporates several different serial communication peripherals to handle a range of application requirements. There are two independent I^2C™ modules that support both Master and Slave modes of operation. Devices also have, through the PPS feature, four independent UARTs with built-in IrDA® encoders/decoders and two SPI modules.
- **CTMU Interface:** In addition to their other analog features, members of the PIC24FJ128GC010 family include the CTMU interface module. This provides a convenient method for precision time measurement and pulse generation, and can serve as an interface for capacitive sensors.
- **Enhanced Parallel Master/Parallel Slave Port:** This module allows rapid and transparent access to the microcontroller data bus, and enables the CPU to directly address external data memory. The parallel port can function in Master or Slave mode, accommodating data widths of 4, 8 or 16 bits, and address widths of up to 23 bits in Master modes.
- **Real-Time Clock and Calendar (RTCC):** This module implements a full-featured clock and calendar with alarm functions in hardware, freeing up timer resources and program memory space for use of the core application.
- **Data Signal Modulator (DSM):** The Data Signal Modulator (DSM) allows the user to mix a digital data stream (the “modulator signal”) with a carrier signal to produce a modulated output.

1.7 Details on Individual Family Members

Devices in the PIC24FJ128GC010 family are available in 64-pin and 100/121-pin packages. The general block diagram for all devices is shown in Figure 1-1.

The devices are differentiated from each other in six ways:

1. Flash program memory (64 Kbytes for PIC24FJ64GC0XX devices and 128 Kbytes for PIC24FJ128GC0XX devices).
2. Available I/O pins and ports (53 pins on 6 ports for 64-pin devices and 85 pins on 7 ports for 100/121-pin devices).
3. Available Interrupt-on-Change Notification (ICN) inputs (52 on 64-pin devices and 82 on 100/121-pin devices).
4. Available remappable pins (29 pins on 64-pin devices and 44 pins on 100/121-pin devices).
5. Maximum available drivable LCD pixels (196 for 64-pin devices and 472 on 100/121-pin devices.)
6. Analog input channels for the pipeline A/D Converter (29 channels for 64-pin devices and 50 channels for 100/121-pin devices).

All other features for devices in this family are identical. These are summarized in Table 1-1 and Table 1-2.

A list of pin features available on the PIC24FJ128GC010 family devices, sorted by function, is shown in Table 1-3. Note that this table shows the pin location of individual peripheral features and not how they are multiplexed on the same pin. This information is provided in the pinout diagrams in the beginning of the data sheet. Multiplexed features are sorted by the priority given to a feature, with the highest priority peripheral being listed first.

TABLE 1-1: DEVICE FEATURES FOR THE PIC24FJ128GC010 FAMILY: 64-PIN

Features	PIC24FJ64GC006	PIC24FJ128GC006
Operating Frequency	DC – 32 MHz	
Program Memory (bytes)	64K	128K
Program Memory (instructions)	22,016	44,032
Data Memory (bytes)	8K	
Interrupt Sources (soft vectors/ NMI traps)	65 (61/4)	
I/O Ports	Ports B, C, D, E, F, G	
Total I/O Pins	53	
Remappable Pins	30 (29 I/O, 1 input only)	
Timers: Total Number (16-bit)	5(1)	
32-Bit (from paired 16-bit timers)	2	
Input Capture w/Timer Channels	9(1)	
Output Compare/PWM Channels	9(1)	
Input Change Notification Interrupt	52	
Serial Communications: UART	4(1)	
SPI (3-wire/4-wire)	2(1)	
I^2C™	2	
Digital Signal Modulator	Yes	
Parallel Communications (EPMP/PSP)	Yes	
JTAG Boundary Scan	Yes	
12-Bit Pipeline Analog-to-Digital Converter (A/D) (input channels)	29	
Sigma-Delta Analog-to-Digital Converter (A/D) (differential channels)	2	
Digital-to-Analog Converter (DAC)	2	
Operational Amplifiers	2	
Analog Comparators	3	
CTMU Interface	Yes	
LCD Controller (available pixels)	196 (28 SEG x 7 COM)	
Resets (and delays)	Core POR, VDD POR, VBAT POR, BOR, RESET Instruction, MCLR, WDT, Illegal Opcode, REPEAT Instruction, Hardware Traps, Configuration Word Mismatch (OST, PLL Lock)	
Instruction Set	76 Base Instructions, Multiple Addressing Mode Variations	
Packages	64-Pin TQFP and QFN	

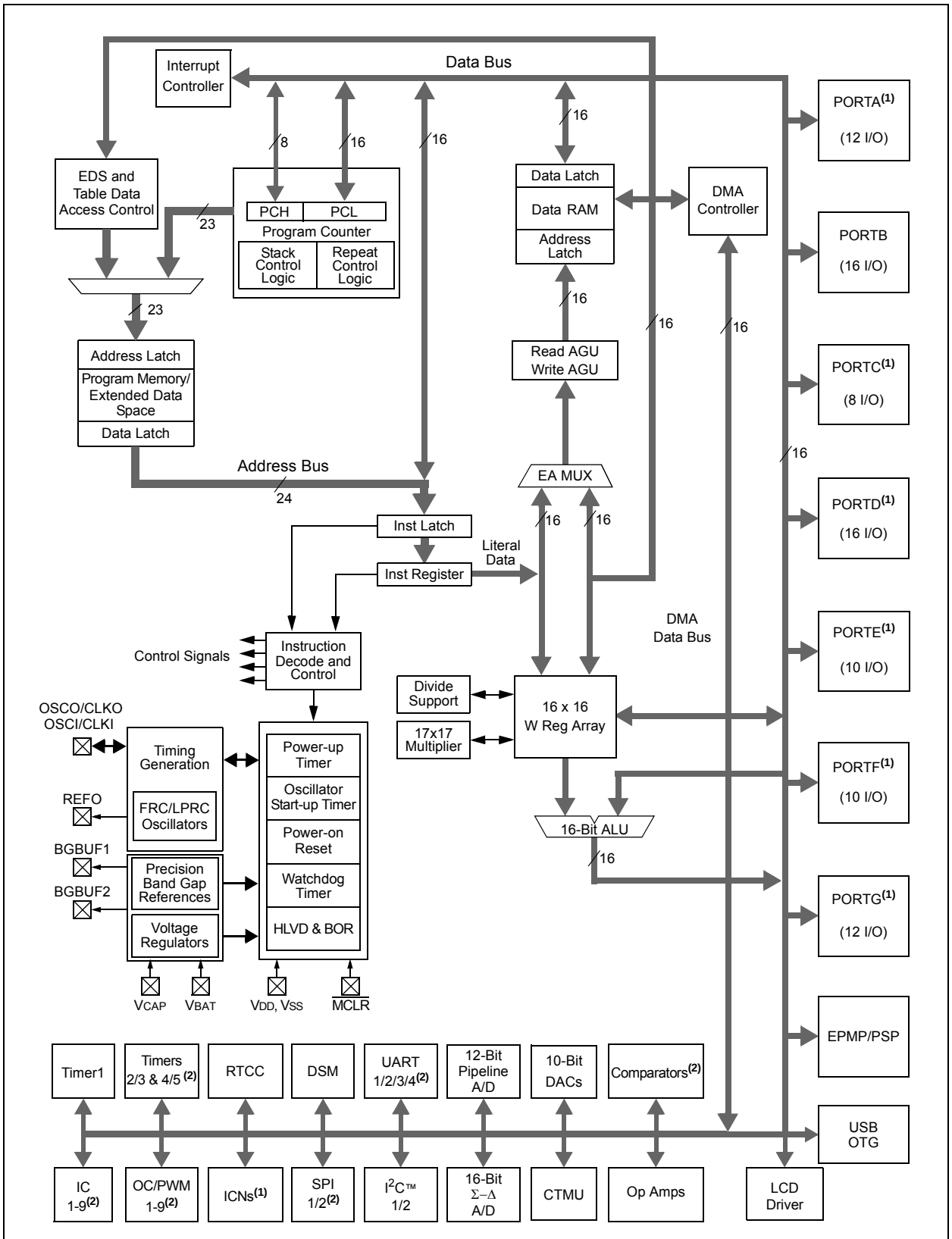
Note 1: Peripherals are accessible through remappable pins.

TABLE 1-2: DEVICE FEATURES FOR THE PIC24FJ128GC010 FAMILY: 100/121-PIN DEVICES

Features	PIC24FJ64GC010	PIC24FJ128GC010
Operating Frequency	DC – 32 MHz	
Program Memory (bytes)	64K	128K
Program Memory (instructions)	22,016	44,032
Data Memory (bytes)	8K	
Interrupt Sources (soft vectors/ NMI traps)	66 (62/4)	
I/O Ports	Ports A, B, C, D, E, F, G	
Total I/O Pins	85	
Remappable Pins	44 (32 I/O, 12 input only)	
Timers: Total Number (16-bit)	5(1)	
32-Bit (from paired 16-bit timers)	2	
Input Capture w/Timer Channels	9(1)	
Output Compare/PWM Channels	9(1)	
Input Change Notification Interrupt	82	
Serial Communications: UART	4(1)	
SPI (3-wire/4-wire)	2(1)	
I^2C™	2	
Digital Signal Modulator	Yes	
Parallel Communications (EPMP/PSP)	Yes	
JTAG Boundary Scan	Yes	
12-Bit Pipeline Analog-to-Digital Converter (A/D) (input channels)	50	
Sigma-Delta Analog-to-Digital Converter (A/D) (differential channels)	2	
Digital-to-Analog Converter (DAC)	2	
Operational Amplifiers	2	
Analog Comparators	3	
CTMU Interface	Yes	
LCD Controller (available pixels)	472 (59 SEG x 8 COM)	
Resets (and delays)	Core POR, VDD POR, VBAT POR, BOR, RESET Instruction, $\overline{MCLR}$, WDT, Illegal Opcode, REPEAT Instruction, Hardware Traps, Configuration Word Mismatch (OST, PLL Lock)	
Instruction Set	76 Base Instructions, Multiple Addressing Mode Variations	
Packages	100-Pin TQFP and 121-Pin BGA	

Note 1: Peripherals are accessible through remappable pins.

FIGURE 1-1: PIC24FJ128GC010 FAMILY GENERAL BLOCK DIAGRAM



Note 1: Not all I/O pins or features are implemented on all device pinout configurations. See Table 1-3 for specific implementations by pin count.
2: These peripheral I/Os are only accessible through remappable pins.

TABLE 1-3: PIC24FJ128GC010 FAMILY PINOUT DESCRIPTION

Pin Function	Pin Number/Grid Locator			I/O	Input Buffer	Description
	64-Pin TQFP/QFN	100-Pin TQFP	121-Pin BGA			
AN0	16	25	K2	I	ANA	12-Bit Pipeline A/D Converter Inputs.
AN1	15	24	K1	I	ANA	
AN2	14	23	J2	I	ANA	
AN3	13	22	J1	I	ANA	
AN4	12	21	H2	I	ANA	
AN5	11	20	H1	I	ANA	
AN6	17	26	L1	I	ANA	
AN7	18	27	J3	I	ANA	
AN8	—	6	D1	I	ANA	
AN9	—	8	E2	I	ANA	
AN10	32	50	L11	I	ANA	
AN11	31	49	L10	I	ANA	
AN12	27	41	J7	I	ANA	
AN13	28	42	L7	I	ANA	
AN14	29	43	K7	I	ANA	
AN15	30	44	L8	I	ANA	
AN16	—	9	E1	I	ANA	
AN17	4	10	E3	I	ANA	
AN18	5	11	F4	I	ANA	
AN19	6	12	F2	I	ANA	
AN20	55	84	C7	I	ANA	
AN21	—	19	G2	I	ANA	
AN22	—	92	B5	I	ANA	
AN23	—	91	C5	I	ANA	
AN24	43	69	E10	I	ANA	
AN25	50	77	A10	I	ANA	
AN26	—	38	J6	I	ANA	
AN27	—	39	L6	I	ANA	
AN28	—	47	L9	I	ANA	
AN29	—	48	K9	I	ANA	
AN30	33	51	K10	I	ANA	
AN31	—	52	K11	I	ANA	
AN32	—	53	J10	I	ANA	
AN33	—	1	B2	I	ANA	
AN34	54	83	D7	I	ANA	
AN35	49	76	A11	I	ANA	
AN36	—	60	G11	I	ANA	
AN37	—	61	G9	I	ANA	
AN38	—	66	E11	I	ANA	
AN39	—	67	E8	I	ANA	
AN40	42	68	E9	I	ANA	

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

TABLE 1-3: PIC24FJ128GC010 FAMILY PINOUT DESCRIPTION (CONTINUED)

Pin Function	Pin Number/Grid Locator			I/O	Input Buffer	Description
	64-Pin TQFP/QFN	100-Pin TQFP	121-Pin BGA			
AN41	44	70	D11	I	ANA	12-Bit Pipeline A/D Converter Inputs.
AN42	45	71	C11	I	ANA	
AN43	46	72	D9	I	ANA	
AN44	51	78	B9	I	ANA	
AN45	—	79	A9	I	ANA	
AN46	—	80	D8	I	ANA	
AN47	52	81	C8	I	ANA	
AN48	53	82	B8	I	ANA	
AN49	8	14	F3	I	ANA	
AVDD	19	30	J4	P	—	Positive Supply for Analog modules.
AVREF+	16	25, 29	K2, K3	I	ANA	Pipeline A/D Reference Voltage (high) Input.
AVREF-	15	24, 28	K1, L2	I	ANA	Pipeline A/D Reference Voltage (low) Input.
AVss	20	31	L3	P	—	Ground Reference for Analog modules.
BGBUF1	16	25	K2	O	—	Buffered Band Gap Reference 1 Voltage Output.
BGBUF2	4	10	E3	O	—	Buffered Band Gap Reference 2 Voltage Output.
C1INA	11	20	H1	I	ANA	Comparator 1 Input A.
C1INB	12	21	H2	I	ANA	Comparator 1 Input B.
C1INC	5	11	F4	I	ANA	Comparator 1 Input C.
C1IND	4	10	E3	I	ANA	Comparator 1 Input D.
C2INA	13	22	J1	I	ANA	Comparator 2 Input A.
C2INB	14	23	J2	I	ANA	Comparator 2 Input B.
C2INC	8	14	F3	I	ANA	Comparator 2 Input C.
C2IND	6	12	F2	I	ANA	Comparator 2 Input D.
C3INA	55	84	C7	I	ANA	Comparator 3 Input A.
C3INB	54	83	D7	I	ANA	Comparator 3 Input B.
C3INC	45	71	C11	I	ANA	Comparator 3 Input C.
C3IND	44	70	D11	I	ANA	Comparator 3 Input D.
CH0+	22	33	L4	I	ANA	Sigma-Delta A/D Converter Channel 0 Non-Inverting Analog Input.
CH0-	23	34	H5	I	ANA	Sigma-Delta A/D Converter Channel 0 Inverting Analog Input.
CH1+	24	35	K5	I	ANA	Sigma-Delta A/D Converter Channel 1 Non-Inverting Analog Input.
CH1-	25	36	L5	I	ANA	Sigma-Delta A/D Converter Channel 1 Inverting Analog Input.
CLKI	39	63	F9	I	ANA	Main Clock Input Connection.
CLKO	40	64	F11	O	—	System Clock Output.

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

TABLE 1-3: PIC24FJ128GC010 FAMILY PINOUT DESCRIPTION (CONTINUED)

Pin Function	Pin Number/Grid Locator			I/O	Input Buffer	Description
	64-Pin TQFP/QFN	100-Pin TQFP	121-Pin BGA			
CN2	16	25	K2	I	ST	Interrupt-on-Change Inputs.
CN3	15	24	K1	I	ST	
CN4	14	23	J2	I	ST	
CN5	13	22	J1	I	ST	
CN6	12	21	H2	I	ST	
CN7	11	20	H1	I	ST	
CN8	4	10	E3	I	ST	
CN9	5	11	F4	I	ST	
CN10	6	12	F2	I	ST	
CN11	8	14	F3	I	ST	
CN12	30	44	L8	I	ST	
CN13	52	81	C8	I	ST	
CN14	53	82	B8	I	ST	
CN15	54	83	D7	I	ST	
CN16	55	84	C7	I	ST	
CN17	31	49	L10	I	ST	
CN18	32	50	L11	I	ST	
CN19	—	80	D8	I	ST	
CN20	—	47	L9	I	ST	
CN21	—	48	K9	I	ST	
CN22	40	64	F11	I	ST	
CN23	39	63	F9	I	ST	
CN24	17	26	L1	I	ST	
CN25	18	27	J3	I	ST	
CN30	27	41	J7	I	ST	
CN31	28	42	L7	I	ST	
CN32	29	43	K7	I	ST	
CN33	—	17	G3	I	ST	
CN34	—	38	J6	I	ST	
CN35	—	58	H11	I	ST	
CN36	—	59	G10	I	ST	
CN37	—	60	G11	I	ST	
CN38	—	61	G9	I	ST	
CN39	—	91	C5	I	ST	
CN40	—	92	B5	I	ST	
CN41	—	28	L2	I	ST	
CN42	—	29	K3	I	ST	
CN43	—	66	E11	I	ST	
CN44	—	67	E8	I	ST	
CN45	—	6	D1	I	ST	
CN46	—	7	E4	I	ST	
CN47	—	8	L11	I	ST	
CN48	—	9	E1	I	ST	
CN49	46	72	D9	I	ST	

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

TABLE 1-3: PIC24FJ128GC010 FAMILY PINOUT DESCRIPTION (CONTINUED)

Pin Function	Pin Number/Grid Locator			I/O	Input Buffer	Description
	64-Pin TQFP/QFN	100-Pin TQFP	121-Pin BGA			
CN50	49	76	A11	I	ST	Interrupt-on-Change Inputs.
CN51	50	77	A10	I	ST	
CN52	51	78	B9	I	ST	
CN53	42	68	E9	I	ST	
CN54	43	69	E10	I	ST	
CN55	44	70	D11	I	ST	
CN56	45	71	C11	I	ST	
CN57	—	79	A9	I	ST	
CN58	60	93	A4	I	ST	
CN59	61	94	B4	I	ST	
CN60	62	98	B3	I	ST	
CN61	63	99	A2	I	ST	
CN62	64	100	A1	I	ST	
CN63	1	3	D3	I	ST	
CN64	2	4	C1	I	ST	
CN65	3	5	D2	I	ST	
CN66	—	18	G1	I	ST	
CN67	—	19	G2	I	ST	
CN68	58	87	B6	I	ST	
CN69	59	88	A6	I	ST	
CN70	—	52	K11	I	ST	
CN71	33	51	K10	I	ST	
CN72	37	57	H10	I	ST	
CN73	36	56	J11	I	ST	
CN74	—	53	J10	I	ST	
CN75	—	40	K6	I	ST	
CN76	—	39	L6	I	ST	
CN77	—	90	A5	I	ST	
CN78	—	89	E6	I	ST	
CN79	—	96	C3	I	ST	
CN80	—	97	A3	I	ST	
CN81	—	95	C4	I	ST	
CN82	—	1	B2	I	ST	
CN83	34	54	H8	I	ST	
COM0	63	99	A2	O	—	LCD Driver Common Outputs.
COM1	62	98	B3	O	—	
COM2	61	94	B4	O	—	
COM3	60	93	A4	O	—	
COM4	59	88	A6	O	—	
COM5	27	41	J7	O	—	
COM6	18	27	J3	O	—	
COM7	58	87	B6	O	—	

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

TABLE 1-3: PIC24FJ128GC010 FAMILY PINOUT DESCRIPTION (CONTINUED)

Pin Function	Pin Number/Grid Locator			I/O	Input Buffer	Description
	64-Pin TQFP/QFN	100-Pin TQFP	121-Pin BGA			
CS1	45	71	C11	I/O	ST/TTL	Parallel Master Port Chip Select 1 Strobe (shared with PMA14).
CS2	44	70	D11	O	—	Parallel Master Port Chip Select 2 Strobe (shared with PMA15).
CTCMP	14	23	J2	I	ANA	CTMU Comparator 2 Input (Pulse mode).
CTED0	—	17	G3	I	ANA	CTMU External Edge Inputs.
CTED1	28	42	L7	I	ANA	
CTED2	27	41	J7	I	ANA	
CTED3	—	1	B2	I	ANA	
CTED4	1	3	D3	I	ANA	
CTED5	29	43	K7	I	ANA	
CTED6	30	44	L8	I	ANA	
CTED7	—	40	K6	I	ANA	
CTED8	64	100	A1	I	ANA	
CTED9	63	99	A2	I	ANA	
CTED10	—	97	A3	I	ANA	
CTED11	—	95	C4	I	ANA	
CTED12	15	24	K1	I	ANA	
CTED13	14	23	J2	I	ANA	
CTPLS	29	43	K7	O	—	CTMU Pulse Output.
CV_{REF}	32	50	L11	O	—	Comparator Voltage Reference Output.
CV_{REF+}	16	25, 29	K2, K3	I	ANA	Comparator Reference Voltage (high) Input.
CV_{REF-}	15	24, 28	K1, L2	I	ANA	Comparator Reference Voltage (low) Input.
D+	37	57	H10	I/O	—	USB Differential Plus Line (internal transceiver).
D-	36	56	J11	I/O	—	USB Differential Minus Line (internal transceiver).
DAC1	8	14	F3	O	—	DAC Converter 1 Analog Output.
DAC2	28	42	L7	O	—	DAC Converter 2 Analog Output.
DMH	46	72	D9	O	—	D- External Pull-up Control Output.
DMLN	42	68	E9	O	—	D- External Pull-Down Control Output.
DPH	50	77	A10	O	—	D+ External Pull-up Control Output.
DPLN	43	69	E10	O	—	D+ External Pull-Down Control Output.
DV_{REF+}	16	25	K2	I	ANA	DAC Positive Reference Input.
INT0	46	72	D9	I	ST	External Interrupt Input 0.
LCDBIAS0	3	5	D2	I	ANA	Bias Inputs for LCD Driver Charge Pump.
LCDBIAS1	2	4	C1	I	ANA	
LCDBIAS2	1	3	D3	I	ANA	
LCDBIAS3	17	26	L1	I	ANA	
HLVDIN	64	100	A1	I	ANA	High/Low-Voltage Detect Input.
$\overline{MCLR}$	7	13	F1	I	ST	Master Clear (device Reset) Input. This line is brought low to cause a Reset.
OCTRIG1	42	68	E9	I	ST	Output Compare External Trigger 1 Input.
OCTRIG2	—	66	E11	I	ST	Output Compare External Trigger 2 Input.

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

TABLE 1-3: PIC24FJ128GC010 FAMILY PINOUT DESCRIPTION (CONTINUED)

Pin Function	Pin Number/Grid Locator			I/O	Input Buffer	Description
	64-Pin TQFP/QFN	100-Pin TQFP	121-Pin BGA			
OA1NA	12	21	H2	I	ANA	Op Amp 1 Negative (inverting) Inputs.
OA1NB	53	82	B8	I	ANA	
OA1NC	—	8	E2	I	ANA	
OA1ND	6	12	F2	I	ANA	
OA1NE	5	11	F4	I	ANA	
OA1OUT	11	20	H1	O	—	Op Amp 1 (analog) Output (digital output in Comparator mode).
OA1PA	8	14	F3	I	ANA	Op Amp 1 Positive (non-inverting) Inputs.
OA1PB	4	10	E3	I	ANA	
OA1PC	54	83	D7	I	ANA	
OA1PD	17	26	L1	I	ANA	
OA1PE	52	81	C8	I	ANA	
OA2NA	46	72	D9	I	ANA	Op Amp 2 Negative (inverting) Inputs.
OA2NB	50	77	A10	I	ANA	
OA2NC	14	23	J2	I	ANA	
OA2ND	31	49	L10	I	ANA	
OA2NE	29	43	K7	I	ANA	
OA2OUT	13	22	J1	O	—	Op Amp 2 (analog) Output (digital output in Comparator mode).
OA2PA	45	71	C11	I	ANA	Op Amp 2 Positive (non-inverting) Inputs.
OA2PB	15	24	K1	I	ANA	
OA2PC	32	50	L11	I	ANA	
OA2PD	28	42	L7	I	ANA	
OA2PE	51	78	B9	I	ANA	
OSCI	39	63	F9	I	ANA	Main Oscillator Input Connection.
OSCO	40	64	F11	O	—	Main Oscillator Output Connection.
PGEC1	15	24	K1	I/O	ST	In-Circuit Debugger/Emulator/ICSP™ Programming Clock.
PGEC2	17	26	L1	I/O	ST	
PGEC3	11	20	H1	I/O	ST	
PGED1	16	25	K2	I/O	ST	In-Circuit Debugger/Emulator/ICSP Programming Data.
PGED2	18	27	J3	I/O	ST	
PGED3	12	21	H2	I/O	ST	
PMA0	30	44	L8	I/O	ST	Parallel Master Port Address Bit 0 Input (Buffered Slave modes) and Output (Master modes).
PMA1	29	43	K7	I/O	ST	Parallel Master Port Address Bit 1 Input (Buffered Slave modes) and Output (Master modes).
PMA2	8	14	F3	O	—	Parallel Master Port Address (bits<22:2>).
PMA3	6	12	F2	O	—	
PMA4	5	11	F4	O	—	
PMA5	4	10	E3	O	—	
PMA6	16	29	K3	O	—	
PMA7	14	28	L2	O	—	

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

TABLE 1-3: PIC24FJ128GC010 FAMILY PINOUT DESCRIPTION (CONTINUED)

Pin Function	Pin Number/Grid Locator			I/O	Input Buffer	Description
	64-Pin TQFP/QFN	100-Pin TQFP	121-Pin BGA			
PMA8	32	50	L11	O	—	Parallel Master Port Address (bits<22:2>).
PMA9	31	49	L10	O	—	
PMA10	28	42	L7	O	—	
PMA11	27	41	J7	O	—	
PMA12	33	51	K10	O	—	
PMA13	42	68	E9	O	—	
PMA14	45	71	C11	O	—	
PMA15	44	70	D11	O	—	
PMA16	—	95	C4	O	—	
PMA17	—	92	B5	O	—	
PMA18	—	40	K6	O	—	
PMA19	—	19	G2	O	—	
PMA20	—	59	G10	O	—	
PMA21	—	60	G11	O	—	
PMA22	—	66	E11	O	—	
PMACK1	50	77	A10	I	ST/TTL	Parallel Master Port Acknowledge Input 1.
PMACK2	43	69	E10	I	ST/TTL	Parallel Master Port Acknowledge Input 2.
PMBE0	51	78	B9	O	—	Parallel Master Port Byte Enable 0 Strobe.
PMBE1	—	67	E8	O	—	Parallel Master Port Byte Enable 1 Strobe.
PMCS1	—	18	G1	I/O	ST/TTL	Parallel Master Port Chip Select 1 Strobe.
PMCS2	—	9	K10	O	—	Parallel Master Port Chip Select 2 Strobe.
PMD0	60	93	A4	I/O	ST/TTL	Parallel Master Port Data (Demultiplexed Master mode) or Address/Data (Multiplexed Master modes).
PMD1	61	94	B4	I/O	ST/TTL	
PMD2	62	98	B3	I/O	ST/TTL	
PMD3	63	99	A2	I/O	ST/TTL	
PMD4	64	100	A1	I/O	ST/TTL	
PMD5	1	3	D3	I/O	ST/TTL	
PMD6	2	4	C1	I/O	ST/TTL	
PMD7	3	5	D2	I/O	ST/TTL	
PMD8	—	90	A5	I/O	ST/TTL	
PMD9	—	89	E6	I/O	ST/TTL	
PMD10	—	88	A6	I/O	ST/TTL	
PMD11	—	87	B6	I/O	ST/TTL	
PMD12	—	79	A9	I/O	ST/TTL	
PMD13	—	80	D8	I/O	ST/TTL	
PMD14	—	83	D7	I/O	ST/TTL	
PMD15	—	84	C7	I/O	ST/TTL	
PMRD	53	82	B8	O	—	Parallel Master Port Read Strobe.
PMWR	52	81	C8	O	—	Parallel Master Port Write Strobe.

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

TABLE 1-3: PIC24FJ128GC010 FAMILY PINOUT DESCRIPTION (CONTINUED)

Pin Function	Pin Number/Grid Locator			I/O	Input Buffer	Description
	64-Pin TQFP/QFN	100-Pin TQFP	121-Pin BGA			
RA0	—	17	G3	I/O	ST	PORTA Digital I/O.
RA1	—	38	J6	I/O	ST	
RA2	—	58	H11	I/O	ST	
RA3	—	59	G10	I/O	ST	
RA4	—	60	G11	I/O	ST	
RA5	—	61	G9	I/O	ST	
RA6	—	91	C5	I/O	ST	
RA7	—	92	B5	I/O	ST	
RA9	—	28	L2	I/O	ST	
RA10	—	29	K3	I/O	ST	
RA14	—	66	E11	I/O	ST	
RA15	—	67	E8	I/O	ST	
RB0	16	25	K2	I/O	ST	PORTB Digital I/O.
RB1	15	24	K1	I/O	ST	
RB2	14	23	J2	I/O	ST	
RB3	13	22	J1	I/O	ST	
RB4	12	21	H2	I/O	ST	
RB5	11	20	H1	I/O	ST	
RB6	17	26	L1	I/O	ST	
RB7	18	27	J3	I/O	ST	
RB12	27	41	J7	I/O	ST	
RB13	28	42	L7	I/O	ST	
RB14	29	43	K7	I/O	ST	
RB15	30	44	L8	I/O	ST	
RC1	—	6	D1	I/O	ST	PORTC Digital I/O.
RC2	—	7	E4	I/O	ST	
RC3	—	8	E2	I/O	ST	
RC4	—	9	E1	I/O	ST	
RC12	39	63	F9	I/O	ST	
RC13	47	73	C10	I	ST	
RC14	48	74	B11	I	ST	
RC15	40	64	F11	I/O	ST	

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

TABLE 1-3: PIC24FJ128GC010 FAMILY PINOUT DESCRIPTION (CONTINUED)

Pin Function	Pin Number/Grid Locator			I/O	Input Buffer	Description
	64-Pin TQFP/QFN	100-Pin TQFP	121-Pin BGA			
RD0	46	72	D9	I/O	ST	PORTD Digital I/O.
RD1	49	76	A11	I/O	ST	
RD2	50	77	A10	I/O	ST	
RD3	51	78	B9	I/O	ST	
RD4	52	81	C8	I/O	ST	
RD5	53	82	B8	I/O	ST	
RD6	54	83	D7	I/O	ST	
RD7	55	84	C7	I/O	ST	
RD8	42	68	E9	I/O	ST	
RD9	43	69	E10	I/O	ST	
RD10	44	70	D11	I/O	ST	
RD11	45	71	C11	I/O	ST	
RD12	—	79	A9	I/O	ST	
RD13	—	80	D8	I/O	ST	
RD14	—	47	L9	I/O	ST	
RD15	—	48	K9	I/O	ST	
RE0	60	93	A4	I/O	ST	PORTE Digital I/O.
RE1	61	94	B4	I/O	ST	
RE2	62	98	B3	I/O	ST	
RE3	63	99	A2	I/O	ST	
RE4	64	100	A1	I/O	ST	
RE5	1	3	D3	I/O	ST	
RE6	2	4	C1	I/O	ST	
RE7	3	5	D2	I/O	ST	
RE8	—	18	G1	I/O	ST	
RE9	—	19	G2	I/O	ST	
REFO	30	44	L8	O	—	Reference Clock Output.
RF0	58	87	B6	I/O	ST	PORTF Digital I/O.
RF1	59	88	A6	I/O	ST	
RF2	—	52	K11	I/O	ST	
RF3	33	51	K10	I/O	ST	
RF4	31	49	L10	I/O	ST	
RF5	32	50	L11	I/O	ST	
RF7	34	54	H8	I/O	ST	
RF8	—	53	J10	I/O	ST	
RF12	—	40	K6	I/O	ST	
RF13	—	39	L6	I/O	ST	

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

TABLE 1-3: PIC24FJ128GC010 FAMILY PINOUT DESCRIPTION (CONTINUED)

Pin Function	Pin Number/Grid Locator			I/O	Input Buffer	Description
	64-Pin TQFP/QFN	100-Pin TQFP	121-Pin BGA			
RG0	—	90	A5	I/O	ST	PORTG Digital I/O.
RG1	—	89	E6	I/O	ST	
RG2	37	57	H10	I/O	ST	
RG3	36	56	J11	I/O	ST	
RG6	4	10	E3	I/O	ST	
RG7	5	11	F4	I/O	ST	
RG8	6	12	F2	I/O	ST	
RG9	8	14	F3	I/O	ST	
RG12	—	96	C3	I/O	ST	
RG13	—	97	A3	I/O	ST	
RG14	—	95	C4	I/O	ST	
RG15	—	1	B2	I/O	ST	
RP0	16	25	K2	I/O	ST	Remappable Peripheral (input or output).
RP1	15	24	K1	I/O	ST	
RP2	42	68	E9	I/O	ST	
RP4	43	69	E10	I/O	ST	
RP5	—	48	K9	I/O	ST	
RP6	17	26	L1	I/O	ST	
RP7	18	27	J3	I/O	ST	
RP10	31	49	L10	I/O	ST	
RP11	46	72	D9	I/O	ST	
RP12	45	71	C11	I/O	ST	
RP13	14	23	J2	I/O	ST	
RP14	29	43	K7	I/O	ST	
RP15	—	53	J10	I/O	ST	
RP16	33	51	K10	I/O	ST	
RP17	32	50	L11	I/O	ST	
RP18	11	20	H1	I/O	ST	
RP19	6	12	F2	I/O	ST	
RP20	53	82	B8	I/O	ST	
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	

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

TABLE 1-3: PIC24FJ128GC010 FAMILY PINOUT DESCRIPTION (CONTINUED)

Pin Function	Pin Number/Grid Locator			I/O	Input Buffer	Description
	64-Pin TQFP/QFN	100-Pin TQFP	121-Pin BGA			
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	50, 58	H11, L11	I/O	I^2C	I2C2 Synchronous Serial Clock Input/Output.
SCLKI	48	74	B11	I	ST	Secondary Oscillator Digital Clock Input.
SDA1	43	67	E8	I/O	I^2C	I2C1 Data Input/Output.
SDA2	31	49, 59	G10, L10	I/O	I^2C	I2C2 Data Input/Output.
SEG0	4	10	E3	O	—	LCD Driver Segment Outputs.
SEG1	8	14	F3	O	—	
SEG2	11	20	H1	O	—	
SEG3	12	21	H2	O	—	
SEG4	13	22	J1	O	—	
SEG5	14	23	J2	O	—	
SEG6	15	24	K1	O	—	
SEG7	16	25	K2	O	—	
SEG8	29	43	K7	O	—	
SEG9	30	44	L8	O	—	
SEG10	31	49	L10	O	—	
SEG11	32	50	L11	O	—	
SEG12	33	51	K10	O	—	
SEG13	42	68	E9	O	—	
SEG14	43	69	E10	O	—	
SEG15	44	70	D11	O	—	
SEG16	45	71	C11	O	—	
SEG17	46	72	D9	O	—	
SEG18	27	41	J7	O	—	
SEG19	28	42	L7	O	—	
SEG20	49	76	A11	O	—	

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

TABLE 1-3: PIC24FJ128GC010 FAMILY PINOUT DESCRIPTION (CONTINUED)

Pin Function	Pin Number/Grid Locator			I/O	Input Buffer	Description
	64-Pin TQFP/QFN	100-Pin TQFP	121-Pin BGA			
SEG21	50	77	A10	O	—	LCD Driver Segment Outputs.
SEG22	51	78	B9	O	—	
SEG23	52	81	C8	O	—	
SEG24	53	82	B8	O	—	
SEG25	54	83	D7	O	—	
SEG26	55	84	C7	O	—	
SEG27	58	87	B6	O	—	
SEG28	—	61	G9	O	—	
SEG29	—	60	G11	O	—	
SEG30	18	27	J3	O	—	
SEG31	—	38	J6	O	—	
SEG32	—	6	D1	O	—	
SEG33	—	8	E2	O	—	
SEG34	—	18	G1	O	—	
SEG35	—	19	G2	O	—	
SEG36	—	28	L2	O	—	
SEG37	—	29	K3	O	—	
SEG38	—	47	L9	O	—	
SEG39	—	48	K9	O	—	
SEG40	—	52	K11	O	—	
SEG41	—	53	J10	O	—	
SEG42	—	66	E11	O	—	
SEG43	—	67	E8	O	—	
SEG44	—	79	A9	O	—	
SEG45	—	80	D8	O	—	
SEG46	—	89	E6	O	—	
SEG47	59	88	A6	O	—	
SEG48	—	17	G3	O	—	
SEG49	—	90	A5	O	—	
SEG50	—	1	B2	O	—	
SEG51	—	7	E4	O	—	
SEG52	—	9	E1	O	—	
SEG53	—	39	L6	O	—	
SEG54	—	40	K6	O	—	
SEG55	—	58	H11	O	—	
SEG56	—	59	G10	O	—	
SEG57	—	91	C5	O	—	
SEG58	—	92	B5	O	—	
SEG59	—	95	C4	O	—	
SEG60	—	96	C3	O	—	
SEG61	—	97	A3	O	—	
SEG62	64	100	A1	O	—	

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

TABLE 1-3: PIC24FJ128GC010 FAMILY PINOUT DESCRIPTION (CONTINUED)

Pin Function	Pin Number/Grid Locator			I/O	Input Buffer	Description
	64-Pin TQFP/QFN	100-Pin TQFP	121-Pin BGA			
SOSCI	47	73	C10	I	ANA	Secondary Oscillator Input.
SOSCO	48	74	B11	O	ANA	Secondary Oscillator Output.
SVDD	26	37	J5	P	—	Positive Supply for Sigma-Delta A/D Converter.
SVREF+	24	35	K5	I	ANA	Sigma-Delta A/D Converter Voltage Reference (high).
SVREF-	25	36	L5	I	ANA	Sigma-Delta A/D Converter Voltage Reference (low).
SVss	21	32	K4	P	—	Ground Reference for Sigma-Delta A/D Converter.
T1CK	27	41	J7	I	ST	External Timer1 Clock Input.
T2CK	30	44	L8	I	ST	External Timer2 Clock Input.
T3CK	31	49	L10	I	ST	External Timer3 Clock Input.
T4CK	14	23	J2	I	ST	External Timer4 Clock Input.
T5CK	4	10	E3	I	ST	External Timer5 Clock Input.
TCK	27	38	J6	I	ST	JTAG Test Clock/Programming Clock Input.
TDI	28	60	G11	I	ST	JTAG Test Data/Programming Data Input.
TDO	29	61	G9	O	—	JTAG Test Data Output.
TMS	45	17	G3	I	ST	JTAG Test Mode Select Input.
USBID	33	51	K10	I	ST	USB OTG ID (OTG mode only).
USBOE	12	21	H2	O	—	USB Output Enable Control (for external transceiver).
VBAT	57	86	A7	P	—	Backup Battery (B+) Input.
VBUS	34	54	H8	P	—	USB VBUS Connection (5V nominal).
VCAP	56	85	B7	P	—	External Filter Capacitor Connection.
VCMPST1	58	87	B6	I	ST	USB VBUS External Comparator Input 1.
VCMPST2	59	88	A6	I	ST	USB VBUS External Comparator Input 2.
VCMPST3	46	72	D9	I	ST	USB VBUS External Comparator Input 3.
VDD	10, 38	2, 16, 46,62	C2,G5, K8, F8	P	—	Positive Supply for Peripheral Digital Logic and I/O Pins.
VLCAP1	5	11	F4	P	ANA	LCD Drive Charge Pump Capacitor Pins.
VLCAP2	6	12	F2	P	ANA	
VMIO	14	23	J2	I/O	ST	USB Differential Minus Input/Output (external transceiver).
VPIO	13	22	J1	I/O	ST	USB Differential Plus Input/Output (external transceiver).
VSS	9, 41	15, 45, 65, 75	F5, G7, F10, B10	P	—	Ground Reference for Logic and I/O Pins.
VUSB3V3	35	55	H9	P	—	USB Transceiver Power Input Voltage (3.3V nominal)

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

NOTES:

2.0 GUIDELINES FOR GETTING STARTED WITH 16-BIT MICROCONTROLLERS

2.1 Basic Connection Requirements

Getting started with the PIC24FJ128GC010 family family of 16-bit microcontrollers requires attention to a minimal set of device pin connections before proceeding with development.

The following pins must always be connected:

- All VDD and VSS pins
  (see **Section 2.2 "Power Supply Pins"**)
- All analog power pins (AVDD, SVDD, AVSS and SVSS), regardless of whether or not the analog device features are used
  (see **Section 2.2 "Power Supply Pins"**)
- The USB transceiver supply, VUSB3V3, regardless of whether or not the USB module is used
  (see **Section 2.2 "Power Supply Pins"**)
- $\overline{\text{MCLR}}$ pin
  (see **Section 2.3 "Master Clear (MCLR) Pin"**)
- VCAP pin
  (see **Section 2.4 "Voltage Regulator Pin (VCAP)"**)

These pins must also be connected if they are being used in the end application:

- PGECx/PGEDx pins used for In-Circuit Serial Programming™ (ICSP™) and debugging purposes
  (see **Section 2.5 "ICSP Pins"**)
- OSCI and OSCO pins when an external oscillator source is used
  (see **Section 2.6 "External Oscillator Pins"**)

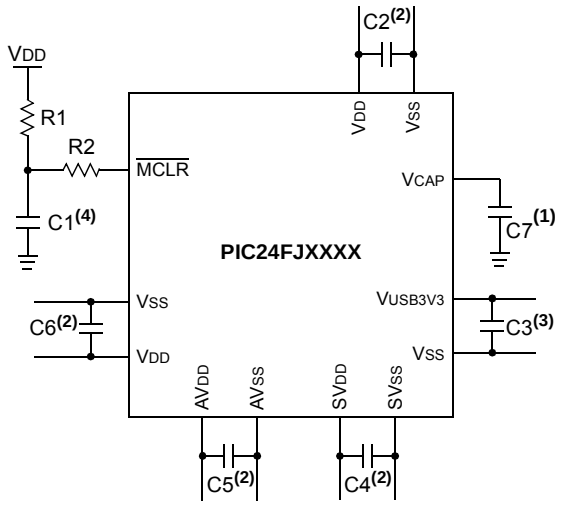
Additionally, the following pins may be required:

- Any voltage reference pins used when external voltage reference for analog modules is implemented (AVREF+/AVREF-, CVREF+/CVREF-, DVREF+ and SVREF+/SVREF-)

Note:	All analog power supply and return pins must always be connected, regardless of whether any of the analog modules are being used.

The minimum mandatory connections are shown in Figure 2-1.

FIGURE 2-1: RECOMMENDED MINIMUM CONNECTIONS



Key (all values are recommendations):

C1 through C6: 0.1 µF, 20V ceramic

C7: 10 µF, 6.3V or greater, tantalum or ceramic

R1: 10 kΩ

R2: 100Ω to 470Ω

Note 1: See **Section 2.4 "Voltage Regulator Pin (VCAP)"** for details on selecting the proper capacitor for VCAP.

2: The example shown is for a PIC24F device with five power and ground pairs (including analog and USB). Other devices may have more or less pairs; adjust the number of decoupling capacitors appropriately.

3: See **Section 19.1 "Hardware Configuration"** for details on connecting the pins for USB operation.

4: C1 is optional, see **Section 2.3 "Master Clear (MCLR) Pin"** and **Section 2.5 "ICSP Pins"** for more information.

2.2 Power Supply Pins

2.2.1 DECOUPLING CAPACITORS

The use of decoupling capacitors on every pair of power supply pins is required. This includes digital supply (V_{DD} and V_{SS}) and all analog supplies (AV_{DD}, SV_{DD}, AV_{SS} and SV_{SS}).

Consider the following criteria when using decoupling capacitors:

- **Value and type of capacitor:** A 0.1 μF (100 nF), 10-20V capacitor is recommended. The capacitor should be a low-ESR device with a resonance frequency in the range of 200 MHz and higher. Ceramic capacitors are recommended.
- **Placement on the printed circuit board:** The decoupling capacitors should be placed as close to the pins as possible. It is recommended to place the capacitors on the same side of the board as the device. If space is constricted, the capacitor can be placed on another layer on the PCB using a via; however, ensure that the trace length from the pin to the capacitor is no greater than 0.25 inch (6 mm).
- **Handling high-frequency noise:** If the board is experiencing high-frequency noise (upward of tens of MHz), add a second ceramic type capacitor in parallel to the above described decoupling capacitor. The value of the second capacitor can be in the range of 0.01 μF to 0.001 μF. Place this second capacitor next to each primary decoupling capacitor. In high-speed circuit designs, consider implementing a pair of capacitances as close to the power and ground pins as possible (e.g., 0.1 μF in parallel with 0.001 μF).
- **Maximizing performance:** On the board layout from the power supply circuit, run the power and return traces to the decoupling capacitors first, and then to the device pins. This ensures that the decoupling capacitors are first in the power chain. Equally important is to keep the trace length between the capacitor and the power pins to a minimum, thereby reducing PCB trace inductance.

2.2.2 BULK CAPACITORS

On boards with power traces running longer than six inches in length, it is suggested to use a tank capacitor for integrated circuits including microcontrollers to supply a local power source. The value of the tank capacitor should be determined based on the trace resistance that connects the power supply source to the device, and the maximum current drawn by the device in the application. In other words, select the tank capacitor so that it meets the acceptable voltage sag at the device. Typical values range from 4.7 μF to 47 μF.

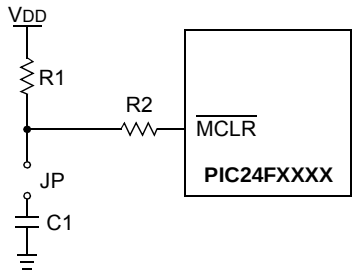
2.3 Master Clear ($\overline{MCLR}$) Pin

The $\overline{MCLR}$ pin provides two specific device functions: device Reset, and device programming and debugging. If programming and debugging are not required in the end application, a direct connection to V_{DD} may be all that is required. The addition of other components, to help increase the application's resistance to spurious Resets from voltage sags, may be beneficial. A typical configuration is shown in Figure 2-1. Other circuit designs may be implemented, depending on the application's requirements.

During programming and debugging, the resistance and capacitance that can be added to the pin must be considered. Device programmers and debuggers drive the $\overline{MCLR}$ pin. Consequently, specific voltage levels (V_{IH} and V_{IL}) and fast signal transitions must not be adversely affected. Therefore, specific values of R1 and C1 will need to be adjusted based on the application and PCB requirements. For example, it is recommended that the capacitor, C1, be isolated from the $\overline{MCLR}$ pin during programming and debugging operations by using a jumper (Figure 2-2). The jumper is replaced for normal run-time operations.

Any components associated with the $\overline{MCLR}$ pin should be placed within 0.25 inch (6 mm) of the pin.

FIGURE 2-2: EXAMPLE OF $\overline{MCLR}$ PIN CONNECTIONS



Note 1: R1 ≤ 10 kΩ is recommended. A suggested starting value is 10 kΩ. Ensure that the $\overline{MCLR}$ pin V_{IH} and V_{IL} specifications are met.

2: R2 ≤ 470Ω will limit any current flowing into $\overline{MCLR}$ from the external capacitor, C, in the event of $\overline{MCLR}$ pin breakdown, due to Electrostatic Discharge (ESD) or Electrical Overstress (EOS). Ensure that the $\overline{MCLR}$ pin V_{IH} and V_{IL} specifications are met.

2.4 Voltage Regulator Pin (VCAP)

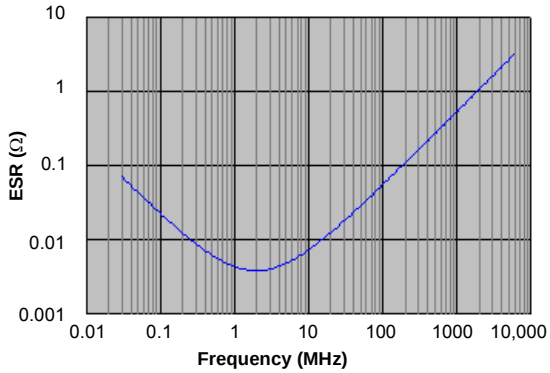
A low-ESR (< 5Ω) capacitor is required on the VCAP pin to stabilize the output voltage of the on-chip voltage regulator. The VCAP pin must not be connected to VDD and must use a capacitor of 10 µF connected to ground. The type can be ceramic or tantalum. Suitable examples of capacitors are shown in Table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

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 37.0 "Electrical Characteristics"** for additional information.

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

Refer to **Section 34.2 "On-Chip Voltage Regulator"** for details on connecting and using the on-chip regulator.

FIGURE 2-3: FREQUENCY vs. ESR PERFORMANCE FOR SUGGESTED VCAP



Note: Typical data measurement at 25°C, 0V DC bias.

TABLE 2-1: SUITABLE CAPACITOR EQUIVALENTS

Make	Part #	Nominal Capacitance	Base Tolerance	Rated Voltage	Temp. Range
TDK	C3216X7R1C106K	10 µF	±10%	16V	-55 to +125ºC
TDK	C3216X5R1C106K	10 µF	±10%	16V	-55 to +85ºC
Panasonic	ECJ-3YX1C106K	10 µF	±10%	16V	-55 to +125ºC
Panasonic	ECJ-4YB1C106K	10 µF	±10%	16V	-55 to +85ºC
Murata	GRM32DR71C106KA01L	10 µF	±10%	16V	-55 to +125ºC
Murata	GRM31CR61C106KC31L	10 µF	±10%	16V	-55 to +85ºC

2.4.1 CONSIDERATIONS FOR CERAMIC CAPACITORS

In recent years, large value, low-voltage, surface-mount ceramic capacitors have become very cost effective in sizes up to a few tens of microfarad. The low-ESR, small physical size and other properties make ceramic capacitors very attractive in many types of applications.

Ceramic capacitors are suitable for use with the internal voltage regulator of this microcontroller. However, some care is needed in selecting the capacitor to ensure that it maintains sufficient capacitance over the intended operating range of the application.

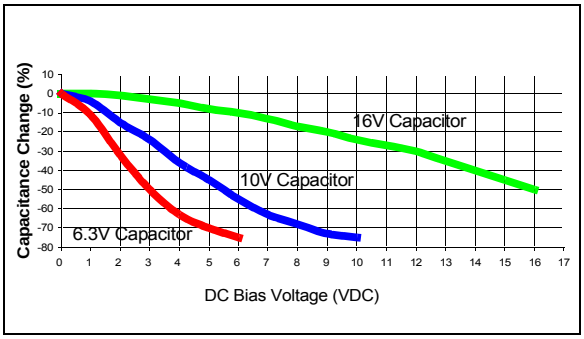
Typical low-cost, 10 μF ceramic capacitors are available in X5R, X7R and Y5V dielectric ratings (other types are also available, but are less common). The initial tolerance specifications for these types of capacitors are often specified as ±10% to ±20% (X5R and X7R), or -20%/+80% (Y5V). However, the effective capacitance that these capacitors provide in an application circuit will also vary based on additional factors, such as the applied DC bias voltage and the temperature. The total in-circuit tolerance is, therefore, much wider than the initial tolerance specification.

The X5R and X7R capacitors typically exhibit satisfactory temperature stability (ex: ±15% over a wide temperature range, but consult the manufacturer's data sheets for exact specifications). However, Y5V capacitors typically have extreme temperature tolerance specifications of +22%/-82%. Due to the extreme temperature tolerance, a 10 μF nominal rated Y5V type capacitor may not deliver enough total capacitance to meet minimum internal voltage regulator stability and transient response requirements. Therefore, Y5V capacitors are not recommended for use with the internal regulator if the application must operate over a wide temperature range.

In addition to temperature tolerance, the effective capacitance of large value ceramic capacitors can vary substantially, based on the amount of DC voltage applied to the capacitor. This effect can be very significant, but is often overlooked or is not always documented.

Typical DC bias voltage vs. capacitance graph for X7R type capacitors is shown in Figure 2-4.

FIGURE 2-4: DC BIAS VOLTAGE vs. CAPACITANCE CHARACTERISTICS



When selecting a ceramic capacitor to be used with the internal voltage regulator, it is suggested to select a high-voltage rating, so that the operating voltage is a small percentage of the maximum rated capacitor voltage. For example, choose a ceramic capacitor rated at 16V for the 2.5V or 1.8V core voltage. Suggested capacitors are shown in Table 2-1.

2.5 ICSP Pins

The PGECx and PGEDx pins are used for In-Circuit Serial Programming (ICSP) and debugging purposes. It is recommended to keep the trace length between the ICSP connector and the ICSP pins on the device as short as possible. If the ICSP connector is expected to experience an ESD event, a series resistor is recommended, with the value in the range of a few tens of ohms, not to exceed 100Ω.

Pull-up resistors, series diodes and capacitors on the PGECx and PGEDx pins are not recommended as they will interfere with the programmer/debugger communications to the device. If such discrete components are an application requirement, they should be removed from the circuit during programming and debugging. Alternatively, refer to the AC/DC characteristics and timing requirements information in the respective device Flash programming specification for information on capacitive loading limits and pin Voltage Input High (V_{IH}) and Voltage Input Low (V_{IL}) requirements.

For device emulation, ensure that the "Communication Channel Select" (i.e., PGECx/PGEDx pins), programmed into the device, matches the physical connections for the ICSP to the Microchip debugger/emulator tool.

The $\overline{MCLR}$ connection from the ICSP header should connect directly to the $\overline{MCLR}$ pin on the device. A capacitor to ground (C1 in Figure 2-2) is optional, but if used, may interfere with ICSP operation if the value exceeds 0.01 μF. In most cases, this capacitor is not required.

For more information on available Microchip development tools connection requirements, refer to **Section 35.0 "Development Support"**.

2.6 External Oscillator Pins

Many microcontrollers have options for at least two oscillators: a high-frequency primary oscillator and a low-frequency secondary oscillator (refer to **Section 9.0 "Oscillator Configuration"** for details).

The oscillator circuit should be placed on the same side of the board as the device. Place the oscillator circuit close to the respective oscillator pins with no more than 0.5 inch (12 mm) between the circuit components and the pins. The load capacitors should be placed next to the oscillator itself, on the same side of the board.

Use a grounded copper pour around the oscillator circuit to isolate it from surrounding circuits. The grounded copper pour should be routed directly to the MCU ground. Do not run any signal traces or power traces inside the ground pour. Also, if using a two-sided board, avoid any traces on the other side of the board where the crystal is placed.

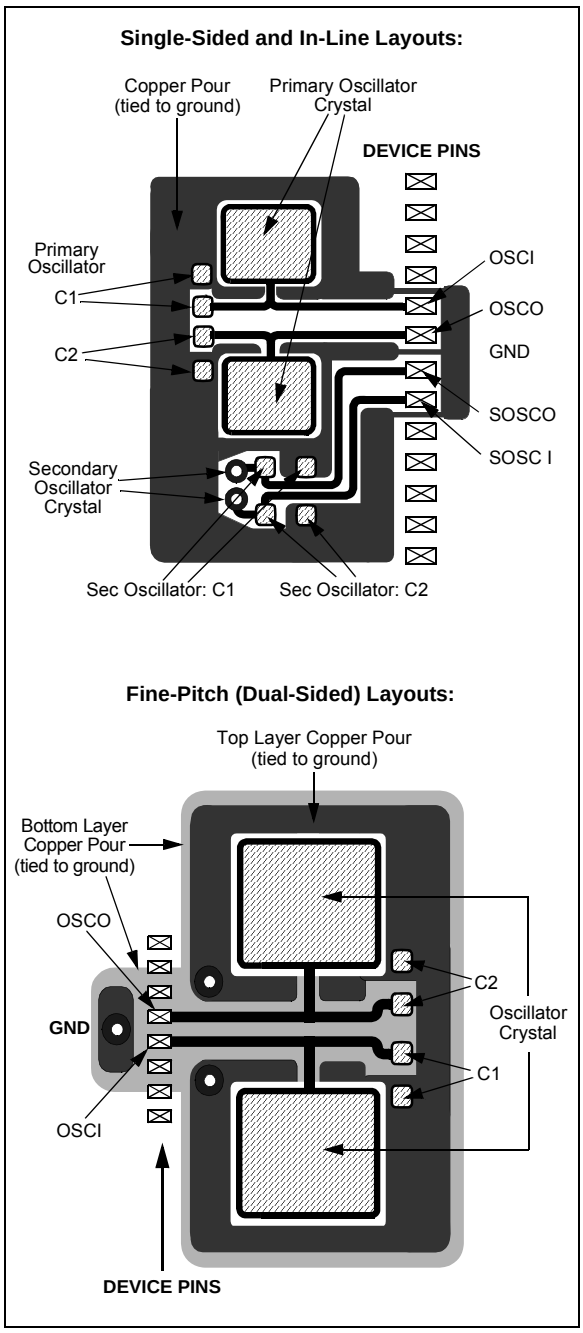
Layout suggestions are shown in Figure 2-5. In-line packages may be handled with a single-sided layout that completely encompasses the oscillator pins. With fine-pitch packages, it is not always possible to completely surround the pins and components. A suitable solution is to tie the broken guard sections to a mirrored ground layer. In all cases, the guard trace(s) must be returned to ground.

In planning the application's routing and I/O assignments, ensure that adjacent port pins, and other signals in close proximity to the oscillator, are benign (i.e., free of high frequencies, short rise and fall times and other similar noise).

For additional information and design guidance on oscillator circuits, please refer to these Microchip Application Notes, available at the corporate web site (www.microchip.com):

- *AN826, "Crystal Oscillator Basics and Crystal Selection for rfPIC™ and PICmicro® Devices"*
- *AN849, "Basic PICmicro® Oscillator Design"*
- *AN943, "Practical PICmicro® Oscillator Analysis and Design"*
- *AN949, "Making Your Oscillator Work"*

FIGURE 2-5: SUGGESTED PLACEMENT OF THE OSCILLATOR CIRCUIT



2.7 Configuration of Analog and Digital Pins During ICSP Operations

If an ICSP compliant emulator is selected as a debugger, it automatically initializes all of the A/D input pins (ANx) as “digital” pins. Depending on the particular device, this is done by setting all bits in the ADnPCFG register(s), or clearing all bit in the ANSx registers.

All PIC24F devices will have either one or more ADnPCFG registers or several ANSx registers (one for each port); no device will have both. Refer to **Section 11.2 “Configuring Analog Port Pins (ANSx)”** for more specific information.

The bits in these registers that correspond to the A/D pins that initialized the emulator must not be changed by the user application firmware; otherwise, communication errors will result between the debugger and the device.

If your application needs to use certain A/D pins as analog input pins during the debug session, the user application must modify the appropriate bits during initialization of the A/D module, as follows:

- For devices with an ADnPCFG register, clear the bits corresponding to the pin(s) to be configured as analog. Do not change any other bits, particularly those corresponding to the PGECx/PGEDx pair, at any time.
- For devices with ANSx registers, set the bits corresponding to the pin(s) to be configured as analog. Do not change any other bits, particularly those corresponding to the PGECx/PGEDx pair, at any time.

When a Microchip debugger/emulator is used as a programmer, the user application firmware must correctly configure the ADnPCFG or ANSx registers. Automatic initialization of this register is only done during debugger operation. Failure to correctly configure the register(s) will result in all A/D pins being recognized as analog input pins, resulting in the port value being read as a logic ‘0’, which may affect user application functionality.

2.8 Sigma-Delta A/D Connections

The Sigma-Delta A/D Converter has input and power connections that are independent from the rest of the microcontroller. These connections are required to use the converter, and are in addition to the connection and layout connections provided in **Section 2.1 “Basic Connection Requirements”** and **Section 2.2 “Power Supply Pins”**.

2.8.1 VOLTAGE AND GROUND CONNECTIONS

To minimize noise interference, the Sigma-Delta A/D Converter has independent voltage pins. Converter circuits are supplied through the SVDD pin. Independent ground return is provided through the SVss pin.

As with the microcontroller’s VDD/VSS and AVDD/AVSS pins, bypass capacitors are required on SVDD and SVSS. Requirements for these capacitors are identical to those for the VDD/VSS and AVDD/AVSS pins.

It is recommended that designs using the Sigma-Delta A/D Converter incorporate a separate ground return path for analog circuits. The analog and digital grounds may be tied to a single point at the power source. Analog pins that require grounding should be tied to this analog return. SVSS can be tied to the digital ground, along with VSS and AVSS.

2.8.2 ANALOG INPUTS

The analog signals to be converted are connected to the pins of CH0 and/or CH1. Each channel has inverting and non-inverting inputs (CHn- and CHn+, respectively), and is fully differential.

If not used for conversion, CH1+ and CH1- can be used to supply an external voltage reference to the converter. If an external reference is not used and CH1 is not needed as a conversion input, both pins should be connected to the analog ground return.

2.9 Unused I/Os

Unused I/O pins should be configured as outputs and driven to a logic low state. Alternatively, connect a 1 kΩ to 10 kΩ resistor to VSS on unused pins and drive the output to logic low.

3.0 CPU

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"*, **"CPU with Extended Data Space (EDS)"** (DS39732). The information in this data sheet supersedes the information in the FRM.

The PIC24F CPU has a 16-bit (data) modified, Harvard architecture with an enhanced instruction set and a 24-bit instruction word with a variable length opcode field. The Program Counter (PC) is 23 bits wide and addresses up to 4M instructions of user program memory space. A single-cycle instruction prefetch mechanism is used to help maintain throughput and provides predictable execution. All instructions execute in a single cycle, with the exception of instructions that change the program flow, the double-word move (`MOV.D`) instruction and the table instructions. Overhead-free program loop constructs are supported using the `REPEAT` instructions, which are interruptible at any point.

PIC24F devices have sixteen, 16-bit working registers in the programmer's model. Each of the working registers can act as a data, address or address offset register. The 16th working register (W15) operates as a Software Stack Pointer for interrupts and calls.

The lower 32 Kbytes of the Data Space can be accessed linearly. The upper 32 Kbytes of the Data Space are referred to as Extended Data Space to which the extended data RAM, EPMP memory space or program memory can be mapped.

The Instruction Set Architecture (ISA) has been significantly enhanced beyond that of the PIC18, but maintains an acceptable level of backward compatibility. All PIC18 instructions and addressing modes are supported, either directly, or through simple macros. Many of the ISA enhancements have been driven by compiler efficiency needs.

The core supports Inherent (no operand), Relative, Literal and Memory Direct Addressing modes, along with three groups of addressing modes. All modes support Register Direct and various Register Indirect modes. Each group offers up to seven addressing modes. Instructions are associated with predefined addressing modes, depending upon their functional requirements.

For most instructions, the core is capable of executing a data (or program data) memory read, a working register (data) read, a data memory write and a program (instruction) memory read per instruction cycle. As a result, three parameter instructions can be supported, allowing trinary operations (that is, A + B = C) to be executed in a single cycle.

A high-speed, 17-bit x 17-bit multiplier has been included to significantly enhance the core arithmetic capability and throughput. The multiplier supports Signed, Unsigned and Mixed mode, 16-bit x 16-bit or 8-bit x 8-bit, integer multiplication. All multiply instructions execute in a single cycle.

The 16-bit ALU has been enhanced with integer divide assist hardware that supports an iterative non-restoring divide algorithm. It operates in conjunction with the `REPEAT` instruction looping mechanism and a selection of iterative divide instructions to support 32-bit (or 16-bit), divided by 16-bit, integer signed and unsigned division. All divide operations require 19 cycles to complete but are interruptible at any cycle boundary.

The PIC24F has a vectored exception scheme with up to 8 sources of non-maskable traps and up to 118 interrupt sources. Each interrupt source can be assigned to one of seven priority levels.

A block diagram of the CPU is shown in Figure 3-1.

3.1 Programmer's Model

The programmer's model for the PIC24F is shown in Figure 3-2. All registers in the programmer's model are memory mapped and can be manipulated directly by instructions. A description of each register is provided in Table 3-1. All registers associated with the programmer's model are memory mapped.

FIGURE 3-1: PIC24F CPU CORE BLOCK DIAGRAM

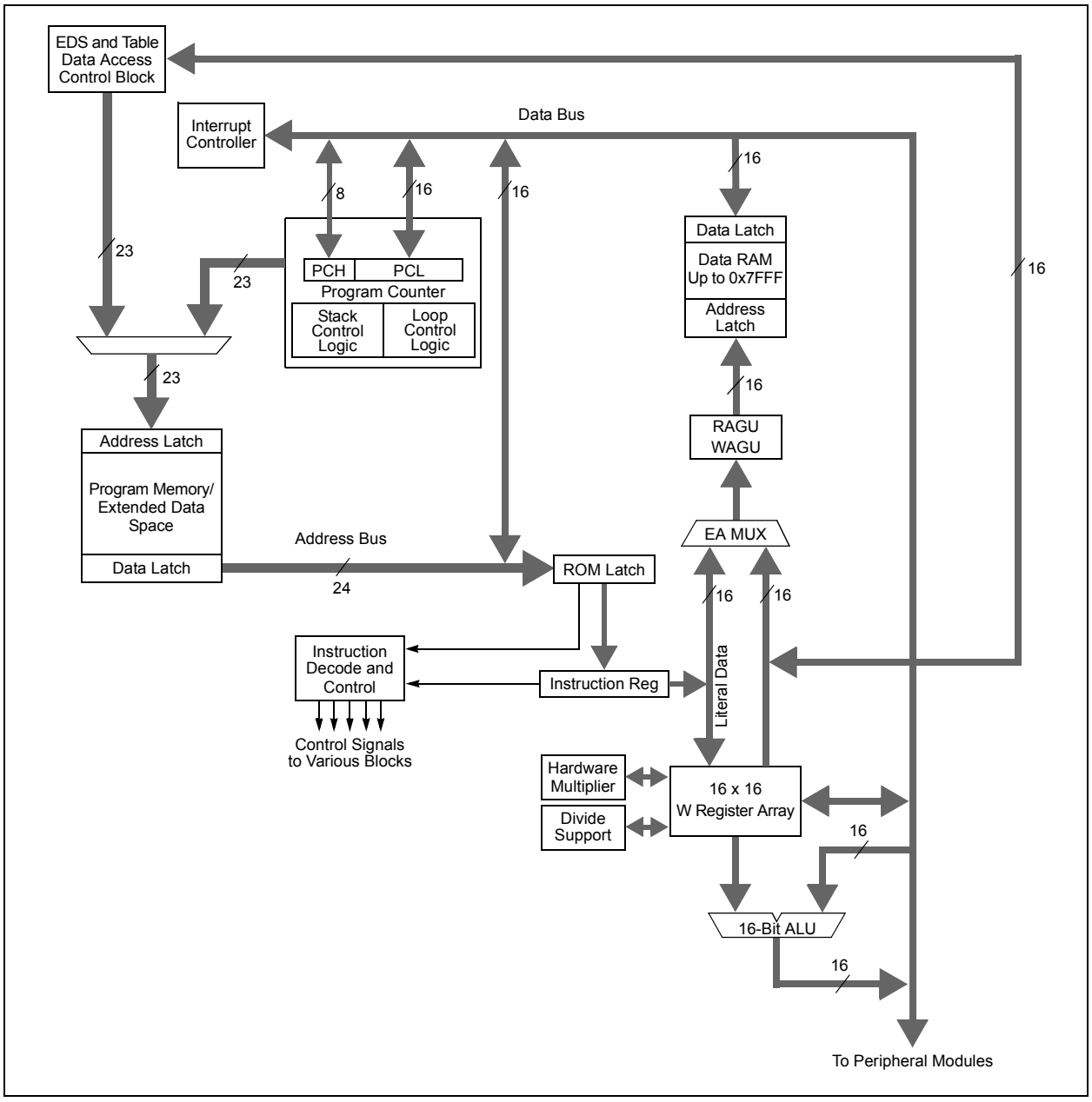
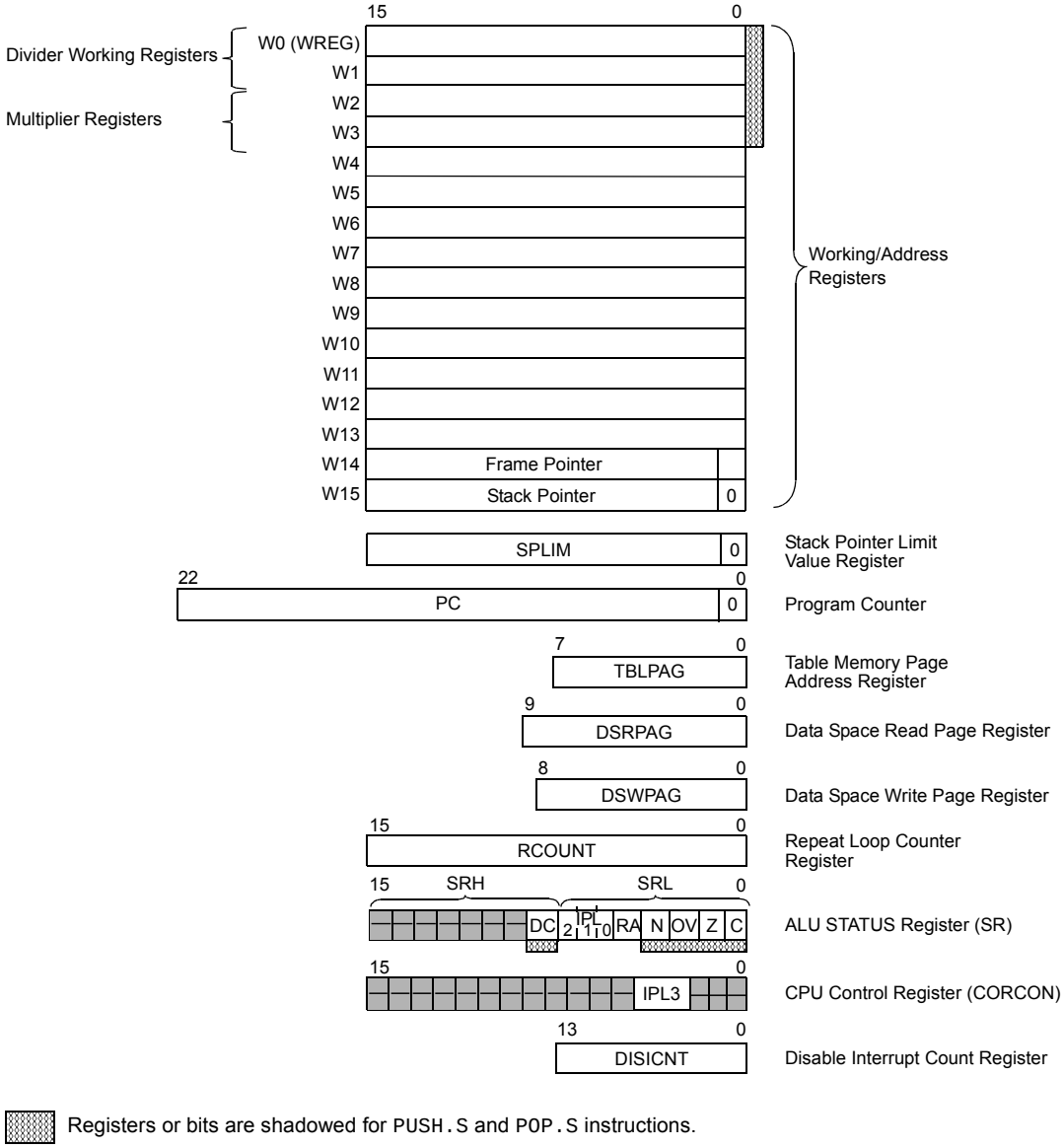


TABLE 3-1: CPU CORE REGISTERS

Register(s) Name	Description
W0 through W15	Working Register Array
PC	23-Bit Program Counter
SR	ALU STATUS Register
SPLIM	Stack Pointer Limit Value Register
TBLPAG	Table Memory Page Address Register
RCOUNT	Repeat Loop Counter Register
CORCON	CPU Control Register
DISICNT	Disable Interrupt Count Register
DSRPAG	Data Space Read Page Register
DSWPAG	Data Space Write Page Register

FIGURE 3-2: PROGRAMMER'S MODEL



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	C
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 15-9 **Unimplemented:** Read as '0'

bit 8 **DC:** ALU Half Carry/Borrow bit
1 = A carry out from the 4th low-order bit (for byte-sized data) or 8th low-order bit (for word-sized data) of the result occurred
0 = No carry out from the 4th or 8th low-order bit of the result has occurred

bit 7-5 **IPL<2:0>:** CPU Interrupt Priority Level (IPL) Status bits(1,2)
111 = CPU Interrupt Priority Level is 7 (15); user interrupts are disabled
110 = CPU Interrupt Priority Level is 6 (14)
101 = CPU Interrupt Priority Level is 5 (13)
100 = CPU Interrupt Priority Level is 4 (12)
011 = CPU Interrupt Priority Level is 3 (11)
010 = CPU Interrupt Priority Level is 2 (10)
001 = CPU Interrupt Priority Level is 1 (9)
000 = CPU Interrupt Priority Level is 0 (8)

bit 4 **RA:** REPEAT Loop Active bit
1 = REPEAT loop is in progress
0 = REPEAT loop is not in progress

bit 3 **N:** ALU Negative bit
1 = Result was negative
0 = Result was not negative (zero or positive)

bit 2 **OV:** ALU Overflow bit
1 = Overflow occurred for signed (2's complement) arithmetic in this arithmetic operation
0 = No overflow has occurred

bit 1 **Z:** ALU Zero bit
1 = An operation, which affects the Z bit, has set it at some time in the past
0 = The most recent operation, which affects the Z bit, has cleared it (i.e., a non-zero result)

bit 0 **C:** ALU Carry/$\overline{\text{Borrow}}$ bit
1 = A carry out from the Most Significant bit (MSb) of the result occurred
0 = No carry out from the Most Significant bit of the result occurred

Note 1: The IPLx Status bits are read-only when NSTDIS (INTCON1<15>) = 1.

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

REGISTER 3-2: CORCON: CPU CORE CONTROL REGISTER

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

U-0	U-0	U-0	U-0	R/C-0	r-1	U-0	U-0
—	—	—	—	IPL3(1)	r	—	—
bit 7							bit 0

Legend:	C = Clearable bit	r = Reserved 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 15-4 **Unimplemented:** Read as '0'

bit 3 **IPL3:** CPU Interrupt Priority Level Status bit(1)
1 = CPU Interrupt Priority Level is greater than 7
0 = CPU Interrupt Priority Level is 7 or less

bit 2 **Reserved:** Read as '1'

bit 1-0 **Unimplemented:** Read as '0'

Note 1: The IPL3 bit is concatenated with the IPL<2:0> bits (SR<7:5>) to form the CPU Interrupt priority Level; see Register 3-1 for bit description.

3.3 Arithmetic Logic Unit (ALU)

The PIC24F ALU is 16 bits wide and is capable of addition, subtraction, bit shifts and logic operations. Unless otherwise mentioned, arithmetic operations are 2's complement in nature. Depending on the operation, the ALU may affect the values of the Carry (C), Zero (Z), Negative (N), Overflow (OV) and Digit Carry (DC) Status bits in the SR register. The C and DC Status bits operate as Borrow and Digit Borrow bits, respectively, for subtraction operations.

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

The PIC24F CPU incorporates hardware support for both multiplication and division. This includes a dedicated hardware multiplier and support hardware for 16-bit divisor division.

3.3.1 MULTIPLIER

The ALU contains a high-speed, 17-bit x 17-bit multiplier. It supports unsigned, signed or mixed sign operation in several multiplication modes:

1. 16-bit x 16-bit signed
2. 16-bit x 16-bit unsigned
3. 16-bit signed x 5-bit (literal) unsigned
4. 16-bit unsigned x 16-bit unsigned
5. 16-bit unsigned x 5-bit (literal) unsigned
6. 16-bit unsigned x 16-bit signed
7. 8-bit unsigned x 8-bit unsigned

3.3.2 DIVIDER

The divide block supports signed and unsigned integer divide operations with the following data sizes:

1. 32-bit signed/16-bit signed divide
2. 32-bit unsigned/16-bit unsigned divide
3. 16-bit signed/16-bit signed divide
4. 16-bit unsigned/16-bit unsigned divide

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

3.3.3 MULTIBIT SHIFT SUPPORT

The PIC24F ALU supports both single bit and single-cycle, multibit arithmetic and logic shifts. Multibit shifts are implemented using a shifter block, capable of performing up to a 15-bit arithmetic right shift, or up to a 15-bit left shift, in a single cycle. All multibit shift instructions only support Register Direct Addressing for both the operand source and result destination.

A full summary of instructions that use the shift operation is provided in Table 3-2.

TABLE 3-2: INSTRUCTIONS THAT USE THE SINGLE BIT AND MULTIBIT SHIFT OPERATION

Instruction	Description
ASR	Arithmetic shift right source register by one or more bits.
SL	Shift left source register by one or more bits.
LSR	Logical shift right source register by one or more bits.

4.0 MEMORY ORGANIZATION

As Harvard architecture devices, PIC24F microcontrollers feature separate program and data memory spaces and buses. This architecture also allows direct access of program memory from the Data Space (DS) during code execution.

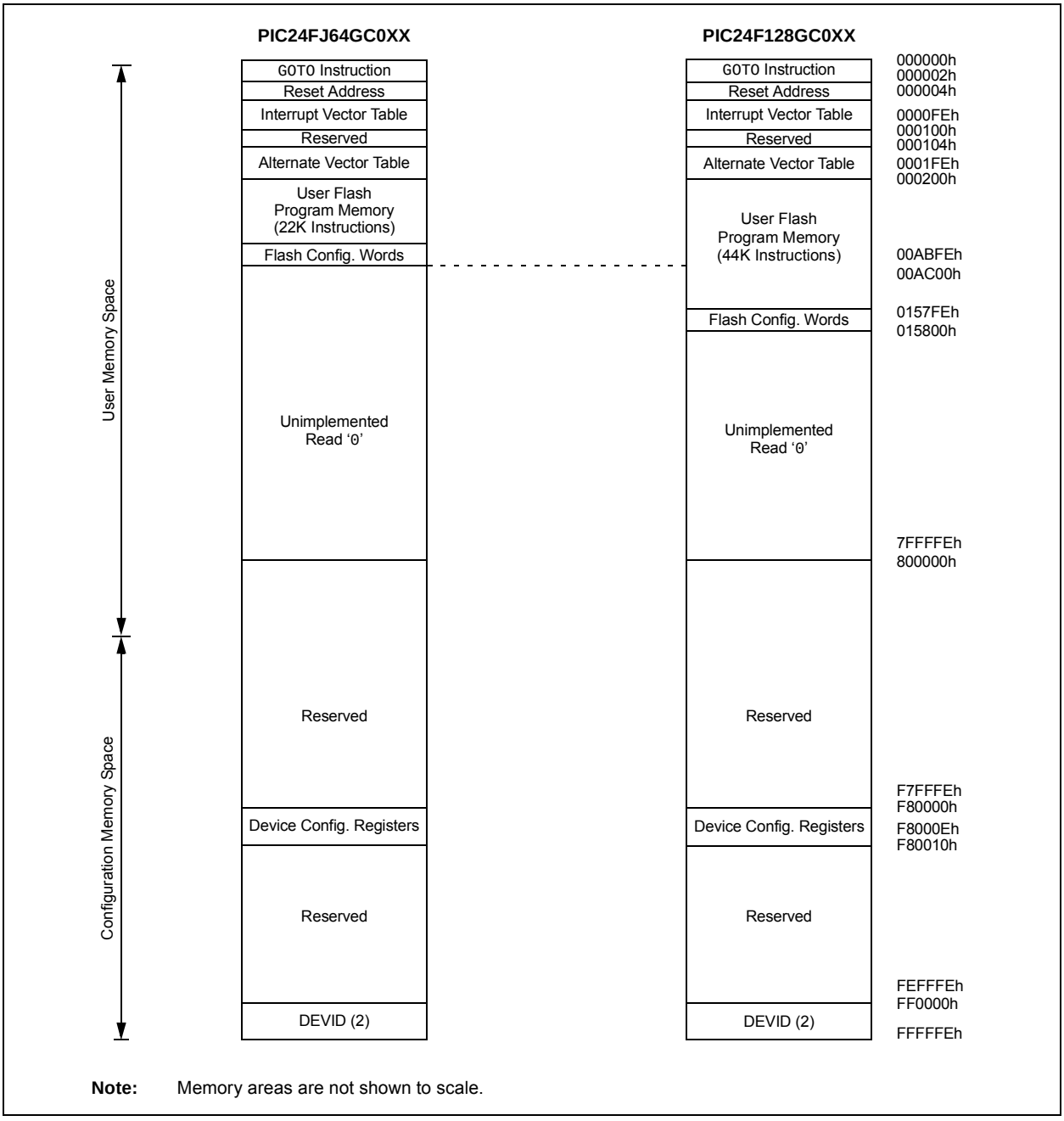
4.1 Program Memory Space

The program address memory space of the PIC24FJ128GC010 family devices is 4M instructions. The space is addressable by a 24-bit value derived from either the 23-bit Program Counter (PC) during program execution, or from table operation or Data Space remapping, as described in **Section 4.3 "Interfacing Program and Data Memory Spaces"**.

User access to the program memory space is restricted to the lower half of the address range (000000h to 7FFFFFh). The exception is the use of TBLRD/TBLWT operations, which use TBLPAG<7> to permit access to the Configuration bits and Device ID sections of the configuration memory space.

Memory maps for the PIC24FJ128GC010 family of devices are shown in Figure 4-1.

FIGURE 4-1: PROGRAM SPACE MEMORY MAP FOR PIC24FJ128GC010 FAMILY DEVICES



Note: Memory areas are not shown to scale.

4.1.1 PROGRAM MEMORY ORGANIZATION

The program memory space is organized in word-addressable blocks. Although it is treated as 24 bits wide, it is more appropriate to think of each address of the program memory as a lower and upper word, with the upper byte of the upper word being unimplemented. The lower word always has an even address, while the upper word has an odd address (Figure 4-2).

Program memory addresses are always word-aligned on the lower word and addresses are incremented or decremented by two during code execution. This arrangement also provides compatibility with data memory space addressing and makes it possible to access data in the program memory space.

4.1.2 HARD MEMORY VECTORS

All PIC24F devices reserve the addresses between 000000h and 000200h for hard-coded program execution vectors. A hardware Reset vector is provided to redirect code execution from the default value of the PC on device Reset to the actual start of code. A GOTO instruction is programmed by the user at 000000h with the actual address for the start of code at 000002h.

PIC24F devices also have two Interrupt Vector Tables (IVTs), located from 000004h to 0000FFh and 000100h to 0001FFh. These vector tables allow each of the many device interrupt sources to be handled by separate ISRs. A more detailed discussion of the Interrupt Vector Tables is provided in **Section 8.1 "Interrupt Vector Table"**.

4.1.3 FLASH CONFIGURATION WORDS

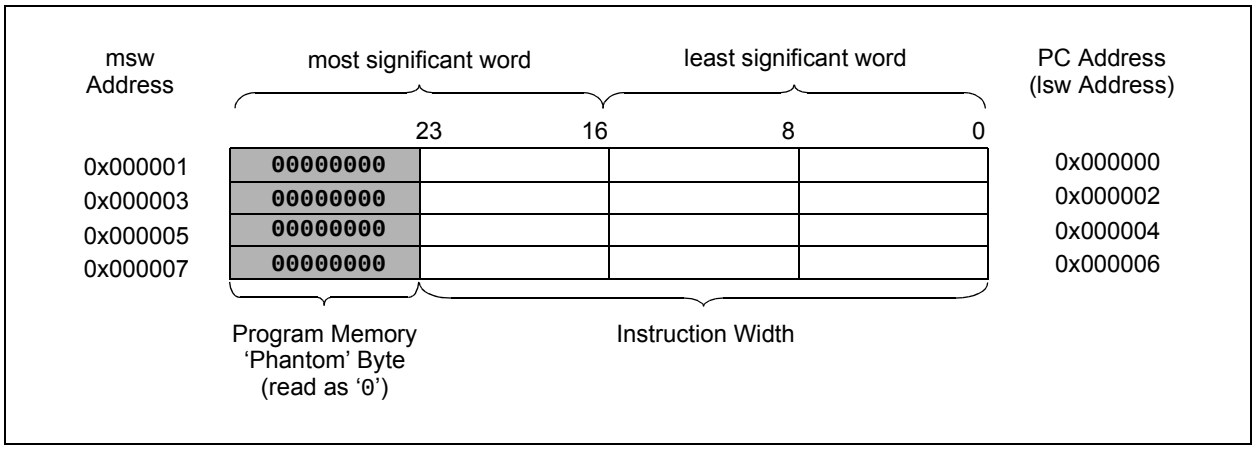
In PIC24FJ128GC010 family devices, the top four words of on-chip program memory are reserved for configuration information. On device Reset, the configuration information is copied into the appropriate Configuration register. The addresses of the Flash Configuration Word for devices in the PIC24FJ128GC010 family are shown in Table 4-1. Their location in the memory map is shown with the other memory vectors in Figure 4-1.

The Configuration Words in program memory are a compact format. The actual Configuration bits are mapped in several different registers in the configuration memory space. Their order in the Flash Configuration Words does not reflect a corresponding arrangement in the configuration space. Additional details on the device Configuration Words are provided in **Section 34.0 "Special Features"**.

TABLE 4-1: FLASH CONFIGURATION WORDS FOR PIC24FJ128GC010 FAMILY DEVICES

Device	Program Memory (Words)	Configuration Word Addresses
PIC24FJ64GC0XX	22,016	00ABF8h:00ABFEh
PIC24FJ128GC0XX	44,032	0157F8h:0157FEh

FIGURE 4-2: PROGRAM MEMORY ORGANIZATION



4.2 Data Memory Space

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 45. "Data Memory with Extended Data Space (EDS)**" (DS39733). The information in this data sheet supersedes the information in the FRM.

The PIC24F core has a 16-bit wide data memory space, addressable as a single linear range. The Data Space (DS) is accessed using two Address Generation Units (AGUs), one each for read and write operations. The Data Space memory map is shown in Figure 4-3.

The 16-bit wide data addresses in the data memory space point to bytes within the Data Space. This gives a DS address range of 64 Kbytes or 32K words. The lower half (0000h to 7FFFh) is used for implemented (on-chip) memory addresses.

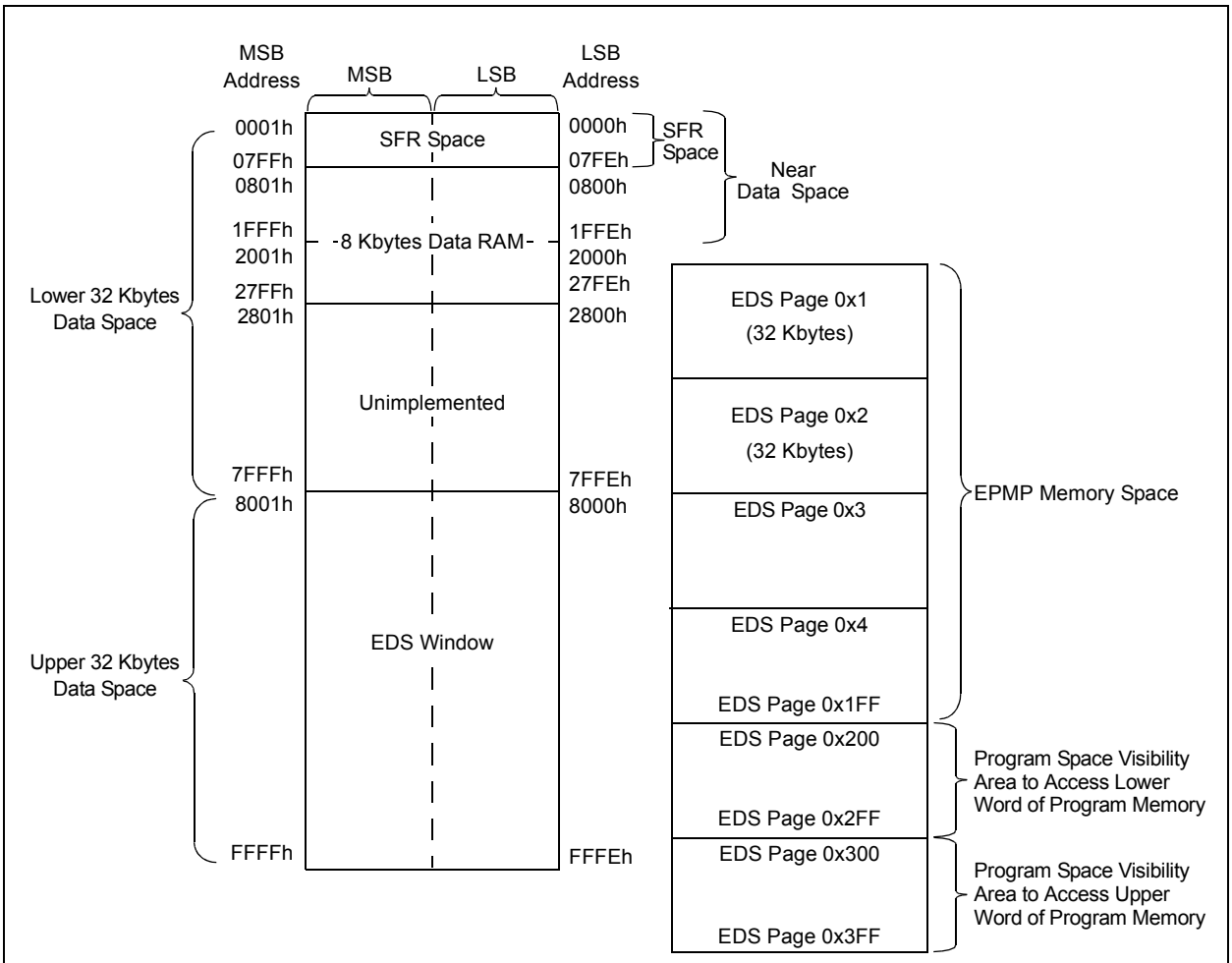
The upper half of data memory address space (8000h to FFFFh) is used as a window into the Extended Data Space (EDS). This allows the microcontroller to directly access a greater range of data beyond the standard 16-bit address range. EDS is discussed in detail in **Section 4.2.5 "Extended Data Space (EDS)"**.

The lower half of DS is compatible with previous PIC24F microcontrollers without EDS. All PIC24FJ128GC010 family devices implement 8 Kbytes of data RAM in the lower half of the DS, from 0800h to 27FFh.

4.2.1 DATA SPACE WIDTH

The data memory space is organized in byte-addressable, 16-bit wide blocks. Data is aligned in data memory and registers as 16-bit words, but all Data Space Effective Addresses (EAs) resolve to bytes. The Least Significant Bytes (LSBs) of each word have even addresses, while the Most Significant Bytes (MSBs) have odd addresses.

FIGURE 4-3: DATA SPACE MEMORY MAP FOR PIC24FJ128GC010 FAMILY DEVICES



Note: Memory areas are not shown to scale.

4.2.2 DATA MEMORY ORGANIZATION AND ALIGNMENT

To maintain backward compatibility with PIC® MCUs and improve Data Space memory usage efficiency, the PIC24F instruction set supports both word and byte operations. As a consequence of byte accessibility, all EA calculations are internally scaled to step through word-aligned memory. For example, the core recognizes that Post-Modified Register Indirect Addressing mode [Ws++] will result in a value of Ws + 1 for byte operations and Ws + 2 for word operations.

Data byte reads will read the complete word, which contains the byte, using the LSB of any EA to determine which byte to select. The selected byte is placed onto the LSB of the data path. That is, data memory and registers are organized as two parallel, byte-wide entities with shared (word) address decode but separate write lines. Data byte writes only write to the corresponding side of the array or register which matches the byte address.

All word accesses must be aligned to an even address. Misaligned word data fetches are not supported, so care must be taken when mixing byte and word operations or translating from 8-bit MCU code. If a misaligned read or write is attempted, an address error trap will be generated. If the error occurred on a read, the instruction underway is completed; if it occurred on a write, the instruction will be executed but the write will not occur. In either case, a trap is then executed, allowing the system and/or user to examine the machine state prior to execution of the address Fault.

All byte loads into any W register are loaded into the LSB. The Most Significant Byte (MSB) is not modified.

A Sign-Extend (SE) instruction is provided to allow users to translate 8-bit signed data to 16-bit signed values. Alternatively, for 16-bit unsigned data, users can clear the MSB of any W register by executing a Zero-Extend (ZE) instruction on the appropriate address.

Although most instructions are capable of operating on word or byte data sizes, it should be noted that some instructions operate only on words.

4.2.3 NEAR DATA SPACE

The 8-Kbyte area between 0000h and 1FFFh is referred to as the Near Data Space. Locations in this space are directly addressable via a 13-bit absolute address field within all memory direct instructions. The remainder of the Data Space is addressable indirectly. Additionally, the whole Data Space is addressable using MOV instructions, which support Memory Direct Addressing with a 16-bit address field.

4.2.4 SPECIAL FUNCTION REGISTER (SFR) SPACE

The first 2 Kbytes of the Near Data Space, from 0000h to 07FFh, are primarily occupied with Special Function Registers (SFRs). These are used by the PIC24F core and peripheral modules for controlling the operation of the device.

SFRs are distributed among the modules that they control and are generally grouped together by module. Much of the SFR space contains unused addresses; these are read as '0'. A diagram of the SFR space, showing where the SFRs are actually implemented, is shown in Table 4-2. Each implemented area indicates a 32-byte region where at least one address is implemented as an SFR. A complete list of implemented SFRs, including their addresses, is shown in Tables 4-3 through 4-39.

TABLE 4-2: IMPLEMENTED REGIONS OF SFR DATA SPACE

	SFR Space Address							
	xx00	**xx20**	**xx40**	**xx60**	**xx80**	**xxA0**	**xxC0**	**xxE0**
000h	Core			ICN	Interrupts			
100h	Timers	—	Capture		Compare			—
200h	I^2C™	URT	SPI/URT(1)	SPI	—	—	—	URT	I/O
300h	A/D			DMA				—
400h	A/D	DAC	—	—	—	USB		S/D	ANA	—
500h	A/D			—	LCD			—
600h	EPMP	RTC/CMP(2)	CRC	—	AVR	PPS				—
700h	—	—	System	NVM/PMD	CTM	—	—	—	—

Legend: — = No implemented SFRs in this block; URT = UART; S/D = Sigma-Delta A/D; AVR = Analog Reference

Note 1: This region includes registers for the op amp module.

2: This region includes registers for the Digital Signal Modulator (DSM) module.

TABLE 4-3: CPU CORE REGISTERS 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
WREG0	0000	Working Register 0																0000
WREG1	0002	Working Register 1																0000
WREG2	0004	Working Register 2																0000
WREG3	0006	Working Register 3																0000
WREG4	0008	Working Register 4																0000
WREG5	000A	Working Register 5																0000
WREG6	000C	Working Register 6																0000
WREG7	000E	Working Register 7																0000
WREG8	0010	Working Register 8																0000
WREG9	0012	Working Register 9																0000
WREG10	0014	Working Register 10																0000
WREG11	0016	Working Register 11																0000
WREG12	0018	Working Register 12																0000
WREG13	001A	Working Register 13																0000
WREG14	001C	Working Register 14																0000
WREG15	001E	Working Register 15																0800
SPLIM	0020	Stack Pointer Limit Value Register																xxxx
PCL	002E	Program Counter Low Word Register																0000
PCH	0030	—	—	—	—	—	—	—	—	Program Counter Register High Byte								0000
DSRPAG	0032	—	—	—	—	—	—	Extended Data Space Read Page Address Register										0000
DSWPAG	0034	—	—	—	—	—	—	—	Extended Data Space Write Page Address Register									0000
RCOUNT	0036	Repeat Loop Counter Register																xxxx
SR	0042	—	—	—	—	—	—	—	DC	IPL2	IPL1	IPL0	RA	N	OV	Z	C	0000
CORCON	0044	—	—	—	—	—	—	—	—	—	—	—	—	IPL3	r	—	—	0004
DISICNT	0052	—	—	Disable Interrupts Counter Register														xxxx
TBLPAG	0054	—	—	—	—	—	—	—	—	Table Memory Page Address Register								0000

Legend: — = unimplemented, read as '0'; r = reserved, do not modify. Reset values are shown in hexadecimal.

TABLE 4-4: ICN 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
CNPD1	0056	CN15PDE	CN14PDE	CN13PDE	CN12PDE	CN11PDE	CN10PDE	CN9PDE	CN8PDE	CN7PDE	CN6PDE	CN5PDE	CN4PDE	CN3PDE	CN2PDE	—	—	0000
CNPD2	0058	CN31PDE	CN30PDE	—	—	—	—	CN25PDE	CN24PDE	CN23PDE	CN22PDE	CN21PDE(1)	CN20PDE(1)	CN19PDE(1)	CN18PDE	CN17PDE	CN16PDE	0000
CNPD3	005A	CN47PDE(1)	CN46PDE(1)	CN45PDE(1)	CN44PDE(1)	CN43PDE(1)	CN42PDE(1)	CN41PDE(1)	CN40PDE(1)	CN39PDE(1)	CN38PDE(1)	CN37PDE(1)	CN36PDE(1)	CN35PDE(1)	CN34PDE(1)	CN33PDE(1)	CN32PDE	0000
CNPD4	005C	CN63PDE	CN62PDE	CN61PDE	CN60PDE	CN59PDE	CN58PDE	CN57PDE(1)	CN56PDE	CN55PDE	CN54PDE	CN53PDE	CN52PDE	CN51PDE	CN50PDE	CN49PDE	CN48PDE(1)	0000
CNPD5	005E	CN79PDE(1)	CN78PDE(1)	CN77PDE(1)	CN76PDE(1)	CN75PDE(1)	CN74PDE(1)	CN73PDE	CN72PDE	CN71PDE	CN70PDE(1)	CN69PDE	CN68PDE	CN67PDE(1)	CN66PDE(1)	CN65PDE	CN64PDE	0000
CNPD6	0060	—	—	—	—	—	—	—	—	—	—	—	—	CN83PDE	CN82PDE(1)	CN81PDE(1)	CN80PDE(1)	0000
CNEN1	0062	CN15IE	CN14IE	CN13IE	CN12IE	CN11IE	CN10IE	CN9IE	CN8IE	CN7IE	CN6IE	CN5IE	CN4IE	CN3IE	CN2IE	—	—	0000
CNEN2	0064	CN31IE	CN30IE	—	—	—	—	CN25IE	CN24IE	CN23IE	CN22IE	CN21IE(1)	CN20IE(1)	CN19IE(1)	CN18IE	CN17IE	CN16IE	0000
CNEN3	0066	CN47IE(1)	CN46IE(1)	CN45IE(1)	CN44IE(1)	CN43IE(1)	CN42IE(1)	CN41IE(1)	CN40IE(1)	CN39IE(1)	CN38IE(1)	CN37IE(1)	CN36IE(1)	CN35IE(1)	CN34IE(1)	CN33IE(1)	CN32IE	0000
CNEN4	0068	CN63IE	CN62IE	CN61IE	CN60IE	CN59IE	CN58IE	CN57IE(1)	CN56IE	CN55IE	CN54IE	CN53IE	CN52IE	CN51IE	CN50IE	CN49IE	CN48IE(1)	0000
CNEN5	006A	CN79IE(1)	CN78IE(1)	CN77IE(1)	CN76IE(1)	CN75IE(1)	CN74IE(1)	CN73IE	CN72IE	CN71IE	CN70IE(1)	CN69IE	CN68IE	CN67IE(1)	CN66IE(1)	CN65IE	CN64IE	0000
CNEN6	006C	—	—	—	—	—	—	—	—	—	—	—	—	CN83IE	CN82IE(1)	CN81IE(1)	CN80IE(1)	0000
CNPU1	006E	CN15PUE	CN14PUE	CN13PUE	CN12PUE	CN11PUE	CN10PUE	CN9PUE	CN8PUE	CN7PUE	CN6PUE	CN5PUE	CN4PUE	CN3PUE	CN2PUE	—	—	0000
CNPU2	0070	CN31PUE	CN30PUE	—	—	—	—	CN25PUE	CN24PUE	CN23PUE	CN22PUE	CN21PUE(1)	CN20PUE(1)	CN19PUE(1)	CN18PUE	CN17PUE	CN16PUE	0000
CNPU3	0072	CN47PUE(1)	CN46PUE(1)	CN45PUE(1)	CN44PUE(1)	CN43PUE(1)	CN42PUE(1)	CN41PUE(1)	CN40PUE(1)	CN39PUE(1)	CN38PUE(1)	CN37PUE(1)	CN36PUE(1)	CN35PUE(1)	CN34PUE(1)	CN33PUE(1)	CN32PUE	0000
CNPU4	0074	CN63PUE	CN62PUE	CN61PUE	CN60PUE	CN59PUE	CN58PUE	CN57PUE(1)	CN56PUE	CN55PUE	CN54PUE	CN53PUE	CN52PUE	CN51PUE	CN50PUE	CN49PUE	CN48PUE(1)	0000
CNPU5	0076	CN79PUE(1)	CN78PUE(1)	CN77PUE(1)	CN76PUE(1)	CN75PUE(1)	CN74PUE(1)	CN73PUE	CN72PUE	CN71PUE	CN70PUE(1)	CN69PUE	CN68PUE	CN67PUE(1)	CN66PUE(1)	CN65PUE	CN64PUE	0000
CNPU6	0078	—	—	—	—	—	—	—	—	—	—	—	—	CN83PUE	CN82PUE(1)	CN81PUE(1)	CN80PUE(1)	0000

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

Note 1: These bits are unimplemented in 64-pin devices, read as '0'.

TABLE 4-5: INTERRUPT CONTROLLER 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
INTCON1	0080	NSTDIS	—	—	—	—	—	—	—	—	—	—	MATHERR	ADDRERR	STKERR	OSCFAIL	—	0000
INTCON2	0082	ALTIVT	DISI	—	—	—	—	—	—	—	—	—	INT4EP	INT3EP	INT2EP	INT1EP	INT0EP	0000
IFS0	0084	—	DMA1IF	AD1IF	U1TXIF	U1RXIF	SPI1IF	SPF1IF	T3IF	T2IF	OC2IF	IC2IF	DMA0IF	T1IF	OC1IF	IC1IF	INT0IF	0000
IFS1	0086	U2TXIF	U2RXIF	INT2IF	T5IF	T4IF	OC4IF	OC3IF	DMA2IF	IC8IF	IC7IF	—	INT1IF	CNIF	CMIF	MI2C1IF	SI2C1IF	0000
IFS2	0088	—	DMA4IF	PMPIF	OC8IF	OC7IF	OC6IF	OC5IF	IC6IF	IC5IF	IC4IF	IC3IF	DMA3IF	—	—	SPI2IF	SPF2IF	0000
IFS3	008A	—	RTCIF	DMA5IF	—	—	—	—	—	—	INT4IF	INT3IF	—	—	MI2C2IF	SI2C2IF	—	0000
IFS4	008C	DAC2IF	DAC1IF	CTMUIF	—	—	—	—	HLVDIF	—	—	—	—	CRCIF	U2ERIF	U1ERIF	—	0000
IFS5	008E	—	—	IC9IF	OC9IF	—	—	U4TXIF	U4RXIF	U4ERIF	USB1IF	—	—	U3TXIF	U3RXIF	U3ERIF	—	0000
IFS6	0090	—	—	—	—	—	FSTIF	SDA1IF	AMP2IF	AMP1IF	—	—	LCDIF	—	—	—	—	0000
IFS7	0092	—	—	—	—	—	—	—	—	—	—	JTAGIF	—	—	—	—	—	0000
IEC0	0094	—	DMA1IE	AD1IE	U1TXIE	U1RXIE	SPI1IE	SPF1IE	T3IE	T2IE	OC2IE	IC2IE	DMA0IE	T1IE	OC1IE	IC1IE	INT0IE	0000
IEC1	0096	U2TXIE	U2RXIE	INT2IE	T5IE	T4IE	OC4IE	OC3IE	DMA2IE	IC8IE	IC7IE	—	INT1IE	CNIE	CMIE	MI2C1IE	SI2C1IE	0000
IEC2	0098	—	DMA4IE	PMPIE	OC8IE	OC7IE	OC6IE	OC5IE	IC6IE	IC5IE	IC4IE	IC3IE	DMA3IE	—	—	SPI2IE	SPF2IE	0000
IEC3	009A	—	RTCIE	DMA5IE	—	—	—	—	—	—	INT4IE	INT3IE	—	—	MI2C2IE	SI2C2IE	—	0000
IEC4	009C	DAC2IE	DAC1IE	CTMUIE	—	—	—	—	HLVDIE	—	—	—	—	CRCIE	U2ERIE	U1ERIE	—	0000
IEC5	009E	—	—	IC9IE	OC9IE	—	—	U4TXIE	U4RXIE	U4ERIE	USB1IE	—	—	U3TXIE	U3RXIE	U3ERIE	—	0000
IEC6	00A0	—	—	—	—	—	FSTIE	SDA1IE	AMP2IE	AMP1IE	—	—	LCDIE	—	—	—	—	0000
IEC7	00A2	—	—	—	—	—	—	—	—	—	—	JTAGIE	—	—	—	—	—	0000
IPC0	00A4	—	T1IP2	T1IP1	T1IP0	—	OC1IP2	OC1IP1	OC1IP0	—	IC1IP2	IC1IP1	IC1IP0	—	INT0IP2	INT0IP1	INT0IP0	4444
IPC1	00A6	—	T2IP2	T2IP1	T2IP0	—	OC2IP2	OC2IP1	OC2IP0	—	IC2IP2	IC2IP1	IC2IP0	—	DMA0IP2	DMA0IP1	DMA0IP0	4444
IPC2	00A8	—	U1RXIP2	U1RXIP1	U1RXIP0	—	SPI1IP2	SPI1IP1	SPI1IP0	—	SPF1IP2	SPF1IP1	SPF1IP0	—	T3IP2	T3IP1	T3IP0	4444
IPC3	00AA	—	—	—	—	—	DMA1IP2	DMA1IP1	DMA1IP0	—	AD1IP2	AD1IP1	AD1IP0	—	U1TXIP2	U1TXIP1	U1TXIP0	0444
IPC4	00AC	—	CNIP2	CNIP1	CNIP0	—	CMIP2	CMIP1	CMIP0	—	MI2C1IP2	MI2C1IP1	MI2C1IP0	—	SI2C1IP2	SI2C1IP1	SI2C1IP0	4444
IPC5	00AE	—	IC8IP2	IC8IP1	IC8IP0	—	IC7IP2	IC7IP1	IC7IP0	—	—	—	—	—	INT1IP2	INT1IP1	INT1IP0	4404
IPC6	00B0	—	T4IP2	T4IP1	T4IP0	—	OC4IP2	OC4IP1	OC4IP0	—	OC3IP2	OC3IP1	OC3IP0	—	DMA2IP2	DMA2IP1	DMA2IP0	4444
IPC7	00B2	—	U2TXIP2	U2TXIP1	U2TXIP0	—	U2RXIP2	U2RXIP1	U2RXIP0	—	INT2IP2	INT2IP1	INT2IP0	—	T5IP2	T5IP1	T5IP0	4444
IPC8	00B4	—	—	—	—	—	—	—	—	—	SPI2IP2	SPI2IP1	SPI2IP0	—	SPF2IP2	SPF2IP1	SPF2IP0	0044
IPC9	00B6	—	IC5IP2	IC5IP1	IC5IP0	—	IC4IP2	IC4IP1	IC4IP0	—	IC3IP2	IC3IP1	IC3IP0	—	DMA3IP2	DMA3IP1	DMA3IP0	4444
IPC10	00B8	—	OC7IP2	OC7IP1	OC7IP0	—	OC6IP2	OC6IP1	OC6IP0	—	OC5IP2	OC5IP1	OC5IP0	—	IC6IP2	IC6IP1	IC6IP0	4444
IPC11	00BA	—	—	—	—	—	DMA4IP2	DMA4IP1	DMA4IP0	—	PMPIP2	PMPIP1	PMPIP0	—	OC8IP2	OC8IP1	OC8IP0	0444
IPC12	00BC	—	—	—	—	—	MI2C2IP2	MI2C2IP1	MI2C2IP0	—	SI2C2IP2	SI2C2IP1	SI2C2IP0	—	—	—	—	0440
IPC13	00BE	—	—	—	—	—	INT4IP2	INT4IP1	INT4IP0	—	INT3IP2	INT3IP1	INT3IP0	—	—	—	—	0440
IPC15	00C2	—	—	—	—	—	RTCIP2	RTCIP1	RTCIP0	—	DMA5IP2	DMA5IP1	DMA5IP0	—	—	—	—	0440

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

TABLE 4-5: INTERRUPT CONTROLLER REGISTER MAP (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
IPC16	00C4	—	CRCIP2	CRCIP1	CRCIP0	—	U2ERIP2	U2ERIP1	U2ERIP0	—	U1ERIP2	U1ERIP1	U1ERIP0	—	—	—	—	4440
IPC18	00C8	—	—	—	—	—	—	—	—	—	—	—	—	—	HLVDIP2	HLVDIP1	HLVDIP0	0004
IPC19	00CA	—	DAC2IP2	DAC2IP1	DAC2IP0	—	DAC1IP2	DAC1IP1	DAC1IP0	—	CTMUIP2	CTMUIP1	CTMUIP0	—	—	—	—	4440
IPC20	00CC	—	U3TXIP2	U3TXIP1	U3TXIP0	—	U3RXIP2	U3RXIP1	U3RXIP0	—	U3ERIP2	U3ERIP1	U3ERIP0	—	—	—	—	4440
IPC21	00CE	—	U4ERIP2	U4ERIP1	U4ERIP0	—	USB1IP2	USB1IP1	USB1IP0	—	—	—	—	—	—	—	—	4400
IPC22	00D0	—	—	—	—	—	—	—	—	—	U4TXIP2	U4TXIP1	U4TXIP0	—	U4RXIP2	U4RXIP1	U4RXIP0	0044
IPC23	00D2	—	—	—	—	—	—	—	—	—	IC9IP2	IC9IP1	IC9IP0	—	OC9IP2	OC9IP1	OC9IP0	0044
IPC25	00D6	—	AMP1IP2	AMP1IP1	AMP1IP0	—	—	—	—	—	—	—	—	—	LCDIP2	LCDIP1	LCDIP0	4004
IPC26	00D8	—	—	—	—	—	FSTIP2	FSTIP1	FSTIP0	—	SDA1IP2	SDA1IP1	SDA1IP0	—	AMP2IP2	AMP2IP1	AMP2IP0	0444
IPC29	00DE	—	—	—	—	—	—	—	—	—	JTAGIP2	JTAGIP1	JTAGIP0	—	—	—	—	0040
INTTREG	00E0	CPUIRQ	r	VHOLD	—	ILR3	ILR2	ILR1	ILR0	—	VECNUM6	VECNUM5	VECNUM4	VECNUM3	VECNUM2	VECNUM1	VECNUM0	0000

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

TABLE 4-6: TIMER 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
TMR1	0100	Timer1 Register																0000
PR1	0102	Timer1 Period Register																FFFF
T1CON	0104	TON	—	TSIDL	—	—	—	TIECS1	TIECS0	—	TGATE	TCKPS1	TCKPS0	—	TSYNC	TCS	—	0000
TMR2	0106	Timer2 Register																0000
TMR3HLD	0108	Timer3 Holding Register (for 32-bit timer operations only)																0000
TMR3	010A	Timer3 Register																0000
PR2	010C	Timer2 Period Register																FFFF
PR3	010E	Timer3 Period Register																FFFF
T2CON	0110	TON	—	TSIDL	—	—	—	TIECS1	TIECS0	—	TGATE	TCKPS1	TCKPS0	T32	—	TCS	—	0000
T3CON	0112	TON	—	TSIDL	—	—	—	TIECS1	TIECS0	—	TGATE	TCKPS1	TCKPS0	—	—	TCS	—	0000
TMR4	0114	Timer4 Register																0000
TMR5HLD	0116	Timer5 Holding Register (for 32-bit operations only)																0000
TMR5	0118	Timer5 Register																0000
PR4	011A	Timer4 Period Register																FFFF
PR5	011C	Timer5 Period Register																FFFF
T4CON	011E	TON	—	TSIDL	—	—	—	TIECS1	TIECS0	—	TGATE	TCKPS1	TCKPS0	T45	—	TCS	—	0000
T5CON	0120	TON	—	TSIDL	—	—	—	TIECS1	TIECS0	—	TGATE	TCKPS1	TCKPS0	—	—	TCS	—	0000

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

TABLE 4-7: INPUT CAPTURE 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
IC1CON1	0140	—	—	ICSIDL	ICTSEL2	ICTSEL1	ICTSEL0	—	—	—	ICI1	ICI0	ICOV	ICBNE	ICM2	ICM1	ICM0	0000
IC1CON2	0142	—	—	—	—	—	—	—	IC32	ICTRIG	TRIGSTAT	—	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000D
IC1BUF	0144	Input Capture 1 Buffer Register																0000
IC1TMR	0146	Timer Value 1 Register																xxxx
IC2CON1	0148	—	—	ICSIDL	ICTSEL2	ICTSEL1	ICTSEL0	—	—	—	ICI1	ICI0	ICOV	ICBNE	ICM2	ICM1	ICM0	0000
IC2CON2	014A	—	—	—	—	—	—	—	IC32	ICTRIG	TRIGSTAT	—	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000D
IC2BUF	014C	Input Capture 2 Buffer Register																0000
IC2TMR	014E	Timer Value 2 Register																xxxx
IC3CON1	0150	—	—	ICSIDL	ICTSEL2	ICTSEL1	ICTSEL0	—	—	—	ICI1	ICI0	ICOV	ICBNE	ICM2	ICM1	ICM0	0000
IC3CON2	0152	—	—	—	—	—	—	—	IC32	ICTRIG	TRIGSTAT	—	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000D
IC3BUF	0154	Input Capture 3 Buffer Register																0000
IC3TMR	0156	Timer Value 3 Register																xxxx
IC4CON1	0158	—	—	ICSIDL	ICTSEL2	ICTSEL1	ICTSEL0	—	—	—	ICI1	ICI0	ICOV	ICBNE	ICM2	ICM1	ICM0	0000
IC4CON2	015A	—	—	—	—	—	—	—	IC32	ICTRIG	TRIGSTAT	—	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000D
IC4BUF	015C	Input Capture 4 Buffer Register																0000
IC4TMR	015E	Timer Value 4 Register																xxxx
IC5CON1	0160	—	—	ICSIDL	ICTSEL2	ICTSEL1	ICTSEL0	—	—	—	ICI1	ICI0	ICOV	ICBNE	ICM2	ICM1	ICM0	0000
IC5CON2	0162	—	—	—	—	—	—	—	IC32	ICTRIG	TRIGSTAT	—	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000D
IC5BUF	0164	Input Capture 5 Buffer Register																0000
IC5TMR	0166	Timer Value 5 Register																xxxx
IC6CON1	0168	—	—	ICSIDL	ICTSEL2	ICTSEL1	ICTSEL0	—	—	—	ICI1	ICI0	ICOV	ICBNE	ICM2	ICM1	ICM0	0000
IC6CON2	016A	—	—	—	—	—	—	—	IC32	ICTRIG	TRIGSTAT	—	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000D
IC6BUF	016C	Input Capture 6 Buffer Register																0000
IC6TMR	016E	Timer Value 6 Register																xxxx
IC7CON1	0170	—	—	ICSIDL	ICTSEL2	ICTSEL1	ICTSEL0	—	—	—	ICI1	ICI0	ICOV	ICBNE	ICM2	ICM1	ICM0	0000
IC7CON2	0172	—	—	—	—	—	—	—	IC32	ICTRIG	TRIGSTAT	—	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000D
IC7BUF	0174	Input Capture 7 Buffer Register																0000
IC7TMR	0176	Timer Value 7 Register																xxxx
IC8CON1	0178	—	—	ICSIDL	ICTSEL2	ICTSEL1	ICTSEL0	—	—	—	ICI1	ICI0	ICOV	ICBNE	ICM2	ICM1	ICM0	0000
IC8CON2	018A	—	—	—	—	—	—	—	IC32	ICTRIG	TRIGSTAT	—	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000D
IC8BUF	018C	Input Capture 8 Buffer Register																0000
IC8TMR	018E	Timer Value 8 Register																xxxx
IC9CON1	0180	—	—	ICSIDL	ICTSEL2	ICTSEL1	ICTSEL0	—	—	—	ICI1	ICI0	ICOV	ICBNE	ICM2	ICM1	ICM0	0000
IC9CON2	0182	—	—	—	—	—	—	—	IC32	ICTRIG	TRIGSTAT	—	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000D
IC9BUF	0184	Input Capture 9 Buffer Register																0000
IC9TMR	0186	Timer Value 9 Register																xxxx

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

TABLE 4-8: OUTPUT COMPARE 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
OC1CON1	0190	—	—	OCSIDL	OCTSEL2	OCTSEL1	OCTSEL0	ENFLT2	ENFLT1	ENFLT0	OCFLT2	OCFLT1	OCFLT0	TRIGMODE	OCM2	OCM1	OCM0	0000
OC1CON2	0192	FLTMD	FLTOUT	FLTTRIEN	OCINV	—	DCB1	DCB0	OC32	OCTRIG	TRIGSTAT	OCTRIS	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000C
OC1RS	0194	Output Compare 1 Secondary Register																0000
OC1R	0196	Output Compare 1 Register																0000
OC1TMR	0198	Timer Value 1 Register																xxxx
OC2CON1	019A	—	—	OCSIDL	OCTSEL2	OCTSEL1	OCTSEL0	ENFLT2	ENFLT1	ENFLT0	OCFLT2	OCFLT1	OCFLT0	TRIGMODE	OCM2	OCM1	OCM0	0000
OC2CON2	019C	FLTMD	FLTOUT	FLTTRIEN	OCINV	—	DCB1	DCB0	OC32	OCTRIG	TRIGSTAT	OCTRIS	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000C
OC2RS	019E	Output Compare 2 Secondary Register																0000
OC2R	01A0	Output Compare 2 Register																0000
OC2TMR	01A2	Timer Value 2 Register																xxxx
OC3CON1	01A4	—	—	OCSIDL	OCTSEL2	OCTSEL1	OCTSEL0	ENFLT2	ENFLT1	ENFLT0	OCFLT2	OCFLT1	OCFLT0	TRIGMODE	OCM2	OCM1	OCM0	0000
OC3CON2	01A6	FLTMD	FLTOUT	FLTTRIEN	OCINV	—	DCB1	DCB0	OC32	OCTRIG	TRIGSTAT	OCTRIS	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000C
OC3RS	01A8	Output Compare 3 Secondary Register																0000
OC3R	01AA	Output Compare 3 Register																0000
OC3TMR	01AC	Timer Value 3 Register																xxxx
OC4CON1	01AE	—	—	OCSIDL	OCTSEL2	OCTSEL1	OCTSEL0	ENFLT2	ENFLT1	ENFLT0	OCFLT2	OCFLT1	OCFLT0	TRIGMODE	OCM2	OCM1	OCM0	0000
OC4CON2	01B0	FLTMD	FLTOUT	FLTTRIEN	OCINV	—	DCB1	DCB0	OC32	OCTRIG	TRIGSTAT	OCTRIS	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000C
OC4RS	01B2	Output Compare 4 Secondary Register																0000
OC4R	01B4	Output Compare 4 Register																0000
OC4TMR	01B6	Timer Value 4 Register																xxxx
OC5CON1	01B8	—	—	OCSIDL	OCTSEL2	OCTSEL1	OCTSEL0	ENFLT2	ENFLT1	ENFLT0	OCFLT1	OCFLT1	OCFLT0	TRIGMODE	OCM2	OCM1	OCM0	0000
OC5CON2	01BA	FLTMD	FLTOUT	FLTTRIEN	OCINV	—	DCB1	DCB0	OC32	OCTRIG	TRIGSTAT	OCTRIS	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000C
OC5RS	01BC	Output Compare 5 Secondary Register																0000
OC5R	01BE	Output Compare 5 Register																0000
OC5TMR	01C0	Timer Value 5 Register																xxxx
OC6CON1	01C2	—	—	OCSIDL	OCTSEL2	OCTSEL1	OCTSEL0	ENFLT2	ENFLT1	ENFLT0	OCFLT2	OCFLT1	OCFLT0	TRIGMODE	OCM2	OCM1	OCM0	0000
OC6CON2	01C4	FLTMD	FLTOUT	FLTTRIEN	OCINV	—	DCB1	DCB0	OC32	OCTRIG	TRIGSTAT	OCTRIS	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000C
OC6RS	01C6	Output Compare 6 Secondary Register																0000
OC6R	01C8	Output Compare 6 Register																0000
OC6TMR	01CA	Timer Value 6 Register																xxxx

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

TABLE 4-8: OUTPUT COMPARE REGISTER MAP (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
OC7CON1	01CC	—	—	OCSIDL	OCTSEL2	OCTSEL1	OCTSEL0	ENFLT2	ENFLT1	ENFLT0	OCFLT2	OCFLT1	OCFLT0	TRIGMODE	OCM2	OCM1	OCM0	0000
OC7CON2	01CE	FLTMD	FLTOUT	FLTTRIEN	OCINV	—	DCB1	DCB0	OC32	OCTRIG	TRIGSTAT	OCTRIS	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000C
OC7RS	01D0	Output Compare 7 Secondary Register																0000
OC7R	01D2	Output Compare 7 Register																0000
OC7TMR	01D4	Timer Value 7 Register																xxxx
OC8CON1	01D6	—	—	OCSIDL	OCTSEL2	OCTSEL1	OCTSEL0	ENFLT2	ENFLT1	ENFLT0	OCFLT2	OCFLT1	OCFLT0	TRIGMODE	OCM2	OCM1	OCM0	0000
OC8CON2	01D8	FLTMD	FLTOUT	FLTTRIEN	OCINV	—	DCB1	DCB0	OC32	OCTRIG	TRIGSTAT	OCTRIS	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000C
OC8RS	01DA	Output Compare 8 Secondary Register																0000
OC8R	01DC	Output Compare 8 Register																0000
OC8TMR	01DE	Timer Value 8 Register																xxxx
OC9CON1	01E0	—	—	OCSIDL	OCTSEL2	OCTSEL1	OCTSEL0	ENFLT2	ENFLT1	ENFLT0	OCFLT2	OCFLT1	OCFLT0	TRIGMODE	OCM2	OCM1	OCM0	0000
OC9CON2	01E2	FLTMD	FLTOUT	FLTTRIEN	OCINV	—	DCB1	DCB0	OC32	OCTRIG	TRIGSTAT	OCTRIS	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000C
OC9RS	01E4	Output Compare 9 Secondary Register																0000
OC9R	01E6	Output Compare 9 Register																0000
OC9TMR	01E8	Timer Value 9 Register																xxxx

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

TABLE 4-9: I^2C™ 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
I2C1RCV	0200	—	—	—	—	—	—	—	—	Receive Register								0000
I2C1TRN	0202	—	—	—	—	—	—	—	—	Transmit Register								00FF
I2C1BRG	0204	—	—	—	—	—	—	—	Baud Rate Generator Register									0000
I2C1CON	0206	I2CEN	—	I2CSIDL	SCLREL	IPMIEN	A10M	DISSLW	SMEN	GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	1000
I2C1STAT	0208	ACKSTAT	TRSTAT	—	—	—	BCL	GCSTAT	ADD10	IWCOL	I2COV	DAC	P	S	R/$\overline{W}$	RBF	TBF	0000
I2C1ADD	020A	—	—	—	—	—	—	Address Register										0000
I2C1MSK	020C	—	—	—	—	—	—	Address Mask Register										0000
I2C2RCV	0210	—	—	—	—	—	—	—	—	Receive Register								0000
I2C2TRN	0212	—	—	—	—	—	—	—	—	Transmit Register								00FF
I2C2BRG	0214	—	—	—	—	—	—	—	Baud Rate Generator Register									0000
I2C2CON	0216	I2CEN	—	I2CSIDL	SCLREL	IPMIEN	A10M	DISSLW	SMEN	GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	1000
I2C2STAT	0218	ACKSTAT	TRSTAT	—	—	—	BCL	GCSTAT	ADD10	IWCOL	I2COV	DAC	P	S	R/$\overline{W}$	RBF	TBF	0000
I2C2ADD	021A	—	—	—	—	—	—	Address Register										0000
I2C2MSK	021C	—	—	—	—	—	—	Address Mask Register										0000

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

TABLE 4-10: UART REGISTER MAPS

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
U1MODE	0220	UARTEN	—	USIDL	IREN	RTSMD	—	UEN1	UEN0	WAKE	LPBACK	ABAUD	RXINV	BRGH	PDSEL1	PDSEL0	STSEL	0000
U1STA	0222	UTXISEL1	UTXINV	UTXISEL0	—	UTXBRK	UTXEN	UTXBF	TRMT	URXISEL1	URXISEL0	ADDEN	RIDLE	PERR	FERR	OERR	URXDA	0110
U1TXREG	0224	—	—	—	—	—	—	—	Transmit Register									xxxx
U1RXREG	0226	—	—	—	—	—	—	—	Receive Register									0000
U1BRG	0228	Baud Rate Generator Prescaler Register																0000
U2MODE	0230	UARTEN	—	USIDL	IREN	RTSMD	—	UEN1	UEN0	WAKE	LPBACK	ABAUD	RXINV	BRGH	PDSEL1	PDSEL0	STSEL	0000
U2STA	0232	UTXISEL1	UTXINV	UTXISEL0	—	UTXBRK	UTXEN	UTXBF	TRMT	URXISEL1	URXISEL0	ADDEN	RIDLE	PERR	FERR	OERR	URXDA	0110
U2TXREG	0234	—	—	—	—	—	—	—	Transmit Register									xxxx
U2RXREG	0236	—	—	—	—	—	—	—	Receive Register									0000
U2BRG	0238	Baud Rate Generator Prescaler Register																0000
U3MODE	0250	UARTEN	—	USIDL	IREN	RTSMD	—	UEN1	UEN0	WAKE	LPBACK	ABAUD	RXINV	BRGH	PDSEL1	PDSEL0	STSEL	0000
U3STA	0252	UTXISEL1	UTXINV	UTXISEL0	—	UTXBRK	UTXEN	UTXBF	TRMT	URXISEL1	URXISEL0	ADDEN	RIDLE	PERR	FERR	OERR	URXDA	0110
U3TXREG	0254	—	—	—	—	—	—	—	Transmit Register									xxxx
U3RXREG	0256	—	—	—	—	—	—	—	Receive Register									0000
U3BRG	0258	Baud Rate Generator Prescaler Register																0000
U4MODE	02B0	UARTEN	—	USIDL	IREN	RTSMD	—	UEN1	UEN0	WAKE	LPBACK	ABAUD	RXINV	BRGH	PDSEL1	PDSEL0	STSEL	0000
U4STA	02B2	UTXISEL1	UTXINV	UTXISEL0	—	UTXBRK	UTXEN	UTXBF	TRMT	URXISEL1	URXISEL0	ADDEN	RIDLE	PERR	FERR	OERR	URXDA	0110
U4TXREG	02B4	—	—	—	—	—	—	—	Transmit Register									xxxx
U4RXREG	02B6	—	—	—	—	—	—	—	Receive Register									0000
U4BRG	02B8	Baud Rate Generator Prescaler Register																0000

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

TABLE 4-11: SPI 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
SPI1STAT	0240	SPIEN	—	SPISIDL	—	—	SPIBEC2	SPIBEC1	SPIBEC0	SRMPT	SPIROV	SRXMPT	SISEL2	SISEL1	SISEL0	SPITBF	SPIRBF	0000
SPI1CON1	0242	—	—	—	DISSCK	DISSDO	MODE16	SMP	CKE	SSEN	CKP	MSTEN	SPRE2	SPRE1	SPRE0	PPRE1	PPRE0	0000
SPI1CON2	0244	FRMEN	SPIFSD	SPIFPOL	—	—	—	—	—	—	—	—	—	—	—	SPIFE	SPIBEN	0000
SPI1BUF	0248	Transmit and Receive Buffer																0000
SPI2STAT	0260	SPIEN	—	SPISIDL	—	—	SPIBEC2	SPIBEC1	SPIBEC0	SRMPT	SPIROV	SRXMPT	SISEL2	SISEL1	SISEL0	SPITBF	SPIRBF	0000
SPI2CON1	0262	—	—	—	DISSCK	DISSDO	MODE16	SMP	CKE	SSEN	CKP	MSTEN	SPRE2	SPRE1	SPRE0	PPRE1	PPRE0	0000
SPI2CON2	0264	FRMEN	SPIFSD	SPIFPOL	—	—	—	—	—	—	—	—	—	—	—	SPIFE	SPIBEN	0000
SPI2BUF	0268	Transmit and Receive Buffer																0000

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

TABLE 4-12: OP AMP 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
AMP1CON	024A	AMPEN	—	AMPSIDL	AMPSLP	INTPOL1	INTPOL0	CMOUT	CMPSEL	SPDSEL	AMPOE	NINSEL2	NINSEL1	NINSEL0	PINSEL2	PINSEL1	PINSEL0	0000
AMP2CON	024C	AMPEN	—	AMPSIDL	AMPSLP	INTPOL1	INTPOL0	CMOUT	CMPSEL	SPDSEL	AMPOE	NINSEL2	NINSEL1	NINSEL0	PINSEL2	PINSEL1	PINSEL0	0000

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

TABLE 4-13: PORTA REGISTER MAP(1)

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	Bit2	Bit 1	Bit 0	All Resets
TRISA	02C0	TRISA15	TRISA14	—	—	—	TRISA10	TRISA9	—	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	C6FF
PORTA	02C2	RA15	RA14	—	—	—	RA10	RA9	—	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0	xxxx
LATA	02C4	LATA15	LATA14	—	—	—	LATA10	LATA9	—	LATA7	LATA6	LATA5	LATA4	LATA3	LATA2	LATA1	LATA0	xxxx
ODCA	02C6	ODA15	ODA14	—	—	—	ODA10	ODA9	—	ODA7	ODA6	ODA5	ODA4	ODA3	ODA2	ODA1	ODA0	0000

Legend: — = unimplemented, read as '0'. Reset values are shown in hexadecimal. Reset values shown are for 100/121-pin devices.
Note 1: PORTA and all associated bits are unimplemented in 64-pin devices.

TABLE 4-14: PORTB 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
TRISB	02C8	TRISB15	TRISB14	TRISB13	TRISB12	—	—	—	—	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	F0FF
PORTB	02CA	RB15	RB14	RB13	RB12	—	—	—	—	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx
LATB	02CC	LATB15	LATB14	LATB13	LATB12	—	—	—	—	LATB7	LATB6	LATB5	LATB4	LATB3	LATB2	LATB1	LATB0	xxxx
ODCB	02CE	ODB15	ODB14	ODB13	ODB12	—	—	—	—	ODB7	ODB6	ODB5	ODB4	ODB3	ODB2	ODB1	ODB0	0000

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

TABLE 4-15: PORTC 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(1)	Bit 3(1)	Bit 2(1)	Bit 1(1)	Bit 0	All Resets
TRISC	02D0	TRISC15	—	—	TRISC12	—	—	—	—	—	—	—	TRISC4	TRISC3	TRISC2	TRISC1	—	901E
PORTC	02D2	RC15(2,3)	RC14(4)	RC13(4)	RC12(2)	—	—	—	—	—	—	—	RC4	RC3	RC2	RC1	—	xxxx
LATC	02D4	LATC15	—	—	LATC12	—	—	—	—	—	—	—	LATC4	LATC3	LATC2	LATC1	—	xxxx
ODCC	02D6	ODC15	—	—	ODC12	—	—	—	—	—	—	—	ODC4	ODC3	ODC2	ODC1	—	0000

Legend: — = unimplemented, read as '0'. Reset values are shown in hexadecimal. Reset values shown are for 100/121-pin devices.
Note 1: These bits are unimplemented in 64-pin devices, read as '0'.
2: RC12 and RC15 are only available when the primary oscillator is disabled or when EC mode is selected (POSCMD<1:0> Configuration bits = 11 or 00); otherwise read as '0'.
3: RC15 is only available when the POSCMD<1:0> Configuration bits = 11 or 00 and the OSCIOFN Configuration bit = 1.
4: RC13 and RC14 are input ports only and cannot be used as output ports.

TABLE 4-16: PORTD REGISTER MAP

File Name	Addr	Bit 15(1)	Bit 14(1)	Bit 13(1)	Bit 12(1)	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
TRISD	02D8	TRISD15	TRISD14	TRISD13	TRISD12	TRISD11	TRISD10	TRISD9	TRISD8	TRISD7	TRISD6	TRISD5	TRISD4	TRISD3	TRISD2	TRISD1	TRISD0	FFFF
PORTD	02DA	RD15	RD14	RD13	RD12	RD11	RD10	RD9	RD8	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	xxxx
LATD	02DC	LATD15	LATD14	LATD13	LATD12	LATD11	LATD10	LATD9	LATD8	LATD7	LATD6	LATD5	LATD4	LATD3	LATD2	LATD1	LATD0	xxxx
ODCD	02DE	ODD15	ODD14	ODD13	ODD12	ODD11	ODD10	ODD9	ODD8	ODD7	ODD6	ODD5	ODD4	ODD3	ODD2	ODD1	ODD0	0000

Legend: — = unimplemented, read as '0'. Reset values are shown in hexadecimal. Reset values shown are for 100/121-pin devices.
Note 1: These bits are unimplemented in 64-pin devices, read as '0'.

TABLE 4-17: PORTE REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9(1)	Bit 8(1)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
TRISE	02E0	—	—	—	—	—	—	TRISE9	TRISE8	TRISE7	TRISE6	TRISE5	TRISE4	TRISE3	TRISE2	TRISE1	TRISE0	03FF
PORTE	02E2	—	—	—	—	—	—	RE9	RE8	RE7	RE6	RE5	RE4	RE3	RE2	RE1	RE0	xxxx
LATE	02E4	—	—	—	—	—	—	LATE9	LATE8	LATE7	LATE6	LATE5	LATE4	LATE3	LATE2	LATE1	LATE0	xxxx
ODCE	02E6	—	—	—	—	—	—	ODE9	ODE8	ODE7	ODE6	ODE5	ODE4	ODE3	ODE2	ODE1	ODE0	0000

Legend: — = unimplemented, read as '0'. Reset values are shown in hexadecimal. Reset values shown are for 100/121-pin devices.
Note 1: These bits are unimplemented in 64-pin devices, read as '0'.

TABLE 4-18: PORTF REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13(1)	Bit 12(1)	Bit 11	Bit 10	Bit 9	Bit 8(1)	Bit 7(1)	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
TRISF	02E8	—	—	TRISF13	TRISF12	—	—	—	TRISF8	TRISF7	—	TRISF5	TRISF4	TRISF3	TRISF2	TRISF1	TRISF0	31BF
PORTF	02EA	—	—	RF13	RF12	—	—	—	RF8	RF7	—	RF5	RF4	RF3	RF2	RF1	RF0	xxxx
LATF	02EC	—	—	LATF13	LATF12	—	—	—	LATF8	LATF7	—	LATF5	LATF4	LATF3	LATF2	LATF1	LATF0	xxxx
ODCF	02EE	—	—	ODF13	ODF12	—	—	—	ODF8	ODF7	—	ODF5	ODF4	ODF3	ODF2	ODF1	ODF0	0000

Legend: — = unimplemented, read as '0'. Reset values are shown in hexadecimal. Reset values shown are for 100/121-pin devices.
Note 1: These bits are unimplemented in 64-pin devices, read as '0'.

TABLE 4-19: PORTG REGISTER MAP

File Name	Addr	Bit 15(1)	Bit 14(1)	Bit 13(1)	Bit 12(1)	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1(1)	Bit 0(1)	All Resets
TRISG	02F0	TRISG15	TRISG14	TRISG13	TRISG12	—	—	TRISG9	TRISG8	TRISG7	TRISG6	—	—	TRISG3	TRISG2	TRISG1	TRISG0	F3CF
PORTG	02F2	RG15	RG14	RG13	RG12	—	—	RG9	RG8	RG7	RG6	—	—	RG3	RG2	RG1	RG0	xxxx
LATG	02F4	LATG15	LATG14	LATG13	LATG12	—	—	LATG9	LATG8	LATG7	LATG6	—	—	LATG3	LATG2	LATG1	LATG0	xxxx
ODCG	02F6	ODG15	ODG14	ODG13	ODG12	—	—	ODG9	ODG8	ODG7	ODG6	—	—	ODG3	ODG2	ODG1	ODG0	0000

Legend: — = unimplemented, read as '0'. Reset values are shown in hexadecimal. Reset values shown are for 100/121-pin devices.
Note 1: These bits are unimplemented in 64-pin devices, read as '0'.

TABLE 4-20: PAD CONFIGURATION REGISTER MAP (PADCFG1)

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
PADCFG1	02FC	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PMPTTL	0000

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

TABLE 4-21: CTMU 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
CTMUCON1	035A	CTMUEN	—	CTMUSIDL	TGEN	EDGEN	EDGSEQEN	IDISSEN	CTTRIG	—	—	—	—	—	—	—	—	0000
CTMUCON2	035C	EDG1MOD	EDG1POL	EDG1SEL3	EDG1SEL2	EDG1SEL1	EDG1SEL0	EDG2STAT	EDG1STAT	EDG2MOD	EDG2POL	EDG2SEL3	EDG2SEL2	EDG2SEL1	EDG2SEL0	—	—	0000
CTMUICON	035E	ITRIM5	ITRIM4	ITRIM3	ITRIM2	ITRIM1	ITRIM0	IRNG1	IRNG0	—	—	—	—	—	—	—	—	0000

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

TABLE 4-22: DAC 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
DAC1CON	0440	DACEN	—	DACSIDL	DACSLP	DACFM	—	—	DACTRIG	—	DACTSEL4	DACTSEL3	DACTSEL2	DACTSEL1	DACTSEL0	DACREF1	DACREF0	0000
DAC1DAT	0442	DAC1 Input Value Register																0000
DAC2CON	0444	DACEN	—	DACSIDL	DACSLP	DACFM	—	—	DACTRIG	—	DACTSEL4	DACTSEL3	DACTSEL2	DACTSEL1	DACTSEL0	DACREF1	DACREF0	0000
DAC2DAT	0446	DAC2 Input Value Register																0000

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

TABLE 4-23: SIGMA-DELTA A/D 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
SD1CON1	04D0	SDON	—	SDSIDL	SDRST	FILTDIS	SDGAIN2	SDGAIN1	SDGAIN0	DITHER1	DITHER0	—	VOSCAL	—	SDREFN	SDREFP	PWRLVL	0000
SD1CON2	04D2	CHOP1	CHOP0	SDINT1	SDINT0	—	—	SDWM1	SDWM0	—	—	—	RNDRES1	RNDRES0	—	—	SDRDY	0000
SD1CON3	04D4	SDDIV2	SDDIV1	SDDIV0	SDOSR2	SDOSR1	SDOSR0	SDCS1	SDCS0	—	—	—	—	—	SDCH2	SDCH1	SDCH0	0000
SD1RESH	04D6	Sigma-Delta A/D Result Register (bits<31-16>)																0000
SD1RESL	04D8	Sigma-Delta A/D Result Register (bits<15-0>)																0000

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

TABLE 4-24: ANALOG CONFIGURATION 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
ANCFG	04DE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VBG2EN	—	0000
ANSA	04E0	ANSA15(1)	ANSA14(1)	—	—	—	ANSA10(1)	ANSA9(1)	—	ANSA7(1)	ANSA6(1)	ANSA5(1)	ANSA4(1)	—	—	ANSA1(1)	—	C6F2
ANSB	04E2	ANSB15	ANSB14	ANSB13	ANSB12	—	—	—	—	ANSB7	ANSB6	ANSB5	ANSB4	ANSB3	ANSB2	ANSB1	ANSB0	F0FF
ANSC	04E4	—	—	—	—	—	—	—	—	—	—	—	ANSC4(1)	ANSC3(1)	—	ANSC1(1)	—	001A
ANSD	04E6	ANSD15(1)	ANSD14(1)	ANSD13(1)	ANSD12(1)	ANSD11	ANSD10	ANSD9	ANSD8	ANSD7	ANSD6	ANSD5	ANSD4	ANSD3	ANSD2	—	ANSD0	FFFD
ANSE	04E8	—	—	—	—	—	—	ANSE9(1)	—	ANSE7	ANSE6	ANSE5	ANSE4	—	—	—	—	02F0
ANSF	04EA	—	—	ANSF13(1)	—	—	—	—	ANSF8(1)	ANSF7	—	ANSF5	ANSF4	ANSF3	ANSF2(1)	—	ANSF0	21BD
ANSG	04EC	ANSG15(1)	—	—	—	—	—	ANSG9	ANSG8	ANSG7	ANSG6	—	—	—	—	—	—	83C0

Legend: — = unimplemented, read as '0'. Reset values are shown in hexadecimal.
Note 1: These bits are unimplemented in 64-pin devices, read as '0'.

TABLE 4-25: 12-BIT PIPELINE A/D CONVERTER 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
ADCON1	0500	ADON	—	ADSIDL	ADSLP	FORM3	FORM2	FORM1	FORM0	PUMPEN	ADCAL	—	—	—	—	—	PWRLVL	0000
ADCON2	0502	PVCFG1	PVCFG0	—	NVCFG0	—	BUFORG	r	r	r	r	—	—	—	—	RFPUMP	r	0300
ADCON3	0504	ADRC	—	—	—	SLEN3	SLEN2	SLEN1	SLEN0	ADCS7	ADCS6	ADCS5	ADCS4	ADCS3	ADCS2	ADCS1	ADCS0	0000
ADTMRPR	0506	A/D Trigger Timer Period Value																0000
ADSTATH	050A	—	—	—	—	—	—	—	—	—	—	—	—	—	PUMPST	ADREADY	ADBUSY	0000
ADSTATL	0508	—	—	—	—	—	—	—	SLOV	—	—	r	ACCIF	SL3IF	SL2IF	SL1IF	SL0IF	0000
ADL0CONH	051E	ASEN	SLINT1	SLINT0	WM1	WM0	CM2	CM1	CM0	CTMEN	r	MULCHEN	SAMC4	SAMC3	SAMC2	SAMC1	SAMC0	0000
ADL0CONL	051C	SLEN	SAMP	SLENCLR	SLTSRC4	SLTSRC3	SLTSRC2	SLTSRC1	SLTSRC0	THSRC	—	—	SLSIZE4	SLSIZE3	SLSIZE2	SLSIZE1	SLSIZE0	0000
ADL1CONH	053A	ASEN	SLINT1	SLINT0	WM1	WM0	CM2	CM1	CM0	CTMEN	r	MULCHEN	SAMC4	SAMC3	SAMC2	SAMC1	SAMC0	0000
ADL1CONL	0538	SLEN	SAMP	SLENCLR	SLTSRC4	SLTSRC3	SLTSRC2	SLTSRC1	SLTSRC0	THSRC	—	—	SLSIZE4	SLSIZE3	SLSIZE2	SLSIZE1	SLSIZE0	0000
ADL2CONH	0556	ASEN	SLINT1	SLINT0	WM1	WM0	CM2	CM1	CM0	CTMEN	r	MULCHEN	SAMC4	SAMC3	SAMC2	SAMC1	SAMC0	0000
ADL2CONL	0554	SLEN	SAMP	SLENCLR	SLTSRC4	SLTSRC3	SLTSRC2	SLTSRC1	SLTSRC0	THSRC	—	—	SLSIZE4	SLSIZE3	SLSIZE2	SLSIZE1	SLSIZE0	0000
ADL3CONH	0402	ASEN	SLINT1	SLINT0	WM1	WM0	CM2	CM1	CM0	CTMEN	r	MULCHEN	SAMC4	SAMC3	SAMC2	SAMC1	SAMC0	0000
ADL3CONL	0400	SLEN	SAMP	SLENCLR	SLTSRC4	SLTSRC3	SLTSRC2	SLTSRC1	SLTSRC0	THSRC	—	—	SLSIZE4	SLSIZE3	SLSIZE2	SLSIZE1	SLSIZE0	0000
ADL0PTR	0520	—	ADNEXT6	ADNEXT5	ADNEXT4	ADNEXT3	ADNEXT2	ADNEXT1	ADNEXT0	—	—	—	—	—	—	—	—	0000
ADL1PTR	053C	—	ADNEXT6	ADNEXT5	ADNEXT4	ADNEXT3	ADNEXT2	ADNEXT1	ADNEXT0	—	—	—	—	—	—	—	—	0000
ADL2PTR	0558	—	ADNEXT6	ADNEXT5	ADNEXT4	ADNEXT3	ADNEXT2	ADNEXT1	ADNEXT0	—	—	—	—	—	—	—	—	0000
ADL3PTR	0404	—	ADNEXT6	ADNEXT5	ADNEXT4	ADNEXT3	ADNEXT2	ADNEXT1	ADNEXT0	—	—	—	—	—	—	—	—	0000
ADL0STAT	0522	ADTACT	LBUSY	—	—	—	—	—	—	ADTDLY	—	ADLIF	—	—	—	—	—	0000
ADL1STAT	053E	ADTACT	LBUSY	—	—	—	—	—	—	ADTDLY	—	ADLIF	—	—	—	—	—	0000
ADL2STAT	055A	ADTACT	LBUSY	—	—	—	—	—	—	ADTDLY	—	ADLIF	—	—	—	—	—	0000
ADL3STAT	0406	ADTACT	LBUSY	—	—	—	—	—	—	ADTDLY	—	ADLIF	—	—	—	—	—	0000

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

TABLE 4-25: 12-BIT PIPELINE A/D CONVERTER REGISTER MAP (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
ADTBL0	0300	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL1	0302	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL2	0304	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL3	0306	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL4	0308	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL5	030A	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL6	030C	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL7	030E	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL8	0310	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL9	0312	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL10	0314	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL11	0316	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL12	0318	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL13	031A	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL14	031C	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL15	031E	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL16	0320	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL17	0322	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL18	0324	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL19	0326	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL20	0328	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL21	032A	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL22	032C	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL23	032E	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL24	0330	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL25	0332	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL26	0334	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL27	0336	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL28	0338	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL29	033A	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL30	033C	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000
ADTBL31	033E	UCTMU	DIFF	—	—	—	—	—	—	—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	0000

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

TABLE 4-25: 12-BIT PIPELINE A/D CONVERTER REGISTER MAP (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
ADRES0	0340	A/D Result Register 0																0000
ADRES1	0342	A/D Result Register 1																0000
ADRES2	0344	A/D Result Register 2																0000
ADRES3	0346	A/D Result Register 3																0000
ADRES4	0348	A/D Result Register 4																0000
ADRES5	034A	A/D Result Register 5																0000
ADRES6	034C	A/D Result Register 6																0000
ADRES7	034E	A/D Result Register 7																0000
ADRES8	0350	A/D Result Register 8																0000
ADRES9	0352	A/D Result Register 9																0000
ADRES10	0354	A/D Result Register 10																0000
ADRES11	0356	A/D Result Register 11																0000
ADRES12	0358	A/D Result Register 12																0000
ADRES13	035A	A/D Result Register 13																0000
ADRES14	035C	A/D Result Register 14																0000
ADRES15	035E	A/D Result Register 15																0000
ADRES16	0360	A/D Result Register 16																0000
ADRES17	0362	A/D Result Register 17																0000
ADRES18	0364	A/D Result Register 18																0000
ADRES19	0366	A/D Result Register 19																0000
ADRES20	0368	A/D Result Register 20																0000
ADRES21	036A	A/D Result Register 21																0000
ADRES22	036C	A/D Result Register 22																0000
ADRES23	036E	A/D Result Register 23																0000
ADRES24	0370	A/D Result Register 24																0000
ADRES25	0372	A/D Result Register 25																0000
ADRES26	0374	A/D Result Register 26																0000
ADRES27	0376	A/D Result Register 27																0000
ADRES28	0378	A/D Result Register 28																0000
ADRES29	037A	A/D Result Register 29																0000
ADRES30	037C	A/D Result Register 30																0000
ADRES31	037E	A/D Result Register 31																0000

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

TABLE 4-25: 12-BIT PIPELINE A/D CONVERTER REGISTER MAP (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
ACCONH	050E	—	—	—	—	—	—	—	—	ACEN	ACIE	—	—	—	—	—	—	0000
ACCONL	050C	—	—	TBLSEL5	TBLSEL4	TBLSEL3	TBLSEL2	TBLSEL1	TBLSEL0	COUNT7	COUNT6	COUNT5	COUNT4	COUNT3	COUNT2	COUNT1	COUNT0	0000
ACRESH	0512	A/D Accumulation High Result Register (bits<31-16>)																0000
ACRESL	0510	A/D Accumulation Low Result Register (bits<15-0>)																0000
ADCHITH	0516	CHH31	CHH30	CHH29	CHH28	CHH27	CHH26	CHH25	CHH24	CHH23	CHH22	CHH21	CHH20	CHH19	CHH18	CHH17	CHH16	0000
ADCHITL	0514	CHH15	CHH14	CHH13	CHH12	CHH11	CHH10	CHH9	CHH8	CHH7	CHH6	CHH5	CHH4	CHH3	CHH2	CHH1	CHH0	0000
ADTH0H	0526	TH15	TH14	TH13	TH12	TH11	TH10	TH9	TH8	TH7	TH6	TH5	TH4	TH3	TH2	TH1	TH0	0000
ADTH0L	0524	TH15	TH14	TH13	TH12	TH11	TH10	TH9	TH8	TH7	TH6	TH5	TH4	TH3	TH2	TH1	TH0	0000
ADTH1H	0542	TH15	TH14	TH13	TH12	TH11	TH10	TH9	TH8	TH7	TH6	TH5	TH4	TH3	TH2	TH1	TH0	0000
ADTH1L	0540	TH15	TH14	TH13	TH12	TH11	TH10	TH9	TH8	TH7	TH6	TH5	TH4	TH3	TH2	TH1	TH0	0000
ADTH2H	055E	TH15	TH14	TH13	TH12	TH11	TH10	TH9	TH8	TH7	TH6	TH5	TH4	TH3	TH2	TH1	TH0	0000
ADTH2L	055C	TH15	TH14	TH13	TH12	TH11	TH10	TH9	TH8	TH7	TH6	TH5	TH4	TH3	TH2	TH1	TH0	0000
ADTH3H	040A	TH15	TH14	TH13	TH12	TH11	TH10	TH9	TH8	TH7	TH6	TH5	TH4	TH3	TH2	TH1	TH0	0000
ADTH3L	0408	TH15	TH14	TH13	TH12	TH11	TH10	TH9	TH8	TH7	TH6	TH5	TH4	TH3	TH2	TH1	TH0	0000
ADL0MSEL3	052E	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MSEL49	MSEL48	0000
ADL0MSEL2	052C	MSEL47	MSEL46	MSEL45	MSEL44	MSEL43	MSEL42	MSEL41	MSEL40	MSEL39	MSEL38	MSEL37	MSEL36	MSEL35	MSEL34	MSEL33	MSEL32	0000
ADL0MSEL1	052A	MSEL31	MSEL30	MSEL29	MSEL28	MSEL27	MSEL26	MSEL25	MSEL24	MSEL23	MSEL22	MSEL21	MSEL20	MSEL19	MSEL18	MSEL17	MSEL16	0000
ADL0MSEL0	0528	MSEL15	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
ADL1MSEL3	054A	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MSEL49	MSEL48	0000
ADL1MSEL2	0548	MSEL47	MSEL46	MSEL45	MSEL44	MSEL43	MSEL42	MSEL41	MSEL40	MSEL39	MSEL38	MSEL37	MSEL36	MSEL35	MSEL34	MSEL33	MSEL32	0000
ADL1MSEL1	0546	MSEL31	MSEL30	MSEL29	MSEL28	MSEL27	MSEL26	MSEL25	MSEL24	MSEL23	MSEL22	MSEL21	MSEL20	MSEL19	MSEL18	MSEL17	MSEL16	0000
ADL1MSEL0	0544	MSEL15	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
ADL2MSEL3	0566	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MSEL49	MSEL48	0000
ADL2MSEL2	0564	MSEL47	MSEL46	MSEL45	MSEL44	MSEL43	MSEL42	MSEL41	MSEL40	MSEL39	MSEL38	MSEL37	MSEL36	MSEL35	MSEL34	MSEL33	MSEL32	0000
ADL2MSEL1	0562	MSEL31	MSEL30	MSEL29	MSEL28	MSEL27	MSEL26	MSEL25	MSEL24	MSEL23	MSEL22	MSEL21	MSEL20	MSEL19	MSEL18	MSEL17	MSEL16	0000
ADL2MSEL0	0560	MSEL15	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
ADL3MSEL3	0412	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MSEL49	MSEL48	0000
ADL3MSEL2	0410	MSEL47	MSEL46	MSEL45	MSEL44	MSEL43	MSEL42	MSEL41	MSEL40	MSEL39	MSEL38	MSEL37	MSEL36	MSEL35	MSEL34	MSEL33	MSEL32	0000
ADL3MSEL1	040E	MSEL31	MSEL30	MSEL29	MSEL28	MSEL27	MSEL26	MSEL25	MSEL24	MSEL23	MSEL22	MSEL21	MSEL20	MSEL19	MSEL18	MSEL17	MSEL16	0000
ADL3MSEL0	040C	MSEL15	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000

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

TABLE 4-26: 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
DMACON	0380	DMAEN	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PRSSEL	0000
DMABUF	0382	DMA Transfer Data Buffer																0000
DMAL	0384	DMA High Address Limit																0000
DMAH	0386	DMA Low Address Limit																0000
DMACH0	0388	—	—	—	—	—	NULLW	RELOAD	CHREQ	SAMODE1	SAMODE0	DAMODE1	DAMODE0	TRMODE1	TRMODE0	SIZE	CHEN	0000
DMAINT0	038A	DBUFWF	—	CHSEL5	CHSEL4	CHSEL3	CHSEL2	CHSEL1	CHSEL0	HIGHIF	LOWIF	DONEIF	HALFIF	OVRUNIF	—	—	HALFEN	0000
DMASRC0	038C	DMA Channel 0 Source Address																0000
DMADST0	038E	DMA Channel 0 Destination Address																0000
DMACNT0	0390	DMA Channel 0 Transaction Count																0001
DMACH1	0392	—	—	—	—	—	NULLW	RELOAD	CHREQ	SAMODE1	SAMODE0	DAMODE1	DAMODE0	TRMODE1	TRMODE0	SIZE	CHEN	0000
DMAINT1	0394	DBUFWF	—	CHSEL5	CHSEL4	CHSEL3	CHSEL2	CHSEL1	CHSEL0	HIGHIF	LOWIF	DONEIF	HALFIF	OVRUNIF	—	—	HALFEN	0000
DMASRC1	0396	DMA Channel 1 Source Address																0000
DMADST1	0398	DMA Channel 1 Destination Address																0000
DMACNT1	039A	DMA Channel 1 Transaction Count																0001
DMACH2	039C	—	—	—	—	—	NULLW	RELOAD	CHREQ	SAMODE1	SAMODE0	DAMODE1	DAMODE0	TRMODE1	TRMODE0	SIZE	CHEN	0000
DMAINT2	039E	DBUFWF	—	CHSEL5	CHSEL4	CHSEL3	CHSEL2	CHSEL1	CHSEL0	HIGHIF	LOWIF	DONEIF	HALFIF	OVRUNIF	—	—	HALFEN	0000
DMASRC2	03A0	DMA Channel 2 Source Address																0000
DMADST2	03A2	DMA Channel 2 Destination Address																0000
DMACNT2	03A4	DMA Channel 2 Transaction Count																0001
DMACH3	03A6	—	—	—	—	—	NULLW	RELOAD	CHREQ	SAMODE1	SAMODE0	DAMODE1	DAMODE0	TRMODE1	TRMODE0	SIZE	CHEN	0000
DMAINT3	03A8	DBUFWF	—	CHSEL5	CHSEL4	CHSEL3	CHSEL2	CHSEL1	CHSEL0	HIGHIF	LOWIF	DONEIF	HALFIF	OVRUNIF	—	—	HALFEN	0000
DMASRC3	03AA	DMA Channel 3 Source Address																0000
DMADST3	03AC	DMA Channel 3 Destination Address																0000
DMACNT3	03AE	DMA Channel 3 Transaction Count																0001
DMACH4	03B0	—	—	—	—	—	NULLW	RELOAD	CHREQ	SAMODE1	SAMODE0	DAMODE1	DAMODE0	TRMODE1	TRMODE0	SIZE	CHEN	0000
DMAINT4	03B2	DBUFWF	—	CHSEL5	CHSEL4	CHSEL3	CHSEL2	CHSEL1	CHSEL0	HIGHIF	LOWIF	DONEIF	HALFIF	OVRUNIF	—	—	HALFEN	0000
DMASRC4	03B4	DMA Channel 4 Source Address																0000
DMADST4	03B6	DMA Channel 4 Destination Address																0000
DMACNT4	03B8	DMA Channel 4 Transaction Count																0001
DMACH5	03BA	—	—	—	—	—	NULLW	RELOAD	CHREQ	SAMODE1	SAMODE0	DAMODE1	DAMODE0	TRMODE1	TRMODE0	SIZE	CHEN	0000
DMAINT5	03BC	DBUFWF	—	CHSEL5	CHSEL4	CHSEL3	CHSEL2	CHSEL1	CHSEL0	HIGHIF	LOWIF	DONEIF	HALFIF	OVRUNIF	—	—	HALFEN	0000
DMASRC5	03BE	DMA Channel 5 Source Address																0000
DMADST5	03C0	DMA Channel 5 Destination Address																0000
DMACNT5	03C2	DMA Channel 5 Transaction Count																0001

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

TABLE 4-27: USB OTG 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
U1OTGIR	0480	—	—	—	—	—	—	—	—	IDIF	T1MSECIF	LSTATEIF	ACTVIF	SESVDIF	SESENDIF	—	VBUSVDIF	0000
U1OTGIE	0482	—	—	—	—	—	—	—	—	IDIE	T1MSECIE	LSTATEIE	ACTVIE	SESVDIE	SESENDIE	—	VBUSVDIE	0000
U1OTGSTAT	0484	—	—	—	—	—	—	—	—	ID	—	LSTATE	—	SESVD	SESEND	—	VBUSVD	0000
U1OTGCON	0486	—	—	—	—	—	—	—	—	DPPULUP	DMPULUP	DPPULDWN	DMPULDWN	r	OTGEN	r	VBUSDIS	0000
U1PWRC	0488	—	—	—	—	—	—	—	—	UACTPND	—	—	USLPGRD	—	—	USUSPND	USBPWR	00x0
U1IR	048A[1]	—	—	—	—	—	—	—	—	STALLIF	—	RESUMEIF	IDLEIF	TRNIF	SOFIF	UERRIF	URSTIF	0000
		—	—	—	—	—	—	—	—	STALLIF	ATTACHIF[1]	RESUMEIF	IDLEIF	TRNIF	SOFIF	UERRIF	DETACHIF[1]	0000
U1IE	048C[1]	—	—	—	—	—	—	—	—	STALLIE	—	RESUMEIE	IDLEIE	TRNIE	SOFIE	UERRIE	URSTIE	0000
		—	—	—	—	—	—	—	—	STALLIE	ATTACHIE[1]	RESUMEIE	IDLEIE	TRNIE	SOFIE	UERRIE	DETACHIE[1]	0000
U1EIR	048E[1]	—	—	—	—	—	—	—	—	BTSEF	—	DMAEF	BTOEF	DFN8EF	CRC16EF	CRC5EF	PIDEF	0000
		—	—	—	—	—	—	—	—	BTSEF	—	DMAEF	BTOEF	DFN8EF	CRC16EF	EOFEF[1]	PIDEF	0000
U1EIE	0490[1]	—	—	—	—	—	—	—	—	BTSEE	—	DMAEE	BTOEE	DFN8EE	CRC16EE	CRC5EE	PIDEE	0000
		—	—	—	—	—	—	—	—	BTSEE	—	DMAEE	BTOEE	DFN8EE	CRC16EE	EOFEE[1]	PIDEE	0000
U1STAT	0492	—	—	—	—	—	—	—	—	ENDPT3	ENDPT2	ENDPT1	ENDPT0	DIR	PPBI	—	—	0000
U1CON	0494[1]	—	—	—	—	—	—	—	—	—	SE0	PKTDIS	—	HOSTEN	RESUME	PPBRST	USBEN	0000
		—	—	—	—	—	—	—	—	JSTATE[1]	SE0	TOKBUSY	USBRST	HOSTEN	RESUME	PPBRST	SOFEN[1]	0000
U1ADDR	0496	—	—	—	—	—	—	—	—	LSPDEN[1]	USB Device Address (ADDR) Register							0000
U1BDTP1	0498	—	—	—	—	—	—	—	—	Buffer Descriptor Table Base Address Register							—	0000
U1FRML	049A	—	—	—	—	—	—	—	—	Frame Count Register Low Byte								0000
U1FRMH	049C	—	—	—	—	—	—	—	—	Frame Count Register High Byte								0000
U1TOK[2]	049E	—	—	—	—	—	—	—	—	PID3	PID2	PID1	PID0	EP3	EP2	EP1	EP0	0000
U1SOF[2]	04A0	—	—	—	—	—	—	—	—	Start-of-Frame Count Register								0000
U1CNFG1	04A6	—	—	—	—	—	—	—	—	UTEYE	UOEMON	—	USBSIDL	—	—	PPB1	PPB0	0000
U1CNFG2	04A8	—	—	—	—	—	—	—	—	—	—	UVCMPSEL	PUVBUS	EXTI2CEN	—	UVCMPDIS	UTRDIS	0000
U1EP0	04AA	—	—	—	—	—	—	—	—	LSPD[1]	RETRYDIS[1]	—	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
U1EP1	04AC	—	—	—	—	—	—	—	—	—	—	—	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
U1EP2	04AE	—	—	—	—	—	—	—	—	—	—	—	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
U1EP3	04B0	—	—	—	—	—	—	—	—	—	—	—	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
U1EP4	04B2	—	—	—	—	—	—	—	—	—	—	—	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
U1EP5	04B4	—	—	—	—	—	—	—	—	—	—	—	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
U1EP6	04B6	—	—	—	—	—	—	—	—	—	—	—	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
U1EP7	04B8	—	—	—	—	—	—	—	—	—	—	—	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
U1EP8	04BA	—	—	—	—	—	—	—	—	—	—	—	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
U1EP9	04BC	—	—	—	—	—	—	—	—	—	—	—	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
U1EP10	04BE	—	—	—	—	—	—	—	—	—	—	—	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000

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

Note 1: Alternate register or bit definitions when the module is operating in Host mode.

2: This register is available in Host mode only.

TABLE 4-27: USB OTG REGISTER MAP (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
U1EP11	04C0	—	—	—	—	—	—	—	—	—	—	—	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
U1EP12	04C2	—	—	—	—	—	—	—	—	—	—	—	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
U1EP13	04C4	—	—	—	—	—	—	—	—	—	—	—	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
U1EP14	04C6	—	—	—	—	—	—	—	—	—	—	—	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000
U1EP15	04C8	—	—	—	—	—	—	—	—	—	—	—	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK	0000

Legend: — = unimplemented, read as '0', r = reserved. Reset values are shown in hexadecimal.
Note 1: Alternate register or bit definitions when the module is operating in Host mode.
2: This register is available in Host mode only.

TABLE 4-28: LCD CONTROLLER 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
LCDREG	0580	CPEN	—	—	—	—	—	—	—	—	—	BIAS2	BIAS1	BIAS0	MODE13	CKSEL1	CKSEL0	0000
LCDREF	0582	LCDIRE	—	LCDCST2	LCDCST1	LCDCST0	VLCD3PE	VLCD2PE	VLCD1PE	LRLAP1	LRLAP0	LRLBP1	LRLBP0	—	LRLAT2	LRLAT1	LRLAT0	0000
LCDCON	0584	LCDEN	—	LCDSIDL	—	—	—	—	—	—	SLPEN	WERR	CS1	CS0	LMUX2	LMUX1	LMUX0	0000
LCDPS	0586	—	—	—	—	—	—	—	—	WFT	BIASMD	LCDA	WA	LP3	LP2	LP1	LP0	0000
LCDSE0	0588	SE15	SE14	SE13	SE12	SE11	SE10	SE09	SE08	SE07	SE06	SE05	SE04	SE03	SE02	SE01	SE00	0000
LCDSE1	058A	SE31(1)	SE30	SE29(1)	SE28(11)	SE27	SE26	SE25	SE24	SE23	SE22	SE21	SE20	SE019	SE18	SE17	SE16	0000
LCDSE2	058C	SE47	SE46(1)	SE45(1)	SE44(1)	SE43(1)	SE42(1)	SE41(1)	SE40(1)	SE39(1)	SE38(1)	SE37(1)	SE36(1)	SE35(1)	SE34(1)	SE33(1)	SE32(1)	0000
LCDSE3	058E	—	SE62	SE61(1)	SE60(1)	SE59(1)	SE58(1)	SE57(1)	SE56(1)	SE55(1)	SE54(1)	SE53(1)	SE52(1)	SE51(1)	SE50(1)	SE49(1)	SE48(1)	0000
LCDDATA0	0590	S15C0	S14C0	S13C0	S12C0	S11C0	S10C0	S09C0	S08C0	S07C0	S06C0	S05C0	S04C0	S03C0	S02C0	S01C0	S00C0	0000
LCDDATA1	0592	S31C0(1)	S30C0	S29C0(1)	S28C0(1)	S27C0	S26C0	S25C0	S24C0	S23C0	S22C0	S21C0	S20C0	S19C0	S18C0	S17C0	S16C0	0000
LCDDATA2	0594	S47C0	S46C0(1)	S45C0(1)	S44C0(1)	S43C0(1)	S42C0(1)	S41C0(1)	S40C0(1)	S39C0(1)	S38C0(1)	S37C0(1)	S36C0(1)	S35C0(1)	S34C0(1)	S33C0(1)	S32C0(1)	0000
LCDDATA3	0596	—	S62C0	S61C0(1)	S60C0(1)	S59C0(1)	S58C0(1)	S57C0(1)	S56C0(1)	S55C0(1)	S54C0(1)	S53C0(1)	S52C0(1)	S51C0(1)	S50C0(1)	S49C0(1)	S48C0	0000
LCDDATA4	0598	S15C1	S14C1	S13C1	S12C1	S11C1	S10C1	S09C1	S08C1	S07C1	S06C1	S05C1	S04C1	S03C1	S02C1	S01C1	S00C1	0000
LCDDATA5	059A	S31C1(1)	S30C1	S29C1(1)	S28C1(1)	S27C1	S26C1	S25C1	S24C1	S23C1	S22C1	S21C1	S20C1	S19C1	S18C1	S17C1	S16C1	0000
LCDDATA6	059C	S47C1	S46C1(1)	S45C1(1)	S44C1(1)	S43C1(1)	S42C1(1)	S41C1(1)	S40C1(1)	S39C1(1)	S38C1(1)	S37C1(1)	S36C1(1)	S35C1(1)	S34C1(1)	S33C1(1)	S32C1(1)	0000
LCDDATA7	059E	—	S62C1	S61C1(1)	S60C1(1)	S59C1(1)	S58C1(1)	S57C1(1)	S56C1(1)	S55C1(1)	S54C1(1)	S53C1(1)	S52C1(1)	S51C1(1)	S50C1(1)	S49C1(1)	S48C1	0000
LCDDATA8	05A0	S15C2	S14C2	S13C2	S12C2	S11C2	S10C2	S09C2	S08C2	S07C2	S06C2	S05C2	S04C2	S03C2	S02C2	S01C2	S00C2	0000
LCDDATA9	05A2	S31C2(1)	S30C2	S29C2(1)	S28C2(1)	S27C2	S26C2	S25C2	S24C2	S23C2	S22C2	S21C2	S20C2	S19C2	S18C2	S17C2	S16C2	0000
LCDDATA10	05A4	S47C2	S46C2(1)	S45C2(1)	S44C2(1)	S43C2(1)	S42C2(1)	S41C2(1)	S40C2(1)	S39C2(1)	S38C2(1)	S37C2(1)	S36C2(1)	S35C2(1)	S34C2(1)	S33C2(1)	S32C2(1)	0000
LCDDATA11	05A6	—	S62C2	S61C2(1)	S60C2(1)	S59C2(1)	S58C2(1)	S57C2(1)	S56C2(1)	S55C2(1)	S54C2(1)	S53C2(1)	S52C2(1)	S51C2(1)	S50C2(1)	S49C2(1)	S48C2	0000
LCDDATA12	05A8	S15C3	S14C3	S13C3	S12C3	S11C3	S10C3	S09C3	S08C3	S07C3	S06C3	S05C3	S04C3	S03C3	S02C3	S01C3	S00C3	0000

Legend: — = unimplemented, read as '0'. Reset values are shown in hexadecimal.
Note 1: These bits are unimplemented in 64-pin devices, read as '0'.

TABLE 4-28: LCD CONTROLLER REGISTER MAP (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
LCDDATA13	05AA	S31C3(1)	S30C3	S29C3(1)	S28C3(1)	S27C3	S26C3	S25C3	S24C3	S23C3	S22C3	S21C3	S20C3	S19C3	S18C3	S17C3	S16C3	0000
LCDDATA14	05AC	S47C3	S46C3(1)	S45C3(1)	S44C3(1)	S43C3(1)	S42C3(1)	S41C3(1)	S40C3(1)	S39C3(1)	S38C3(1)	S37C3(1)	S36C3(1)	S35C3(1)	S34C3(1)	S33C3(1)	S32C3(1)	0000
LCDDATA15	05AE	—	S62C3	S61C3(1)	S60C3(1)	S59C3(1)	S58C3(1)	S57C3(1)	S56C3(1)	S55C3(1)	S54C3(1)	S53C3(1)	S52C3(1)	S51C3(1)	S50C3(1)	S49C3(1)	S48C3	0000
LCDDATA16	05B0	S15C4	S14C4	S13C4	S12C4	S11C4	S10C4	S09C4	S08C4	S07C4	S06C4	S05C4	S04C4	S03C4	S02C4	S01C4	S00C4	0000
LCDDATA17	05B2	S31C4(1)	S30C4	S29C4(1)	S28C4(1)	S27C4	S26C4	S25C4	S24C4	S23C4	S22C4	S21C4	S20C4	S19C4	S18C4	S17C4	S16C4	0000
LCDDATA18	05B4	S47C4	S46C4(1)	S45C4(1)	S44C4(1)	S43C4(1)	S42C4(1)	S41C4(1)	S40C4(1)	S39C4(1)	S38C4(1)	S37C4(1)	S36C4(1)	S35C4(1)	S34C4(1)	S33C4(1)	S32C4(1)	0000
LCDDATA19	05B6	—	S62C4	S61C4(1)	S60C4(1)	S59C4(1)	S58C4(1)	S57C4(1)	S56C4(1)	S55C4(1)	S54C4(1)	S53C4(1)	S52C4(1)	S51C4(1)	S50C4(1)	S49C4(1)	S48C4	0000
LCDDATA20	05B8	S15C5	S14C5	S13C5	S12C5	S11C5	S10C5	S09C5	S08C5	S07C5	S06C5	S05C5	S04C5	S03C5	S02C5	S01C5	S00C5	0000
LCDDATA21	05BA	S31C5(1)	S30C5	S29C5(1)	S28C5(1)	S27C5	S26C5	S25C5	S24C5	S23C5	S22C5	S21C5	S20C5	S19C5	S18C5	S17C5	S16C5	0000
LCDDATA22	05BC	S47C5	S46C5(1)	S45C5(1)	S44C5(1)	S43C5(1)	S42C5(1)	S41C5(1)	S40C5(1)	S39C5(1)	S38C5(1)	S37C5(1)	S36C5(1)	S35C5(1)	S34C5(1)	S33C5(1)	S32C5(1)	0000
LCDDATA23	05BE	—	S62C5	S61C5(1)	S60C5(1)	S59C5(1)	S58C5(1)	S57C5(1)	S56C5(1)	S55C5(1)	S54C5(1)	S53C5(1)	S52C5(1)	S51C5(1)	S50C5(1)	S49C5(1)	S48C5	0000
LCDDATA24	05C0	S15C6	S14C6	S13C6	S12C6	S11C6	S10C6	S09C6	S08C6	S07C6	S06C6	S05C6	S04C6	S03C6	S02C6	S01C6	S00C6	0000
LCDDATA25	05C2	S31C6(1)	S30C6	S29C6(1)	S28C6(1)	S27C6	S26C6	S25C6	S24C6	S23C6	S22C6	S21C6	S20C6	S19C6	S18C6	S17C6	S16C6	0000
LCDDATA26	05C4	S47C6	S46C6(1)	S45C6(1)	S44C6(1)	S43C6(1)	S42C6(1)	S41C6(1)	S40C6(1)	S39C6(1)	S38C6(1)	S37C6(1)	S36C6(1)	S35C6(1)	S34C6(1)	S33C6(1)	S32C6(1)	0000
LCDDATA27	05C6	—	S62C6	S61C6(1)	S60C6(1)	S59C6(1)	S58C6(1)	S57C6(1)	S56C6(1)	S55C6(1)	S54C6(1)	S53C6(1)	S52C6(1)	S51C6(1)	S50C6(1)	S49C6(1)	S48C6	0000
LCDDATA28	05C8	S15C7	S14C7	S13C7	S12C7	S11C7	S10C7	S09C7	S08C7	S07C7	S06C7	S05C7	S04C7	S03C7	S02C7	S01C7	S00C7	0000
LCDDATA29	05CA	S31C7(1)	S30C7	S29C7(1)	S28C7(1)	S27C7	S26C7	S25C7	S24C7	S23C7	S22C7	S21C7	S20C7	S19C7	S18C7	S17C7	S16C7	0000
LCDDATA30	05CC	S47C7	S46C7(1)	S45C7(1)	S44C7(1)	S43C7(1)	S42C7(1)	S41C7(1)	S40C7(1)	S39C7(1)	S38C7(1)	S37C7(1)	S36C7(1)	S35C7(1)	S34C7(1)	S33C7(1)	S32C7(1)	0000
LCDDATA31	05CE	—	S62C7	S61C7(1)	S60C7(1)	S59C7(1)	S58C7(1)	S57C7(1)	S56C7(1)	S55C7(1)	S54C7(1)	S53C7(1)	S52C7(1)	S51C7(1)	S50C7(1)	S49C7(1)	S48C7	0000

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

Note 1: These bits are unimplemented in 64-pin devices, read as '0'.

TABLE 4-29: PARALLEL MASTER/SLAVE PORT 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
PMCON1	0600	PMPEN	—	PSIDL	ADRMUX1	ADRMUX0	—	MODE1	MODE0	CSF1	CSF0	ALP	ALMODE	—	BUSKEEP	IRQM1	IRQM0	0000
PMCON2	0602	BUSY	—	ERROR	TIMEOUT	—	—	—	—	RADDR23	RADDR22	RADDR21	RADDR20	RADDR19	RADDR18	RADDR17	RADDR16	0000
PMCON3	0604	PTWREN	PTRDEN	PTBE1EN	PTBE0EN	—	AWAITM1	AWAITM0	AWAITE	—	PTEN22	PTEN21	PTEN20	PTEN19	PTEN18	PTEN17	PTEN16	0000
PMCON4	0606	PTEN15	PTEN14	PTEN13	PTEN12	PTEN11	PTEN10	PTEN9	PTEN8	PTEN7	PTEN6	PTEN5	PTEN4	PTEN3	PTEN2	PTEN1	PTEN0	0000
PMCS1CF	0608	CSDIS	CSP	CSPTEN	BEP	—	WRSP	RDSP	SM	ACKP	PTSZ1	PTSZ0	—	—	—	—	—	0000
PMCS1BS	060A	BASE23	BASE22	BASE21	BASE20	BASE19	BASE18	BASE17	BASE16	BASE15	—	—	—	—	—	—	—	0200
PMCS1MD	060C	ACKM1	ACKM0	AMWAIT2	AMWAIT1	AMWAIT0	—	—	—	DWAITB1	DWAITB0	DWAITM3	DWAITM2	DWAITM1	DWAITM0	DWAITE1	DWAITE0	0000
PMCS2CF	060E	CSDIS	CSP	CSPTEN	BEP	—	WRSP	RDSP	SM	ACKP	PTSZ1	PTSZ0	—	—	—	—	—	0000
PMCS2BS	0610	BASE23	BASE22	BASE21	BASE20	BASE19	BASE18	BASE17	BASE16	BASE15	—	—	—	—	—	—	—	0600
PMCS2MD	0612	ACKM1	ACKM0	AMWAIT2	AMWAIT1	AMWAIT0	—	—	—	DWAITB1	DWAITB0	DWAITM3	DWAITM2	DWAITM1	DWAITM0	DWAITE1	DWAITE0	0000
PMDOUT1	0614	Data Out Register 1<15:8>								Data Out Register 1<7:0>								xxxx
PMDOUT2	0616	Data Out Register 2<15:8>								Data Out Register 2<7:0>								xxxx
PMDIN1	0618	Data In Register 1<15:8>								Data In Register 1<7:0>								xxxx
PMDIN2	061A	Data In Register 2<15:8>								Data In Register 2<7:0>								xxxx
PMSTAT	061C	IBF	IBOV	—	—	IB3F	IB2F	IB1F	IB0F	OBE	OBUF	—	—	OB3E	OB2E	OB1E	OB0E	008F

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

TABLE 4-30: REAL-TIME CLOCK AND CALENDAR (RTCC) 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
ALRMVAL	0620	Alarm Value Register Window Based on ALRMPTR<1:0>																xxxx
ALCFGRPT	0622	ALRMEN	CHIME	AMASK3	AMASK2	AMASK1	AMASK0	ALRMPTR1	ALRMPTR0	ARPT7	ARPT6	ARPT5	ARPT4	ARPT3	ARPT2	ARPT1	ARPT0	0000
RTCVAL	0624	RTCC Value Register Window Based on RTCPTR<1:0>																xxxx
RCFGCAL	0626	RTCEN	—	RTCWREN	RTCSYNC	HALFSEC	RTCOE	RTCPTR1	RTCPTR0	CAL7	CAL6	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	Note 1
RTCPWC	0628	PWCEN	PWCPOL	PWCPRE	PWSPRE	RTCLK1	RTCLK0	RTCOUT1	RTCOUT0	—	—	—	—	—	—	—	—	Note 1

Legend: — = unimplemented, read as '0'. Reset values are shown in hexadecimal.
Note 1: The status of the RCFGCAL and RTCPWR registers on POR is '0000', and on other Resets, it is unchanged

TABLE 4-31: DATA SIGNAL MODULATOR (DSM) 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
MDCON	062A	MDEN	—	MDSIDL	—	—	—	—	—	—	MDOE	MDSLR	MDOPOL	—	—	—	MDBIT	0020
MDSRC	062C	—	—	—	—	—	—	—	—	SODIS	—	—	—	MS3	MS2	MS1	MS0	000x
MDCAR	062E	CHODIS	CHPOL	CHSYNC	—	CH3	CH2	CH1	CH0	CLODIS	CLPOL	CLSYNC	—	CL3	CL2	CL1	CL0	0000

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

TABLE 4-32: 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	CVRSS	CVR4	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 4-33: 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 4-34: BAND GAP BUFFER INTERFACE 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
BUFCON0	0670	BUFEN	—	BUFSIDL	BUFSLP	—	—	—	—	—	BUFSTBY	—	—	—	—	BUFREF1	BUFREF0	0000
BUFCON1	0672	BUFEN	—	BUFSIDL	BUFSLP	—	—	—	—	BUFOE	BUFSTBY	—	—	—	—	BUFREF1	BUFREF0	0000
BUFCON2	0674	BUFEN	—	BUFSIDL	BUFSLP	—	—	—	—	BUFOE	BUFSTBY	—	—	—	—	BUFREF1	BUFREF0	0000

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

TABLE 4-35: PERIPHERAL PIN SELECT 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
RPINR0	0680	—	—	INT1R5	INT1R4	INT1R3	INT1R2	INT1R1	INT1R0	—	—	—	—	—	—	—	—	003F
RPINR1	0682	—	—	INT3R5	INT3R4	INT3R3	INT3R2	INT3R1	INT3R0	—	—	INT2R5	INT2R4	INT2R3	INT2R2	INT2R1	INT2R0	3F00
RPINR2	0684	—	—	—	—	—	—	—	—	—	—	INT4R5	INT4R4	INT4R3	INT4R2	INT4R1	INT4R0	3F3F
RPINR7	068E	—	—	IC2R5	IC2R4	IC2R3	IC2R2	IC2R1	IC2R0	—	—	IC1R5	IC1R4	IC1R3	IC1R2	IC1R1	IC1R0	3F3F
RPINR8	0690	—	—	IC4R5	IC4R4	IC4R3	IC4R2	IC4R1	IC4R0	—	—	IC3R5	IC3R4	IC3R3	IC3R2	IC3R1	IC3R0	3F3F
RPINR9	0692	—	—	IC6R5	IC6R4	IC6R3	IC6R2	IC6R1	IC6R0	—	—	IC5R5	IC5R4	IC5R3	IC5R2	IC5R1	IC5R0	3F3F
RPINR10	0694	—	—	IC8R5	IC8R4	IC8R3	IC8R2	IC8R1	IC8R0	—	—	IC7R5	IC7R4	IC7R3	IC7R2	IC7R1	IC7R0	003F
RPINR11	0696	—	—	OCFBR5	OCFBR4	OCFBR3	OCFBR2	OCFBR1	OCFBR0	—	—	OCFAR5	OCFAR4	OCFAR3	OCFAR2	OCFAR1	OCFAR0	3F3F
RPINR15	06A2	—	—	IC9R5	IC9R4	IC9R3	IC9R2	IC9R1	IC9R0	—	—	—	—	—	—	—	—	3F00
RPINR17	06A2	—	—	U3RXR5	U3RXR4	U3RXR3	U3RXR2	U3RXR1	U3RXR0	—	—	—	—	—	—	—	—	3F00
RPINR18	06A4	—	—	U1CTSR5	U1CTSR4	U1CTSR3	U1CTSR2	U1CTSR1	U1CTSR0	—	—	U1RXR5	U1RXR4	U1RXR3	U1RXR2	U1RXR1	U1RXR0	3F3F
RPINR19	06A6	—	—	U2CTSR5	U2CTSR4	U2CTSR3	U2CTSR2	U2CTSR1	U2CTSR0	—	—	U2RXR5	U2RXR4	U2RXR3	U2RXR2	U2RXR1	U2RXR0	3F3F
RPINR20	06A8	—	—	SCK1R5	SCK1R4	SCK1R3	SCK1R2	SCK1R1	SCK1R0	—	—	SDI1R5	SDI1R4	SDI1R3	SDI1R2	SDI1R1	SDI1R0	3F3F
RPINR21	06AA	—	—	U3CTSR5	U3CTSR4	U3CTSR3	U3CTSR2	U3CTSR1	U3CTSR0	—	—	SS1R5	SS1R4	SS1R3	SS1R2	SS1R1	SS1R0	3F3F
RPINR22	06AC	—	—	SCK2R5	SCK2R4	SCK2R3	SCK2R2	SCK2R1	SCK2R0	—	—	SDI2R5	SDI2R4	SDI2R3	SDI2R2	SDI2R1	SDI2R0	3F3F
RPINR23	06AE	—	—	TMRCKR5	TMRCKR4	TMRCKR3	TMRCKR2	TMRCKR1	TMRCKR0	—	—	SS2R5	SS2R4	SS2R3	SS2R2	SS2R1	SS2R0	003F
RPINR27	06B6	—	—	U4CTSR5	U4CTSR4	U4CTSR3	U4CTSR2	U4CTSR1	U4CTSR0	—	—	U4RXR5	U4RXR4	U4RXR3	U4RXR2	U4RXR1	U4RXR0	3F3F
RPINR30	06BC	—	—	—	—	—	—	—	—	—	—	MDMIR5	MDMIR4	MDMIR3	MDMIR2	MDMIR1	MDMIR0	003F
RPINR31	06BE	—	—	MDC2R5	MDC2R4	MDC2R3	MDC2R2	MDC2R1	MDC2R0	—	—	MDC1R5	MDC1R4	MDC1R3	MDC1R2	MDC1R1	MDC1R0	3F3F

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

Note 1: These bits are unimplemented in 64-pin devices, read as '0'.

TABLE 4-35: PERIPHERAL PIN SELECT REGISTER MAP (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
RPOR0	06C0	—	—	RP1R5	RP1R4	RP1R3	RP1R2	RP1R1	RP1R0	—	—	RP0R5	RP0R4	RP0R3	RP0R2	RP0R1	RP0R0	0000
RPOR1	06C2	—	—	RP3R5	RP3R4	RP3R3	RP3R2	RP3R1	RP3R0	—	—	RP2R5	RP2R4	RP2R3	RP2R2	RP2R1	RP2R0	0000
RPOR2	06C4	—	—	RP5R5(1)	RP5R4(1)	RP5R3(1)	RP5R2(1)	RP5R1(1)	RP5R0(1)	—	—	RP4R5	RP4R4	RP4R3	RP4R2	RP4R1	RP4R0	0000
RPOR3	06C6	—	—	RP7R5	RP7R4	RP7R3	RP7R2	RP7R1	RP7R0	—	—	RP6R5	RP6R4	RP6R3	RP6R2	RP6R1	RP6R0	0000
RPOR4	06C8	—	—	RP9R5	RP9R4	RP9R3	RP9R2	RP9R1	RP9R0	—	—	RP8R5	RP8R4	RP8R3	RP8R2	RP8R1	RP8R0	0000
RPOR5	06CA	—	—	RP11R5	RP11R4	RP11R3	RP11R2	RP11R1	RP11R0	—	—	RP10R5	RP10R4	RP10R3	RP10R2	RP10R1	RP10R0	0000
RPOR6	06CC	—	—	RP13R5	RP13R4	RP13R3	RP13R2	RP13R1	RP13R0	—	—	RP12R5	RP12R4	RP12R3	RP12R2	RP12R1	RP12R0	0000
RPOR7	06CE	—	—	RP15R5(1)	RP15R4(1)	RP15R3(1)	RP15R2(1)	RP15R1(1)	RP15R0(1)	—	—	RP14R5	RP14R4	RP14R3	RP14R2	RP14R1	RP14R0	0000
RPOR8	06D0	—	—	RP17R5	RP17R4	RP17R3	RP17R2	RP17R1	RP17R0	—	—	RP16R5	RP16R4	RP16R3	RP16R2	RP16R1	RP16R0	0000
RPOR9	06D2	—	—	RP19R5	RP19R4	RP19R3	RP19R2	RP19R1	RP19R0	—	—	RP18R5	RP18R4	RP18R3	RP18R2	RP18R1	RP18R0	0000
RPOR10	06D4	—	—	RP21R5	RP21R4	RP21R3	RP21R2	RP21R1	RP21R0	—	—	RP20R5	RP20R4	RP20R3	RP20R2	RP20R1	RP20R0	0000
RPOR11	06D6	—	—	RP23R5	RP23R4	RP23R3	RP23R2	RP23R1	RP23R0	—	—	RP22R5	RP22R4	RP22R3	RP22R2	RP22R1	RP22R0	0000
RPOR12	06D8	—	—	RP25R5	RP25R4	RP25R3	RP25R2	RP25R1	RP25R0	—	—	RP24R5	RP24R4	RP24R3	RP24R2	RP24R1	RP24R0	0000
RPOR13	06DA	—	—	RP27R5	RP27R4	RP27R3	RP27R2	RP27R1	RP27R0	—	—	RP26R5	RP26R4	RP26R3	RP26R2	RP26R1	RP26R0	0000
RPOR14	06DC	—	—	RP29R5	RP29R4	RP29R3	RP29R2	RP29R1	RP29R0	—	—	RP28R5	RP28R4	RP28R3	RP28R2	RP28R1	RP28R0	0000
RPOR15	06DE	—	—	RP31R5(1)	RP31R4(1)	RP31R3(1)	RP31R2(1)	RP31R1(1)	RP31R0(1)	—	—	RP30R5	RP30R4	RP30R3	RP30R2	RP30R1	RP30R0	0000

Legend: — = unimplemented, read as '0'. Reset values are shown in hexadecimal.
Note 1: These bits are unimplemented in 64-pin devices, read as '0'.

TABLE 4-36: SYSTEM CONTROL (CLOCK AND RESET) REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
RCON	0740	TRAPR	IOPUWR	—	RETEN	—	DPSLP	CM	PMSLP	EXTR	SWR	SWDTEN	WDTO	SLEEP	IDLE	BOR	POR	Note 1
OSCCON	0742	—	COSC2	COSC1	COSC0	—	NOSC2	NOSC1	NOSC0	CLKLOCK	IOLOCK	LOCK	—	CF	POSCEN	SOSCEN	OSWEN	Note 2
CLKDIV	0744	ROI	DOZE2	DOZE1	DOZE0	DOZEN	RCDIV2	RCDIV1	RCDIV0	CPDIV1	CPDIV0	PLLEN	—	—	—	—	—	3100
OSCTUN	0748	STEN	—	STSIDL	STSRC	STLOCK	STLPOL	STOR	STORPOL	—	—	TUN5	TUN4	TUN3	TUN2	TUN1	TUN0	0000
REFOCON	074E	ROEN	—	ROSSLP	ROSEL	RODIV3	RODIV2	RODIV1	RODIV0	—	—	—	—	—	—	—	—	0000
HLVDCON	0756	HLVDEN	—	LSIDL	—	—	—	—	—	VDIR	BGVST	IRVST	—	HLVDL3	HLVDL2	HLVDL1	HLVDL0	0000
RCON2	0762	—	—	—	—	—	—	—	—	—	—	—	r	VDDBOR	VDDPOR	VBPOR	VBAT	Note 1

Legend: — = unimplemented, read as '0'; r = reserved. Reset values are shown in hexadecimal.
Note 1: The Reset value of the RCON register is dependent on the type of Reset event. See **Section 7.0 "Resets"** for more information.
2: The Reset value of the OSCCON register is dependent on both the type of Reset event and the device configuration. See **Section 9.0 "Oscillator Configuration"** for more information.

TABLE 4-37: DEEP SLEEP 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
DSCON	0758	DSEN	—	—	—	—	—	—	—	—	—	—	—	—	r	DSBOR	RELEASE	0000(1)
DSWAKE	075A	—	—	—	—	—	—	—	DSINT0	DSFLT	—	—	DSWDT	DSRTCC	DSMCLR	—	—	0000(1)
DSGPR0	075C	Deep Sleep Semaphore Data 0																0000(1)
DSGPR1	075E	Deep Sleep Semaphore Data 1																0000(1)

Legend: — = unimplemented, read as '0'; r = reserved. Reset values are shown in hexadecimal.
Note 1: These registers are only reset on a VDD POR event.

TABLE 4-38: NVM REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
NVMCON	0760	WR	WREN	WRERR	—	—	—	—	—	—	ERASE	—	—	NVMOP3	NVMOP2	NVMOP1	NVMOP0	0000(1)
NVMKEY	0766	—	—	—	—	—	—	—	—	NVMKEY Register<7:0>								0000

Legend: — = unimplemented, read as '0'. Reset values are shown in hexadecimal.
Note 1: The Reset value shown is for POR only. The value on other Reset states is dependent on the state of memory write or erase operations at the time of Reset.

TABLE 4-39: PMD 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
PMD1	0770	T5MD	T4MD	T3MD	T2MD	T1MD	—	—	—	I2C1MD	U2MD	U1MD	SPI2MD	SPI1MD	—	—	ADC1MD	0000
PMD2	0772	IC8MD	IC7MD	IC6MD	IC5MD	IC4MD	IC3MD	IC2MD	IC1MD	OC8MD	OC7MD	OC6MD	OC5MD	OC4MD	OC3MD	OC2MD	OC1MD	0000
PMD3	0774	—	—	—	—	TXMMD	CMPMD	RTCCMD	PMPMD	CRCMD	DAC1MD	—	—	U3MD	—	I2C2MD	—	0000
PMD4	0776	—	—	—	—	—	—	—	—	—	UPWMMD	U4MD	—	REFOMD	CTMUMD	HLVDMD	USB1MD	0000
PMD5	0778	—	—	—	—	—	—	—	IC9MD	—	—	—	—	—	—	—	OC9MD	0000
PMD6	077A	—	—	—	—	—	—	—	—	—	LCDMD	AMP1MD	DAC2MD	AMP2MD	SDA1MD	—	—	0000
PMD7	077C	—	—	—	—	—	—	—	—	—	—	DMA1MD	DMA0MD	—	—	—	—	0000

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

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 the Enhanced Parallel Master Port (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 4-4 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 register (DSRPAG) or Data Space Write 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).

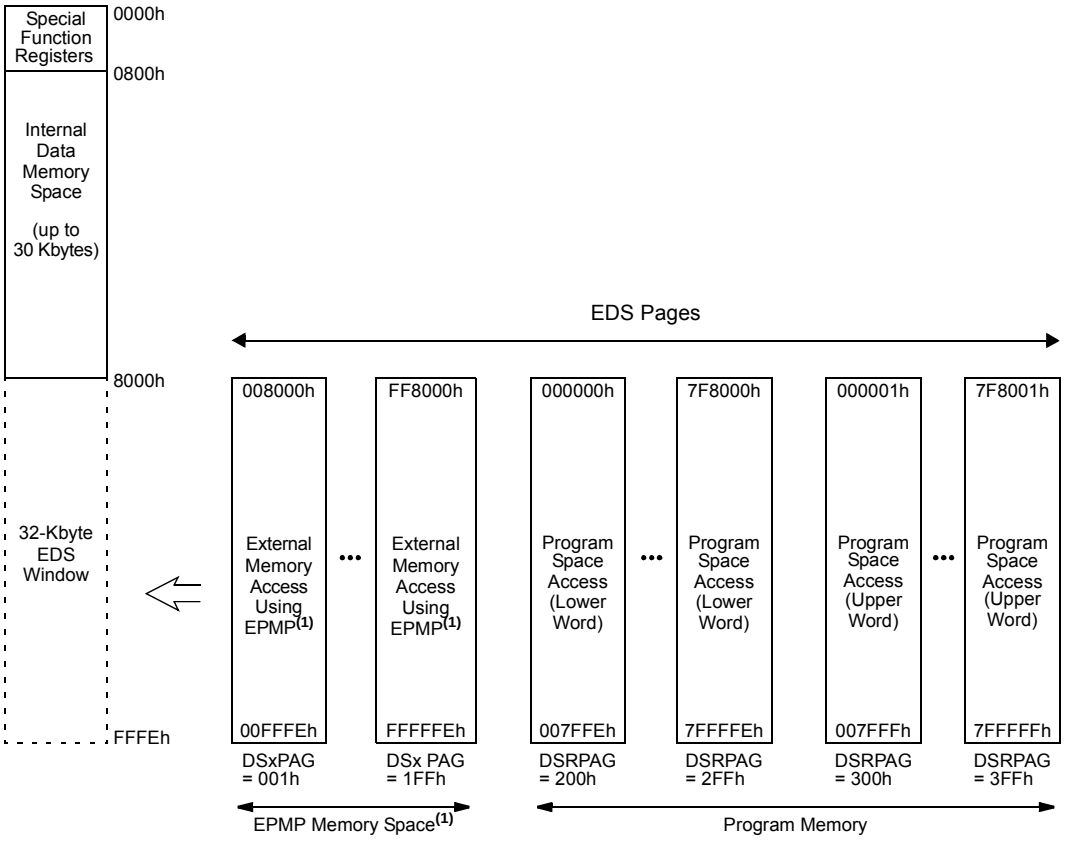
The data addressing range of PIC24FJ128GC010 family devices depends on the version of the Enhanced Parallel Master Port (EPMP) implemented on a particular device; this is, in turn, is a function of the device pin count. Table 4-40 lists the total memory accessible by each of the devices in this family. For more details on accessing external memory using EPMP, refer to the "*PIC24F Family Reference Manual*", **Section 42. "Enhanced Parallel Master Port (EPMP)"** (DS39730).

TABLE 4-40: TOTAL ACCESSIBLE DATA MEMORY

Family	Internal RAM	External RAM Access Using EPMP
PIC24FJXXXGC010	8K	Up to 16 Mbytes
PIC24FJXXXGC006	8K	Up to 64K

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 4-4: EXTENDED DATA SPACE



Note 1: The range of addressable memory available is dependent on the device pin count and EPMP implementation.

4.2.5.1 Data Read from EDS

In order to read the data from the EDS space, first, an Address Pointer is set up by loading the required EDS page number into the DSRPAG register and assigning the offset address to one of the W registers. Once the above assignment is done, the EDS window is enabled by setting bit 15 of the working register, assigned with the offset address; then, the contents of the pointed EDS location can be read.

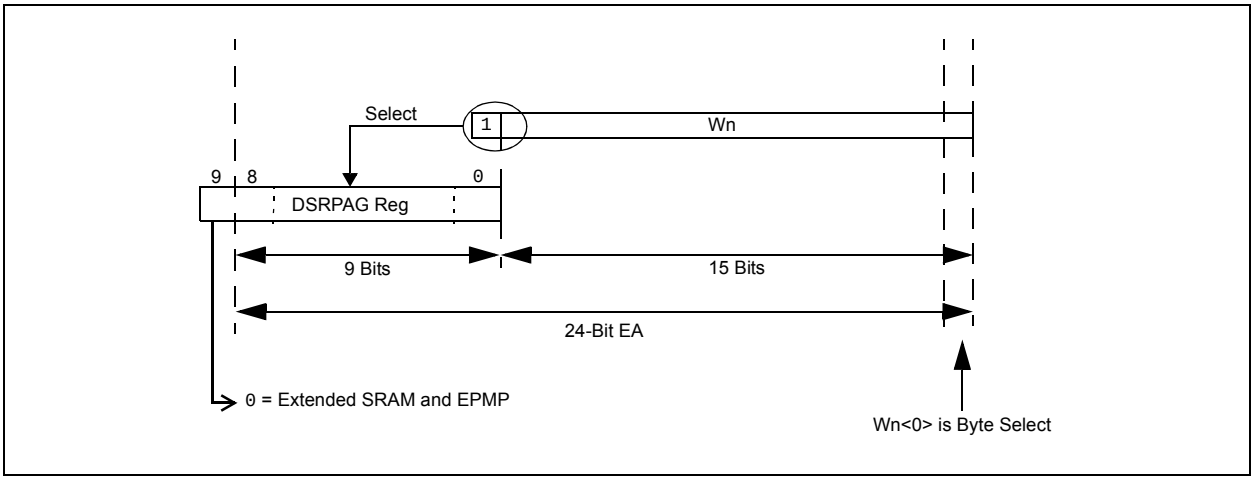
Figure 4-5 illustrates how the EDS space address is generated for read operations.

When the Most Significant bit (MSb) of EA is '1' and DSRPAG<9> = 0, the lower 9 bits of DSRPAG are concatenated to the lower 15 bits of the EA to form a 24-bit EDS space address for read operations.

Example 4-1 shows how to read a byte, word and double-word from EDS.

Note:	All read operations from EDS space have an overhead of one instruction cycle. Therefore, a minimum of two instruction cycles is required to complete an EDS read. EDS reads under the REPEAT instruction; the first two accesses take three cycles and the subsequent accesses take one cycle.

FIGURE 4-5: EDS ADDRESS GENERATION FOR READ OPERATIONS



EXAMPLE 4-1: EDS READ CODE IN ASSEMBLY

```
; Set the EDS page from where the data to be read
    mov         #0x0002, w0
    mov         w0, DSRPAG      ;page 2 is selected for read
    mov         #0x0800, w1     ;select the location (0x800) to be read
    bset        w1, #15         ;set the MSB of the base address, enable EDS mode

;Read a byte from the selected location
    mov.b       [w1++], w2      ;read Low byte
    mov.b       [w1++], w3      ;read High byte

;Read a word from the selected location
    mov         [w1], w2        ;

;Read Double - word from the selected location
    mov.d       [w1], w2        ;two word read, stored in w2 and w3
```

4.2.5.2 Data Write into EDS

In order to write data to EDS space, such as in EDS reads, an Address Pointer is set up by loading the required EDS page number into the DSWPAG register and assigning the offset address to one of the W registers. Once the above assignment is done, then the EDS window is enabled by setting bit 15 of the working register, assigned with the offset address, and the accessed location can be written.

Figure 4-6 illustrates how the EDS space address is generated for write operations.

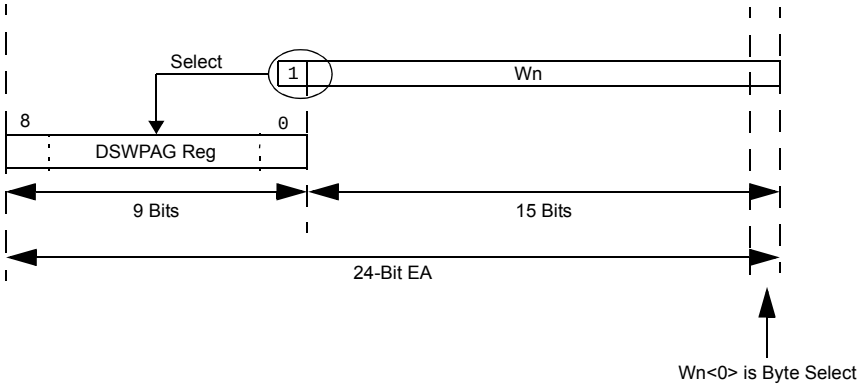
When the MSb of EA is '1', the lower 9 bits of DSWPAG are concatenated to the lower 15 bits of EA to form a 24-bit EDS address for write operations. Example 4-2 shows how to write a byte, word and double-word to EDS.

The Page registers (DSRPAG/DSWPAG) do not update automatically while crossing a page boundary when the rollover happens from 0xFFFF to 0x8000.

While developing code in assembly, care must be taken to update the Page registers when an Address Pointer crosses the page boundary. The 'C' compiler keeps track of the addressing, and increments or decrements the Page registers accordingly, while accessing contiguous data memory locations.

Note 1:	All write operations to EDS are executed in a single cycle.
2:	Use of Read/Modify/Write operation on any EDS location under a REPEAT instruction is not supported. For example, `BCLR, BSW, BTG, RLC f, RLNC f, RRC f, RRNC f, ADD f, SUB f, SUBR f, AND f, IOR f, XOR f, ASR f, ASL f`.
3:	Use the DSRPAG register while performing Read/Modify/Write operations.

FIGURE 4-6: EDS ADDRESS GENERATION FOR WRITE OPERATIONS



EXAMPLE 4-2: EDS WRITE CODE IN ASSEMBLY

```
; Set the EDS page where the data to be written
    mov     #0x0002, w0
    mov     w0, DSWPAG      ;page 2 is selected for write
    mov     #0x0800, w1     ;select the location (0x800) to be written
    bset    w1, #15         ;set the MSB of the base address, enable EDS mode

;Write a byte to the selected location
    mov     #0x00A5, w2
    mov     #0x003C, w3
    mov.b   w2, [w1++]      ;write Low byte
    mov.b   w3, [w1++]      ;write High byte

;Write a word to the selected location
    mov     #0x1234, w2     ;
    mov     w2, [w1]        ;

;Write a Double - word to the selected location
    mov     #0x1122, w2
    mov     #0x4455, w3
    mov.d   w2, [w1]        ;2 EDS writes
```

TABLE 4-41: EDS MEMORY ADDRESS WITH DIFFERENT PAGES AND ADDRESSES

DSRPAG (Data Space Read Register)	DSWPAG (Data Space Write Register)	Source/Destination Address While Indirect Addressing	24-Bit EA Pointing to EDS	Comment
x(1)	x(1)	0000h to 1FFFh	000000h to 001FFFh	Near Data Space(2)
		2000h to 7FFFh	002000h to 007FFFh	
001h	001h	8000h to FFFFh	008000h to 00FFFEh	EPMP Memory Space
002h	002h		010000h to 017FFEh	
003h • • • • • 1FFh	003h • • • • • 1FFh		018000h to 0187FEh • • • • FF8000h to FFFFFEh	
000h	000h		Invalid Address	Address Error Trap(3)

Note 1: If the source/destination address is below 8000h, the DSRPAG and DSWPAG registers are not considered.

2: This Data Space can also be accessed by Direct Addressing.

3: When the source/destination address is above 8000h and DSRPAG/DSWPAG are '0', an address error trap will occur.

4.2.6 SOFTWARE STACK

Apart from its use as a working register, the W15 register in PIC24F devices is also used as a Software Stack Pointer (SSP). The pointer always points to the first available free word and grows from lower to higher addresses. It predecrements for stack pops and post-increments for stack pushes, as shown in Figure 4-7. Note that for a PC push during any CALL instruction, the MSB of the PC is zero-extended before the push, ensuring that the MSB is always clear.

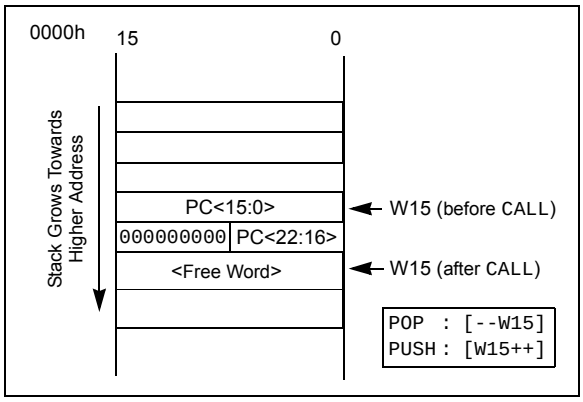
Note:	A PC push during exception processing will concatenate the SRL register to the MSB of the PC prior to the push.

The Stack Pointer Limit Value register (SPLIM), associated with the Stack Pointer, sets an upper address boundary for the stack. SPLIM is uninitialized at Reset. As is the case for the Stack Pointer, SPLIM<0> is forced to '0' as all stack operations must be word-aligned. Whenever an EA is generated using W15 as a source or destination pointer, the resulting address is compared with the value in SPLIM. If the contents of the Stack Pointer (W15) and the SPLIM register are equal, and a push operation is performed, a stack error trap will not occur. The stack error trap will occur on a subsequent push operation. Thus, for example, if it is desirable to cause a stack error trap when the stack grows beyond address, 2000h in RAM, initialize the SPLIM with the value, 1FFEh.

Similarly, a Stack Pointer underflow (stack error) trap is generated when the Stack Pointer address is found to be less than 0800h. This prevents the stack from interfering with the SFR space.

A write to the SPLIM register should not be immediately followed by an indirect read operation using W15.

FIGURE 4-7: CALL STACK FRAME



4.3 Interfacing Program and Data Memory Spaces

The PIC24F architecture uses a 24-bit wide program space and 16-bit wide Data Space. The architecture is also a modified Harvard scheme, meaning that data can also be present in the program space. To use this data successfully, it must be accessed in a way that preserves the alignment of information in both spaces.

Aside from normal execution, the PIC24F architecture provides two methods by which program space can be accessed during operation:

- Using table instructions to access individual bytes or words anywhere in the program space
- Remapping a portion of the program space into the Data Space (Program Space Visibility)

Table instructions allow an application to read or write to small areas of the program memory. This makes the method ideal for accessing data tables that need to be updated from time to time. It also allows access to all bytes of the program word. The remapping method allows an application to access a large block of data on a read-only basis, which is ideal for look-ups from a large table of static data. It can only access the least significant word of the program word.

4.3.1 ADDRESSING PROGRAM SPACE

Since the address ranges for the data and program spaces are 16 and 24 bits, respectively, a method is needed to create a 23-bit or 24-bit program address from 16-bit data registers. The solution depends on the interface method to be used.

For table operations, the 8-bit Table Memory Page Address register (TBLPAG) is used to define a 32K word region within the program space. This is concatenated with a 16-bit EA to arrive at a full 24-bit program space address. In this format, the MSBs of TBLPAG are used to determine if the operation occurs in the user memory (TBLPAG<7> = 0) or the configuration memory (TBLPAG<7> = 1).

For remapping operations, the 10-bit Extended Data Space Read register (DSRPAG) is used to define a 16K word page in the program space. When the Most Significant bit (MSb) of the EA is '1', and the MSb (bit 9) of DSRPAG is '1', the lower 8 bits of DSRPAG are concatenated with the lower 15 bits of the EA to form a 23-bit program space address. The DSRPAG<8> bit decides whether the lower word (when the bit is '0') or the higher word (when the bit is '1') of program memory is mapped. Unlike table operations, this strictly limits remapping operations to the user memory area.

Table 4-42 and Figure 4-8 show how the program EA is created for table operations, and remapping accesses from the data EA. Here, P<23:0> refer to a program space word, whereas D<15:0> refer to a Data Space word.

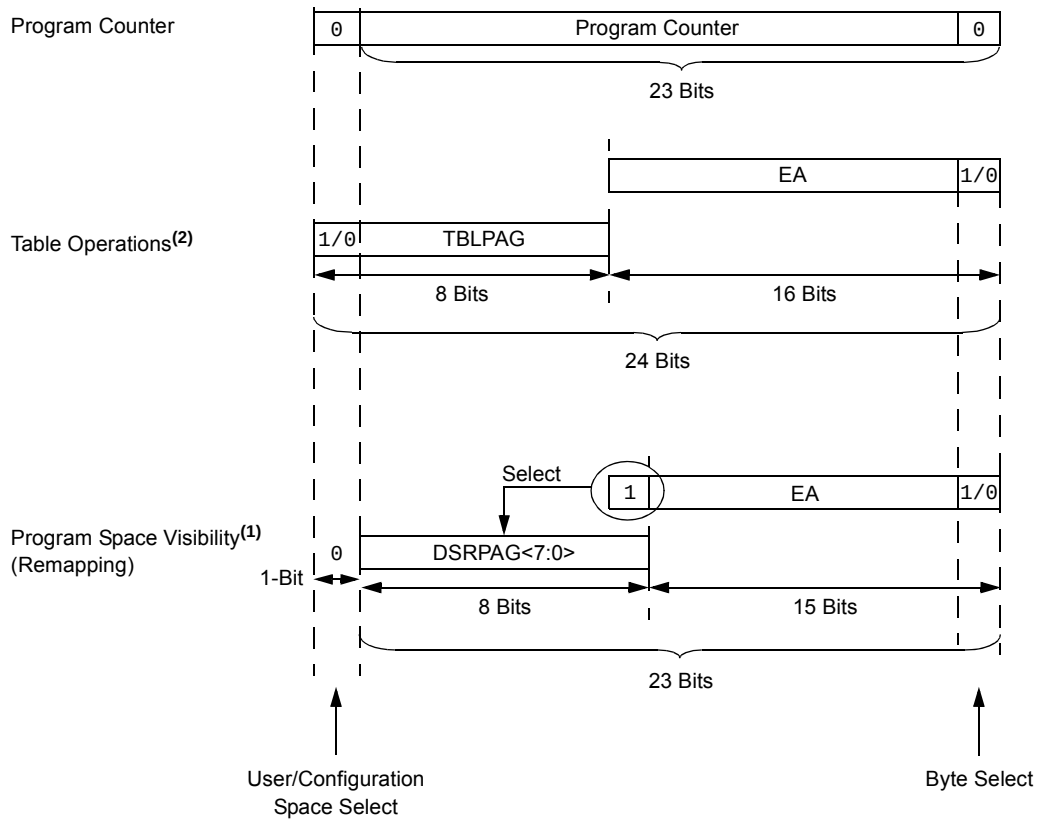
TABLE 4-42: PROGRAM SPACE ADDRESS CONSTRUCTION

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.

4.3.2 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'.

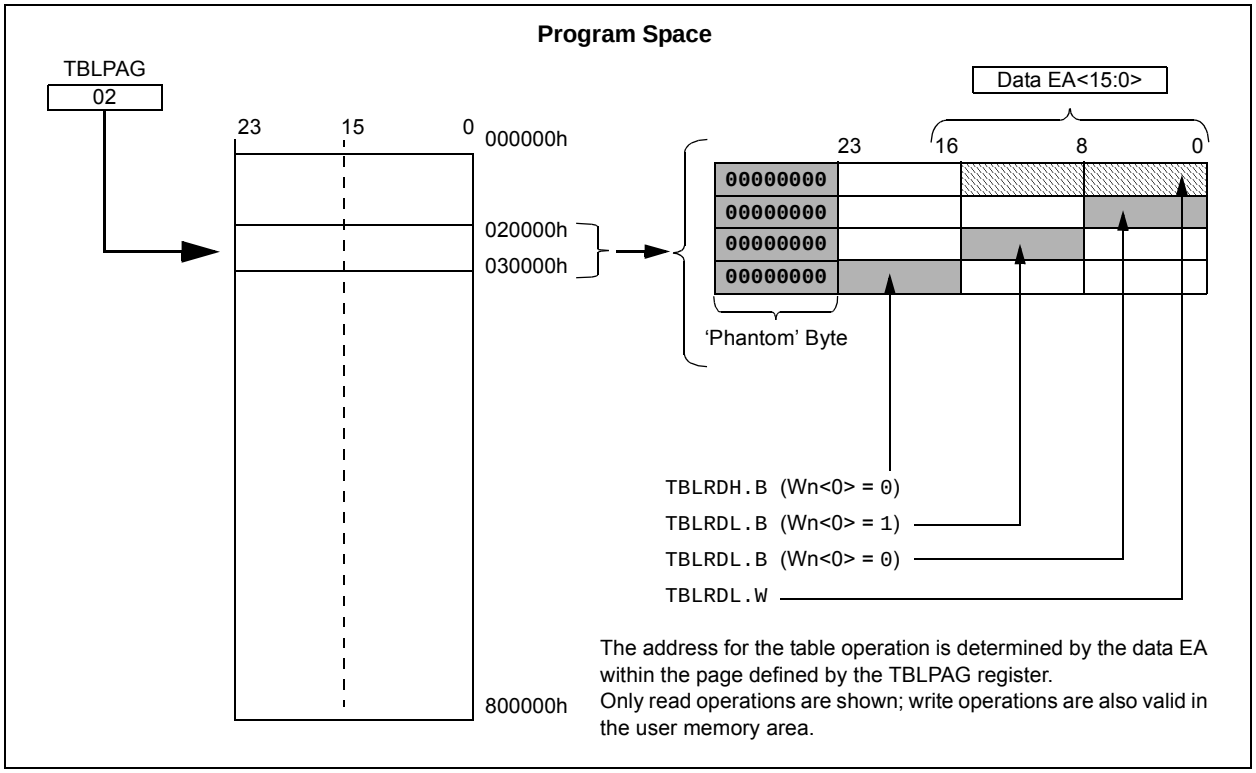
   In Byte mode, it maps the upper or lower byte of the program word to D<7:0> of the data address, as above. Note that the data will always be '0' when the upper 'phantom' byte is selected (Byte Select = 1).

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 described in **Section 6.0 "Flash Program Memory"**.

For all table operations, the area of program memory space to be accessed is determined by the Table Memory Page Address 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.

Note:	Only table read operations will execute in the configuration memory space where Device IDs are located. Table write operations are not allowed.

FIGURE 4-9: ACCESSING PROGRAM MEMORY WITH TABLE INSTRUCTIONS



4.3.3 READING DATA FROM PROGRAM MEMORY USING EDS

The upper 32 Kbytes of Data Space may optionally be mapped into any 16K word page of the program space. This provides transparent access of stored constant data from the Data Space without the need to use special instructions (i.e., TBLRDL/H).

Program space access through the Data Space occurs when the MSb of EA is '1' and the DSRPAG<9> is also '1'. The lower 8 bits of DSRPAG are concatenated to the Wn<14:0> bits to form a 23-bit EA to access program memory. The DSRPAG<8> decides which word should be addressed; when the bit is '0', the lower word and when '1', the upper word of the program memory is accessed.

The entire program memory is divided into 512 EDS pages, from 200h to 3FFh, each consisting of 16K words of data. Pages, 200h to 2FFh, correspond to the lower words of the program memory, while 300h to 3FFh correspond to the upper words of the program memory.

Using this EDS technique, the entire program memory can be accessed. Previously, the access to the upper word of the program memory was not supported.

Table 4-43 provides the corresponding 23-bit EDS address for program memory with EDS page and source addresses.

For operations that use PSV and are executed outside a REPEAT loop, the MOV and MOV.D instructions will require one instruction cycle in addition to the specified execution time. All other instructions will require two instruction cycles in addition to the specified execution time.

For operations that use PSV, which are executed inside a REPEAT loop, there will be some instances that require two instruction cycles in addition to the specified execution time of the instruction:

- Execution in the first iteration
- Execution in the last iteration
- Execution prior to exiting the loop due to an interrupt
- Execution upon re-entering the loop after an interrupt is serviced

Any other iteration of the REPEAT loop will allow the instruction accessing data, using PSV, to execute in a single cycle.

TABLE 4-43: EDS PROGRAM ADDRESS WITH DIFFERENT PAGES AND ADDRESSES

DSRPAG (Data Space Read Register)	Source Address While Indirect Addressing	23-Bit EA Pointing to EDS	Comment
200h • • • 2FFh	8000h to FFFFh	000000h to 007FFEh • • • 7F8000h to 7FFFFEh	Lower words of 4M program instructions; (8 Mbytes) for read operations only.
300h • • • 3FFh		000001h to 007FFFh • • • 7F8001h to 7FFFFFh	Upper words of 4M program instructions (4 Mbytes remaining, 4 Mbytes are phantom bytes); for read operations only.
000h		Invalid Address	Address error trap(1)

Note 1: When the source/destination address is above 8000h and DSRPAG/DSWPAG are '0', an address error trap will occur.

EXAMPLE 4-3: EDS READ CODE FROM PROGRAM MEMORY IN ASSEMBLY

```
; Set the EDS page from where the data to be read
    mov     #0x0202, w0
    mov     w0, DSRPAG                ;page 0x202, consisting lower words, is selected for read
    mov     #0x000A, w1               ;select the location (0x0A) to be read
    bset    w1, #15                   ;set the MSB of the base address, enable EDS mode
;Read a byte from the selected location
    mov.b   [w1++], w2                ;read Low byte
    mov.b   [w1++], w3                ;read High byte
;Read a word from the selected location
    mov     [w1], w2                  ;
;Read Double - word from the selected location
    mov.d   [w1], w2                  ;two word read, stored in w2 and w3
```

FIGURE 4-10: PROGRAM SPACE VISIBILITY OPERATION TO ACCESS LOWER WORD

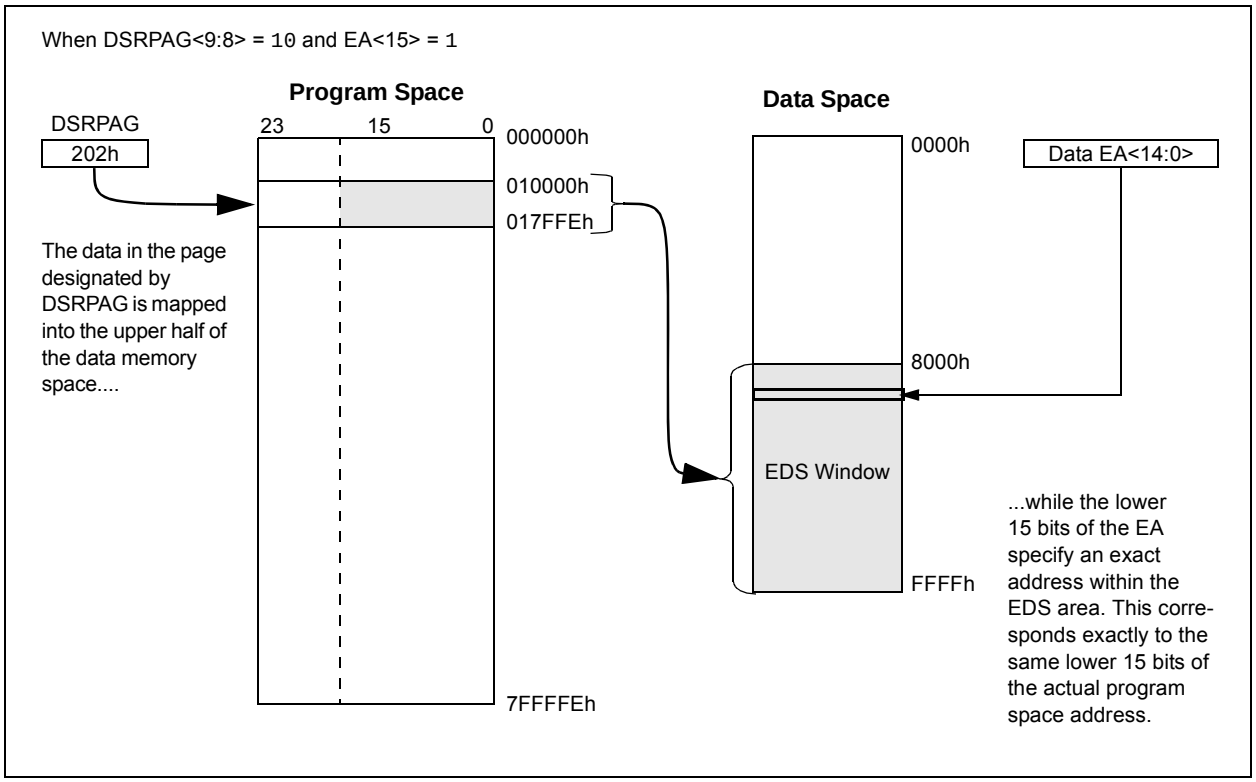
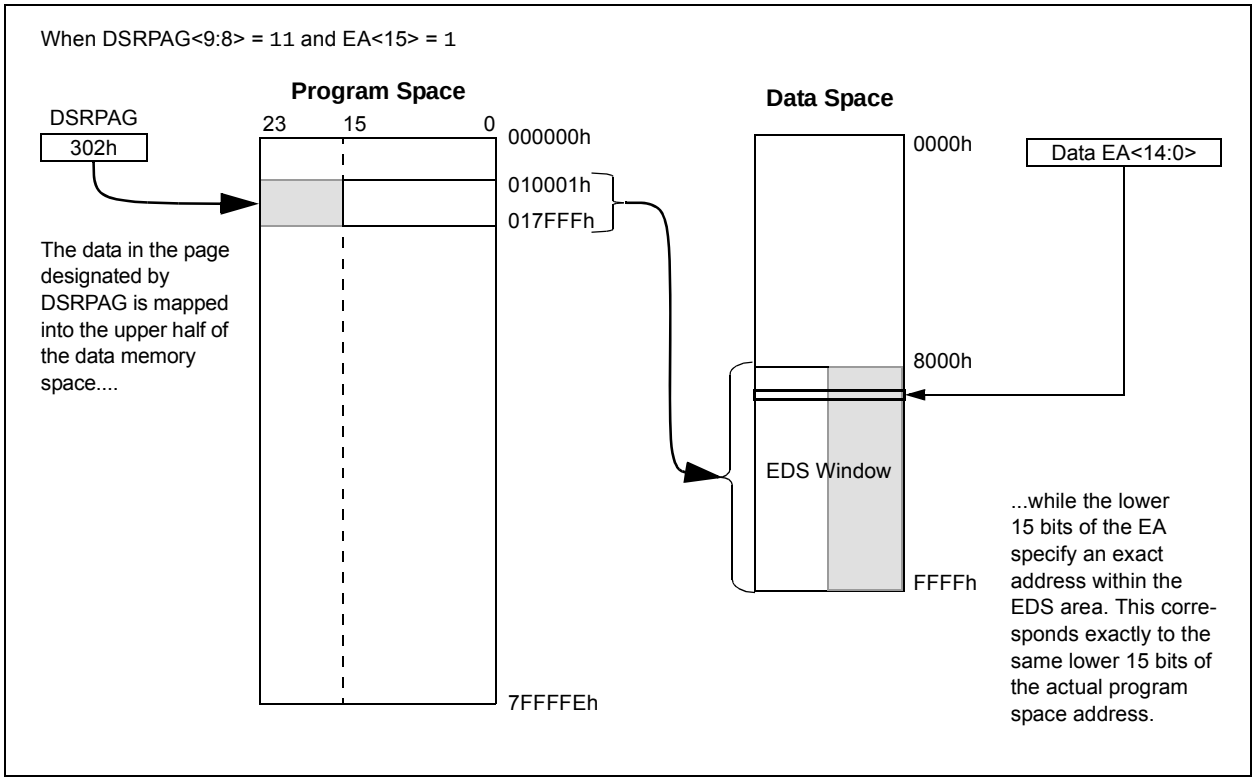


FIGURE 4-11: PROGRAM SPACE VISIBILITY OPERATION TO ACCESS UPPER WORD



5.0 DIRECT MEMORY ACCESS CONTROLLER (DMA)

Note: This data sheet summarizes the features of the PIC24FJ128GC010 family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to the *"PIC24F Family Reference Manual"*, **"Direct Memory Access Controller (DMA)"** (DS39742). The information in this data sheet supersedes the information in the FRM.

The Direct Memory Access (DMA) controller is designed to service high data throughput peripherals operating on the SFR bus, allowing them to access data memory directly and alleviating the need for CPU intensive management. By allowing these data intensive peripherals to share their own data path, the main data bus is also deloaded, resulting in additional power savings.

The DMA controller functions both as a peripheral and a direct extension of the CPU. It is located on the microcontroller data bus, between the CPU and DMA controller-enabled peripherals, with direct access to SRAM. This partitions the SFR bus into two buses, allowing the DMA controller access to the DMA capable peripherals located on the new DMA SFR bus. The controller serves as a master device on the DMA SFR bus, controlling data flow from DMA capable peripherals.

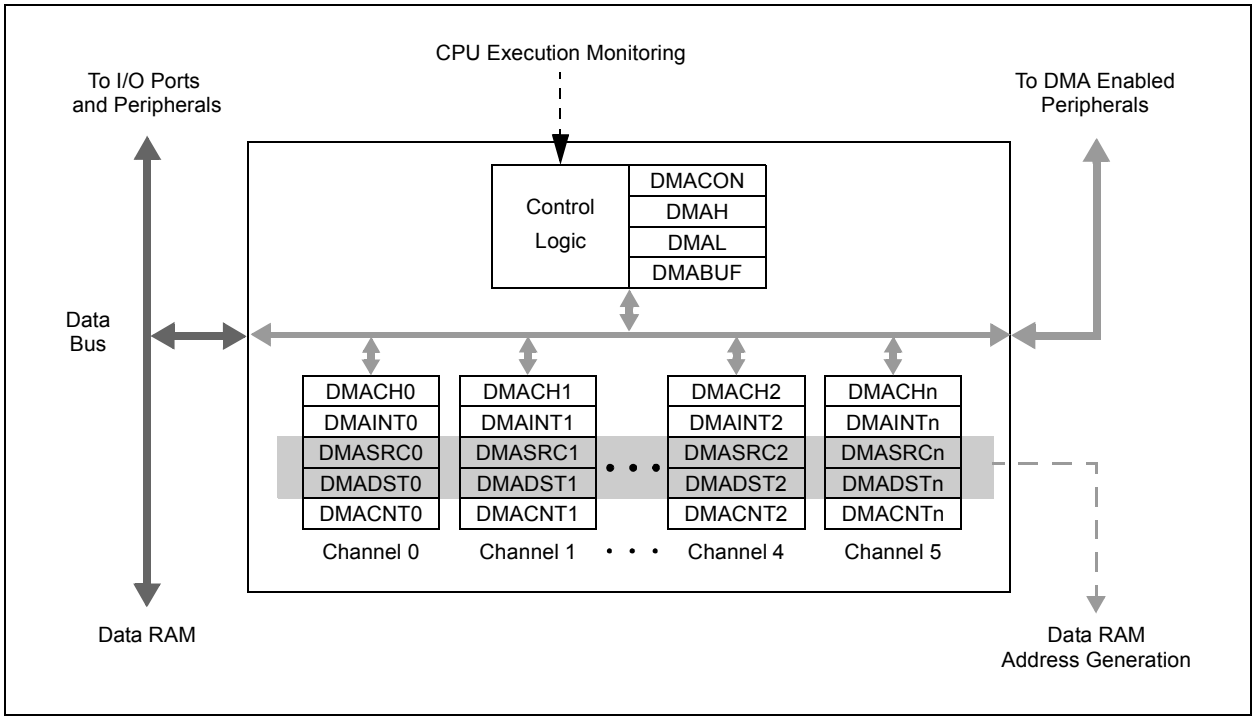
The controller also monitors CPU instruction processing directly, allowing it to be aware of when the CPU requires access to peripherals on the DMA bus, and automatically relinquishing control to the CPU as needed. This increases the effective bandwidth for handling data without DMA operations, causing a processor Stall. This makes the controller essentially transparent to the user.

The DMA controller includes these features:

- Six multiple independent and independently programmable channels
- Concurrent operation with the CPU (no DMA caused Wait states)
- DMA bus arbitration
- Five Programmable Address modes
- Four Programmable Transfer modes
- Four Flexible Internal Data Transfer modes
- Byte or word support for data transfer
- 16-Bit Source and Destination Address register for each channel, dynamically updated and reloadable
- 16-Bit Transaction Count register, dynamically updated and reloadable
- Upper and Lower Address Limit registers
- Counter half-full level interrupt
- Software triggered transfer
- Null Write mode for symmetric buffer operations

A simplified block diagram of the DMA controller is shown if Figure 5-1.

FIGURE 5-1: DMA CONTROLLER FUNCTIONAL BLOCK DIAGRAM



5.1 Summary of DMA Operations

The DMA controller is capable of moving data between addresses according to a number of different parameters. Each of these parameters can be independently configured for any transaction. In addition, any or all of the DMA channels can independently perform a different transaction at the same time. Transactions are classified by these parameters:

- Source and destination (SFRs and data RAM)
- Data size (byte or word)
- Trigger source
- Transfer mode (One-Shot, Repeated or Continuous)
- Addressing modes (Fixed Address or Address Blocks, with or without Address Increment/Decrement)

In addition, the DMA controller provides channel priority arbitration for all channels.

5.1.1 SOURCE AND DESTINATION

Using the DMA controller, data may be moved between any two addresses in the Data Space. The SFR space (0000h to 07FFh) or the data RAM space (0800h to FFFFh) can serve as either the source or the destination. Data can be moved between these areas in either direction, or between addresses in either area. The four different combinations are shown in Figure 5-2.

If it is necessary to protect areas of data RAM, the DMA controller allows the user to set upper and lower address boundaries for operations in the Data Space above the SFR space. The boundaries are set by the DMAH and DMAL Limit registers. If a DMA channel attempts an operation outside of the address boundaries, the transaction is terminated and an interrupt is generated.

5.1.2 DATA SIZE

The DMA controller can handle both 8-bit and 16-bit transactions. Size is user-selectable using the SIZE bit (DMACHn<1>). By default, each channel is configured for word-size transactions. When byte-size transactions are chosen, the LSb of the source and/or destination address determines if the data represents the upper or lower byte of the data RAM location.

5.1.3 TRIGGER SOURCE

The DMA controller can use 63 of the device's interrupt sources to initiate a transaction. The DMA trigger sources occur in reverse order of their natural interrupt priority and are shown in Table 5-1.

These sources cannot be used as DMA triggers:

- Input Capture 8 and 9
- Output Compare 7, 8 and 9
- USB

Since the source and destination addresses for any transaction can be programmed independently of the trigger source, the DMA controller can use any trigger to perform an operation on any peripheral. This also allows DMA channels to be cascaded to perform more complex transfer operations.

5.1.4 TRANSFER MODE

The DMA controller supports four types of data transfers, based on the volume of data to be moved for each trigger.

- One-Shot: A single transaction occurs for each trigger.
- Continuous: A series of back-to-back transactions occur for each trigger. The number of transactions is determined by the DMACNTn Transaction Counter register.
- Repeated One-Shot: A single transaction is performed repeatedly, once per trigger, until the DMA channel is disabled.
- Repeated Continuous: A series of transactions are performed repeatedly, one cycle per trigger, until the DMA channel is disabled.

All transfer modes allow the option to have the source and destination addresses, and counter value automatically reloaded after the completion of a transaction. Repeated mode transfers do this automatically.

5.1.5 ADDRESSING MODES

The DMA controller also supports transfers between single addresses or address ranges. The four basic options are:

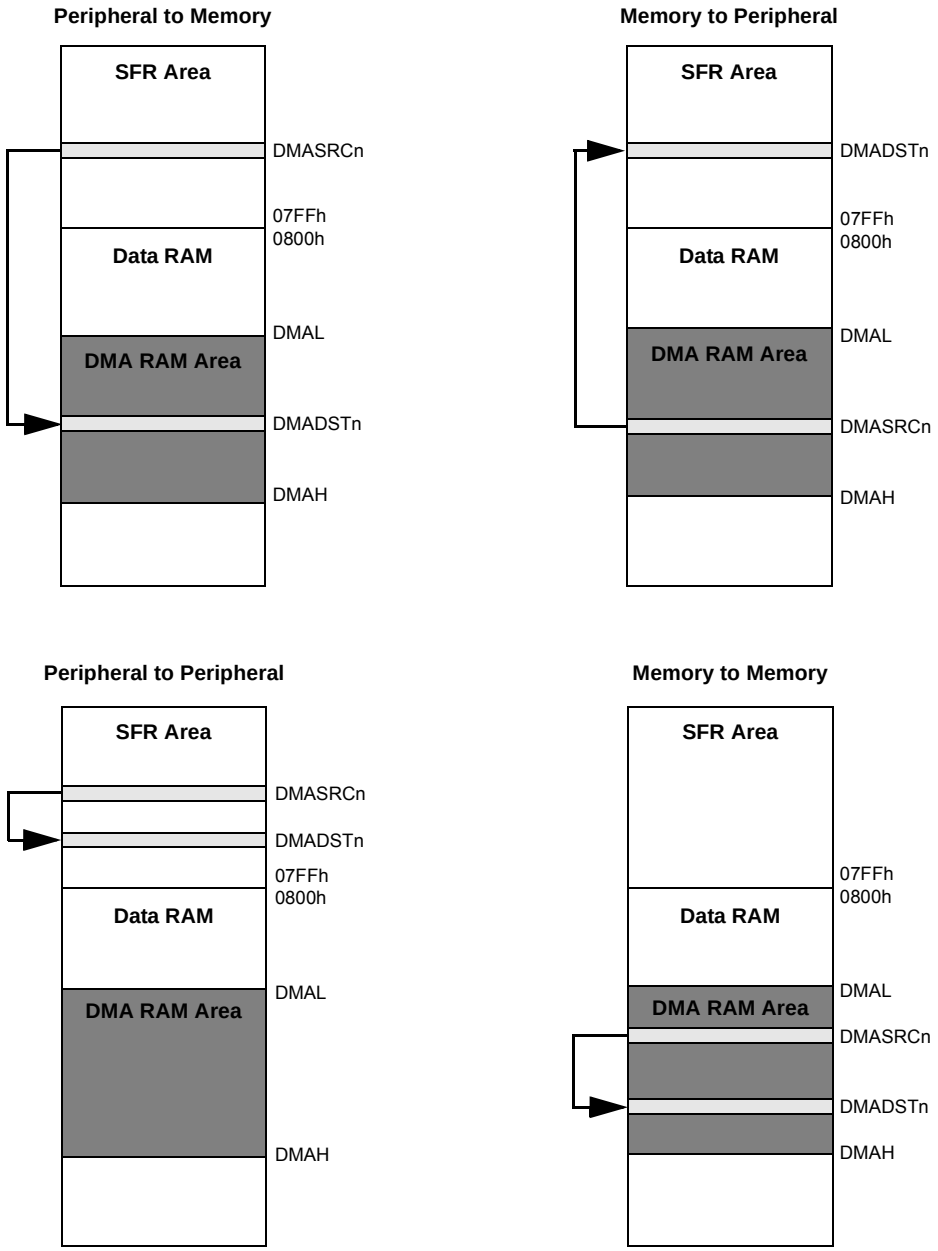
- Fixed-to-Fixed: Between two constant addresses
- Fixed-to-Block: From a constant source address to a range of destination addresses
- Block-to-Fixed: From a range of source addresses to a single, constant destination address
- Block-to-Block: From a range to source addresses to a range of destination addresses

The option to select auto-increment or auto-decrement of source and/or destination addresses is available for Block Addressing modes.

In addition to the four basic modes, the DMA controller also supports Peripheral Indirect Addressing (PIA) mode, where the source or destination address is generated jointly by the DMA controller and a PIA capable peripheral. When enabled, the DMA channel provides a base source and/or destination address, while the peripheral provides a fixed range, offset address.

For PIC24FJ128GC010 family devices, the 12-bit A/D Converter module is the only PIA capable peripheral. Details for its use in PIA mode are provided in **Section 26.0 "12-Bit High-Speed, Pipeline A/D Converter"**.

FIGURE 5-2: TYPES OF DMA DATA TRANSFERS



Note: Relative sizes of memory areas are not shown to scale.

5.1.6 CHANNEL PRIORITY

Each DMA channel functions independently of the others, but also competes with the others for access to the data and DMA buses. When access collisions occur, the DMA controller arbitrates between the channels using a user-selectable priority scheme. Two schemes are available:

- Round Robin: When two or more channels collide, the lower numbered channel receives priority on the first collision. On subsequent collisions, the higher numbered channels each receive priority based on their channel number.
- Fixed: When two or more channels collide, the lowest numbered channel always receives priority, regardless of past history.

5.2 Typical Setup

To set up a DMA channel for a basic data transfer:

1. Enable the DMA controller (DMAEN = 1) and select an appropriate channel priority scheme by setting or clearing PRSSEL.
2. Program DMAH and DMAL with appropriate upper and lower address boundaries for data RAM operations.
3. Select the DMA channel to be used and disable its operation (CHEN = 0).
4. Program the appropriate source and destination addresses for the transaction into the channel's DMASRCn and DMADSTn registers. For PIA Mode Addressing, use the base address value.
5. Program the DMACNTn register for the number of triggers per transfer (One-Shot or Continuous modes) or the number of words (bytes) to be transferred (Repeated modes).
6. Set or clear the SIZE bit to select the data size.
7. Program the TRMODEx bits to select the Data Transfer mode.
8. Program the SAMODEx and DAMODEx bits to select the addressing mode.
9. Enable the DMA channel by setting CHEN.
10. Enable the trigger source interrupt.

5.3 Peripheral Module Disable

Unlike other peripheral modules, the channels of the DMA controller cannot be individually powered down using the Peripheral Module Disable (PMD) registers. Instead, the channels are controlled as two groups. The DMA0MD bit (PMD7<4>) selectively controls DMACH0 through DMACH3. The DMA1MD bit (PMD7<5>) controls DMACH4 and DMACH5. Setting both bits effectively disables the DMA controller.

5.4 Registers

The DMA controller uses a number of registers to control its operation. The number of registers depends on the number of channels implemented for a particular device.

There are always four module level registers (one control and three buffer/address):

- DMACON: DMA Control Register (Register 5-1)
- DMAH and DMAL: DMA High and Low Address Limit Registers
- DMABUF: DMA Data Buffer Register

Each of the DMA channels implements five registers (two control and three buffer/address):

- DMACHn: DMA Channel n Control Register (Register 5-2)
- DMAINTn: DMA Channel n Interrupt Control Register (Register 5-3)
- DMASRCn: DMA Data Source Address Pointer for Channel n Register
- DMADSTn: DMA Data Destination Source for Channel n Register
- DMACNTn: DMA Transaction Counter for Channel n Register

For PIC24FJ128GC010 family devices, there are a total of 34 DMA registers.

REGISTER 5-1: DMACON: DMA ENGINE CONTROL REGISTER

R/W-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
DMAEN	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	PRSSEL
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 **DMAEN:** DMA Module Enable bit
1 = Enables module
0 = Disables module and terminates all active DMA operation(s)

bit 14-1 **Unimplemented:** Read as '0'

bit 0 **PRSSEL:** Channel Priority Scheme Selection bit
1 = Round robin scheme
0 = Fixed priority scheme

REGISTER 5-2: DMACHn: DMA CHANNEL n CONTROL REGISTER

U-0	U-0	U-0	r-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	r	—	NULLW	RELOAD(1)	CHREQ(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
SAMODE1	SAMODE0	DAMODE1	DAMODE0	TRMODE1	TRMODE0	SIZE	CHEN
bit 7							bit 0

Legend:	r = Reserved 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 15-13 **Unimplemented:** Read as '0'

bit 12 **Reserved:** Maintain as '0'

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

bit 10 **NULLW:** Null Write Mode bit
- 1 = A dummy write is initiated to DMASRCn for every write to DMADSTn
- 0 = No dummy write is initiated

bit 9 **RELOAD:** Address and Count Reload bit(1)
- 1 = DMASRCn, DMADSTn and DMACNTn registers are reloaded to their previous values upon the start of the next operation
- 0 = DMASRCn, DMADSTn and DMACNTn are not reloaded on the start of the next operation(2)

bit 8 **CHREQ:** DMA Channel Software Request bit(3)
- 1 = A DMA request is initiated by software; automatically cleared upon completion of a DMA transfer
- 0 = No DMA request is pending

bit 7-6 **SAMODE<1:0>:** Source Address Mode Selection bits
- 11 = Reserved
- 10 = DMASRCn is decremented based on the SIZE bit after a transfer completion
- 01 = DMASRCn is incremented based on the SIZE bit after a transfer completion
- 00 = DMASRCn remains unchanged after a transfer completion

bit 5-4 **DAMODE<1:0>:** Destination Address Mode Selection bits
- 11 = Reserved
- 10 = DMADSTn is decremented based on the SIZE bit after a transfer completion
- 01 = DMADSTn is incremented based on the SIZE bit after a transfer completion
- 00 = DMADSTn remains unchanged after a transfer completion

bit 3-2 **TRMODE<1:0>:** Transfer Mode Selection bits
- 11 = Repeated Continuous
- 10 = Continuous
- 01 = Repeated One-Shot
- 00 = One-Shot

bit 1 **SIZE:** Data Size Selection bit
- 1 = Byte (8-bit)
- 0 = Word (16-bit)

bit 0 **CHEN:** DMA Channel Enable bit
- 1 = The corresponding channel is enabled
- 0 = The corresponding channel is disabled

Note 1: Only the original DMACNTn is required to be stored to recover the original DMASRCn and DMADSTn.

2: DMASRCn, DMADSTn and DMACNTn are always reloaded in Repeated mode transfers (DMACHn<2> = 1), regardless of the state of the RELOAD bit.

3: The number of transfers executed while CHREQ is set depends on the configuration of TRMODE<1:0>.

REGISTER 5-3: DMAINTn: DMA CHANNEL n INTERRUPT REGISTER

R-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DBUFWF(1)	—	CHSEL5	CHSEL4	CHSEL3	CHSEL2	CHSEL1	CHSEL0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0
HIGHIF(1,2)	LOWIF(1,2)	DONEIF(1)	HALFIF(1)	OVRUNIF(1)	—	—	HALFEN
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 **DBUFWF:** Buffered Data Write Flag bit(1)
- 1 = The content of the DMA buffer has not been written to the location specified in DMADSTn or DMASRCn in Null Write mode
- 0 = The content of the DMA buffer has been written to the location specified in DMADSTn or DMASRCn in Null Write mode

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

bit 13-8 **CHSEL<5:0>:** DMA Channel Trigger Selection bits

See Table 5-1 for a complete list.

bit 7 **HIGHIF:** DMA High Address Limit Interrupt Flag bit(1,2)
- 1 = The DMA channel has attempted to access an address higher than DMAH or the upper limit of the data RAM space
- 0 = The DMA channel has not invoked the high address limit interrupt

bit 6 **LOWIF:** DMA Low Address Limit Interrupt Flag bit(1,2)
- 1 = The DMA channel has attempted to access the DMA SFR address lower than DMAL but above the SFR range (07FFh)
- 0 = The DMA channel has not invoked the low address limit interrupt

bit 5 **DONEIF:** DMA Complete Operation Interrupt Flag bit(1)

If CHEN = 1:
- 1 = The previous DMA session has ended with completion
- 0 = The current DMA session has not yet completed

If CHEN = 0:
- 1 = The previous DMA session has ended with completion
- 0 = The previous DMA session has ended without completion

bit 4 **HALFIF:** DMA 50% Watermark Level Interrupt Flag bit(1)
- 1 = DMACNTn has reached the halfway point to 0000h
- 0 = DMACNTn has not reached the halfway point

bit 3 **OVRUNIF:** DMA Channel Overrun Flag bit(1)
- 1 = The DMA channel is triggered while it is still completing the operation based on the previous trigger
- 0 = The overrun condition has not occurred

bit 2-1 **Unimplemented:** Read as '0'

bit 0 **HALFEN:** Halfway Completion Watermark bit
- 1 = Interrupts are invoked when DMACNTn has reached its halfway point and at completion
- 0 = An interrupt is invoked only at the completion of the transfer

Note 1: Setting these flags in software does not generate an interrupt.

2: Testing for address limit violations (DMASRCn or DMADSTn is either greater than DMAH or less than DMAL) is NOT done before the actual access.

TABLE 5-1: DMA TRIGGER SOURCES

CHSEL<5:0>	Trigger (Interrupt)	CHSEL<5:0>	Trigger (Interrupt)
000000	(Unimplemented)	100000	UART2 Transmit
000001	DAC2	100001	UART2 Receive
000010	LCD	100010	External Interrupt 2
000011	UART4 Transmit	100011	Timer5
000100	UART4 Receive	100100	Timer4
000101	UART4 Error	100101	Output Compare 4
000110	UART3 Transmit	100110	Output Compare 3
000111	UART3 Receive	100111	DMA Channel 2
001000	UART3 Error	101000	DAC1
001001	CTMU Event	101001	External Interrupt 1
001010	HLVD	101010	Interrupt-on-Change
001011	CRC Done	101011	Comparators Event
001100	UART2 Error	101100	I2C1 Master Event
001101	UART1 Error	101101	I2C1 Slave Event
001110	RTCC	101110	DMA Channel 1
001111	DMA Channel 5	101111	Pipeline A/D Converter
010000	External Interrupt 4	110000	UART1 Transmit
010001	External Interrupt 3	110001	UART1 Receive
010010	I2C2 Master Event	110010	SPI1 Event
010011	I2C2 Slave Event	110011	SPI1 Error
010100	DMA Channel 4	110100	Timer3
010101	EPMP	110101	Timer2
010110	Output Compare 7	110110	Output Compare 2
010111	Output Compare 6	110111	Input Capture 2
011000	Output Compare 5	111000	DMA Channel 0
011001	Input Capture 6	111001	Timer1
011010	Input Capture 5	111010	Output Compare 1
011011	Input Capture 4	111011	Input Capture 1
011100	Input Capture 3	111100	External Interrupt 0
011101	DMA Channel 3	111101	Op Amp 2
011110	SPI2 Event	111110	Op Amp 1
011111	SPI2 Error	111111	Sigma-Delta A/D Converter

6.0 FLASH PROGRAM MEMORY

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"*, **"Program Memory"** (DS39715). The information in this data sheet supersedes the information in the FRM.

The PIC24FJ128GC010 family of devices contains internal Flash program memory for storing and executing application code. The program memory is readable, writable and erasable. The Flash can be programmed in three ways:

- In-Circuit Serial Programming™ (ICSP™)
- Run-Time Self-Programming (RTSP)
- Enhanced In-Circuit Serial Programming (Enhanced ICSP)

ICSP allows a PIC24FJ128GC010 family device to be serially programmed while in the end application circuit. This is simply done with two lines for the programming clock and programming data (named PGECx and PGEDx, respectively), and three other lines for power (V_{DD}), ground (V_{SS}) and Master Clear ($\overline{MCLR}$). This allows customers to manufacture boards with unprogrammed devices and then program the microcontroller just before shipping the product. This also allows the most recent firmware or a custom firmware to be programmed.

RTSP is accomplished using TBLRD (Table Read) and TBLWT (Table Write) instructions. With RTSP, the user may write program memory data in blocks of 64 instructions (192 bytes) at a time and erase program memory in blocks of 512 instructions (1536 bytes) at a time.

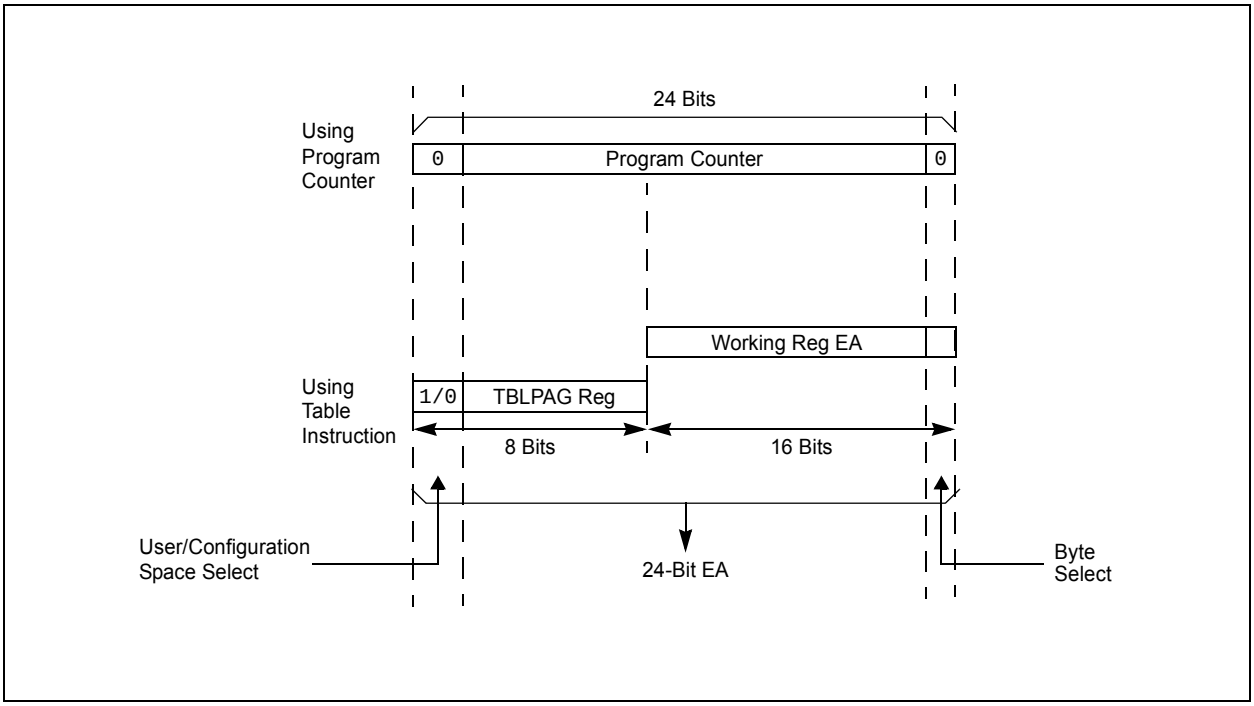
6.1 Table Instructions and Flash Programming

Regardless of the method used, all programming of Flash memory is done with the table read and write instructions. These allow direct read and write access to the program memory space from the data memory while the device is in normal operating mode. The 24-bit target address in the program memory is formed using the TBLPAG<7:0> bits and the Effective Address (EA) from a W register, specified in the table instruction, as shown in Figure 6-1.

The TBLRDL and the TBLWTL instructions are used to read or write to bits<15:0> of program memory. TBLRDL and TBLWTL can access program memory in both Word and Byte modes.

The TBLRDH and TBLWTH instructions are used to read or write to bits<23:16> of program memory. TBLRDH and TBLWTH can also access program memory in Word or Byte mode.

FIGURE 6-1: ADDRESSING FOR TABLE REGISTERS



6.2 RTSP Operation

The PIC24F Flash program memory array is organized into rows of 64 instructions or 192 bytes. RTSP allows the user to erase blocks of eight rows (512 instructions) at a time and to program one row at a time. It is also possible to program single words.

The 8-row erase blocks and single row write blocks are edge-aligned, from the beginning of program memory, on boundaries of 1536 bytes and 192 bytes, respectively.

When data is written to program memory using TBLWT instructions, the data is not written directly to memory. Instead, data written using table writes is stored in holding latches until the programming sequence is executed.

Any number of TBLWT instructions can be executed and a write will be successfully performed. However, 64 TBLWT instructions are required to write the full row of memory.

To ensure that no data is corrupted during a write, any unused address should be programmed with FFFFFFh. This is because the holding latches reset to an unknown state, so if the addresses are left in the Reset state, they may overwrite the locations on rows which were not rewritten.

The basic sequence for RTSP programming is to set up a Table Pointer, then do a series of TBLWT instructions to load the buffers. Programming is performed by setting the control bits in the NVMCON register.

Data can be loaded in any order and the holding registers can be written to multiple times before performing a write operation. Subsequent writes, however, will wipe out any previous writes.

Note:	Writing to a location multiple times without erasing is *not* recommended.

All of the table write operations are single-word writes (2 instruction cycles), because only the buffers are written. A programming cycle is required for programming each row.

6.3 Enhanced In-Circuit Serial Programming

Enhanced In-Circuit Serial Programming uses an on-board bootloader, known as the Program Executive (PE), to manage the programming process. Using an SPI data frame format, the Program Executive can erase, program and verify program memory. For more information on Enhanced ICSP, see the device programming specification.

6.4 Control Registers

There are two SFRs used to read and write the program Flash memory: NVMCON and NVMKEY.

The NVMCON register (Register 6-1) controls which blocks are to be erased, which memory type is to be programmed and when the programming cycle starts.

NVMKEY is a write-only register that is used for write protection. To start a programming or erase sequence, the user must consecutively write 55h and AAh to the NVMKEY register. Refer to **Section 6.5 “Programming Operations”** for further details.

REGISTER 6-1: NVMCON: FLASH MEMORY CONTROL REGISTER

R/S-0, HC(1)	R/W-0(1)	R-0, HSC(1)	U-0	U-0	U-0	U-0	U-0
WR	WREN	WRERR	—	—	—	—	—
bit 15							bit 8

U-0	R/W-0(1)	U-0	U-0	R/W-0(1)	R/W-0(1)	R/W-0(1)	R/W-0(1)
—	ERASE	—	—	NVMOP3(2)	NVMOP2(2)	NVMOP1(2)	NVMOP0(2)
bit 7							bit 0

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

bit 15 **WR:** Write Control bit(1)
- 1 = Initiates a Flash memory program or erase operation; the operation is self-timed and the bit is cleared by hardware once the operation is complete
- 0 = Program or erase operation is complete and inactive

bit 14 **WREN:** Write Enable bit(1)
- 1 = Enables Flash program/erase operations
- 0 = Inhibits Flash program/erase operations

bit 13 **WRERR:** Write Sequence Error Flag bit(1)
- 1 = An improper program or erase sequence attempt, or termination has occurred (bit is set automatically on any set attempt of the WR bit)
- 0 = The program or erase operation completed normally

bit 12-7 **Unimplemented:** Read as '0'

bit 6 **ERASE:** Erase/Program Enable bit(1)
- 1 = Performs the erase operation specified by NVMOP<3:0> on the next WR command
- 0 = Performs the program operation specified by NVMOP<3:0> on the next WR command

bit 5-4 **Unimplemented:** Read as '0'

bit 3-0 **NVMOP<3:0>:** NVM Operation Select bits(1,2)
- 1111 = Memory bulk erase operation (ERASE = 1) or no operation (ERASE = 0)(3)
- 0011 = Memory word program operation (ERASE = 0) or no operation (ERASE = 1)
- 0010 = Memory page erase operation (ERASE = 1) or no operation (ERASE = 0)
- 0001 = Memory row program operation (ERASE = 0) or no operation (ERASE = 1)

Note 1: These bits can only be reset on a Power-on Reset.

2: All other combinations of NVMOP<3:0> are unimplemented.

3: Available in ICSP™ mode only; refer to the device programming specification.

6.5 Programming Operations

A complete programming sequence is necessary for programming or erasing the internal Flash in RTSP mode. During a programming or erase operation, the processor stalls (Waits) until the operation is finished. Setting the WR bit (NVMCON<15>) starts the operation and the WR bit is automatically cleared when the operation is finished.

6.5.1 PROGRAMMING ALGORITHM FOR FLASH PROGRAM MEMORY

The user can program one row of Flash program memory at a time. To do this, it is necessary to erase the 8-row erase block containing the desired row. The general process is:

1. Read eight rows of program memory (512 instructions) and store in data RAM.
2. Update the program data in RAM with the desired new data.
3. Erase the block (see Example 6-1):
   a) Set the NVMOPx bits (NVMCON<3:0>) to '0010' to configure for block erase. Set the ERASE (NVMCON<6>) and WREN (NVMCON<14>) bits.
   b) Write the starting address of the block to be erased into the TBLPAG and W registers.
   c) Write 55h to NVMKEY.
   d) Write AAh to NVMKEY.
   e) Set the WR bit (NVMCON<15>). The erase cycle begins and the CPU stalls for the duration of the erase cycle. When the erase is done, the WR bit is cleared automatically.
4. Write the first 64 instructions from data RAM into the program memory buffers (see Example 6-3).
5. Write the program block to Flash memory:
   a) Set the NVMOPx bits to '0001' to configure for row programming. Clear the ERASE bit and set the WREN bit.
   b) Write 55h to NVMKEY.
   c) Write AAh to NVMKEY.
   d) Set the WR bit. The programming cycle begins and the CPU stalls for the duration of the write cycle. When the write to Flash memory is done, the WR bit is cleared automatically.
6. Repeat Steps 4 and 5, using the next available 64 instructions from the block in data RAM by incrementing the value in TBLPAG, until all 512 instructions are written back to Flash memory.

For protection against accidental operations, the write initiate sequence for NVMKEY must be used to allow any erase or program operation to proceed. After the programming command has been executed, the user must wait for the programming time until programming is complete. The two instructions following the start of the programming sequence should be NOPs, as shown in Example 6-4.

EXAMPLE 6-1: ERASING A PROGRAM MEMORY BLOCK (ASSEMBLY LANGUAGE CODE)

```
; Set up NVMCON for block erase operation
    MOV    #0x4042, W0                  ;
    MOV    W0, NVMCON                   ; Initialize NVMCON
; Init pointer to row to be ERASED
    MOV    #tblpage(PROG_ADDR), W0      ;
    MOV    W0, TBLPAG                   ; Initialize Program Memory (PM) Page Boundary SFR
    MOV    #tbloffset(PROG_ADDR), W0    ; Initialize in-page EA<15:0> pointer
    TBLWTL W0, [W0]                     ; Set base address of erase block
    DISI   #5                           ; Block all interrupts with priority <7
                                        ; for next 5 instructions
    MOV.B  #0x55, W0
    MOV    W0, NVMKEY                   ; Write the 0x55 key
    MOV.B  #0xAA, W1                    ;
    MOV    W1, NVMKEY                   ; Write the 0xAA key
    BSET   NVMCON, #WR                  ; Start the erase sequence
    NOP                                 ; Insert two NOPs after the erase
    NOP                                 ; command is asserted
```

EXAMPLE 6-2: ERASING A PROGRAM MEMORY BLOCK ('C' LANGUAGE CODE)

```
// C example using MPLAB C30
    unsigned long progAddr = 0xXXXXXX;          // Address of row to write
    unsigned int offset;
//Set up pointer to the first memory location to be written
    TBLPAG = progAddr>>16;                      // Initialize PM Page Boundary SFR
    offset = progAddr & 0xFFFF;                 // Initialize lower word of address
    __builtin_tblwtl(offset, 0x0000);           // Set base address of erase block
                                                // with dummy latch write
    NVMCON = 0x4042;                            // Initialize NVMCON
    asm("DISI #5");                             // Block all interrupts with priority <7
                                                // for next 5 instructions
    __builtin_write_NVM();                      // check function to perform unlock
                                                // sequence and set WR
```

EXAMPLE 6-3: LOADING THE WRITE BUFFERS

```
; Set up NVMCON for row programming operations
       MOV     #0x4001, W0                      ;
       MOV     W0, NVMCON                       ; Initialize NVMCON
; Set up a pointer to the first program memory location to be written
; program memory selected, and writes enabled
       MOV     #0x0000, W0                      ;
       MOV     W0, TBLPAG                       ; Initialize PM Page Boundary SFR
       MOV     #0x6000, W0                      ; An example program memory address
; Perform the TBLWT instructions to write the latches
; 0th_program_word
       MOV     #LOW_WORD_0, W2                  ;
       MOV     #HIGH_BYTE_0, W3                 ;
       TBLWTL  W2, [W0]                         ; Write PM low word into program latch
       TBLWTH  W3, [W0++]                       ; Write PM high byte into program latch
; 1st_program_word
       MOV     #LOW_WORD_1, W2                  ;
       MOV     #HIGH_BYTE_1, W3                 ;
       TBLWTL  W2, [W0]                         ; Write PM low word into program latch
       TBLWTH  W3, [W0++]                       ; Write PM high byte into program latch
;  2nd_program_word
       MOV     #LOW_WORD_2, W2                  ;
       MOV     #HIGH_BYTE_2, W3                 ;
       TBLWTL  W2, [W0]                         ; Write PM low word into program latch
       TBLWTH  W3, [W0++]                       ; Write PM high byte into program latch
       •
       •
       •
; 63rd_program_word
       MOV     #LOW_WORD_63, W2                 ;
       MOV     #HIGH_BYTE_63, W3                ;
       TBLWTL  W2, [W0]                         ; Write PM low word into program latch
       TBLWTH  W3, [W0]                         ; Write PM high byte into program latch
```

EXAMPLE 6-4: INITIATING A PROGRAMMING SEQUENCE

```
       DISI    #5                       ; Block all interrupts with priority <7
                                        ; for next 5 instructions
       MOV.B   #0x55, W0
       MOV     W0, NVMKEY               ; Write the 0x55 key
       MOV.B   #0xAA, W1                ;
       MOV     W1, NVMKEY               ; Write the 0xAA key
       BSET    NVMCON, #WR              ; Start the programming sequence
       NOP                              ; Required delays
       NOP
       BTSC    NVMCON, #15              ; and wait for it to be
       BRA     $-2                      ; completed
```

6.5.2 PROGRAMMING A SINGLE WORD OF FLASH PROGRAM MEMORY

If a Flash location has been erased, it can be programmed using table write instructions to write an instruction word (24-bit) into the write latch. The TBLPAG register is loaded with the 8 Most Significant Bytes (MSBs) of the Flash address. The TBLWTL and TBLWTH instructions write the desired data into the write latches and specify the lower 16 bits of the program memory address to write to. To configure the NVMCON register for a word write, set the NVMOPx bits (NVMCON<3:0>) to '0011'. The write is performed by executing the unlock sequence and setting the WR bit (see Example 6-5). An equivalent procedure in 'C' compiler, using the MPLAB® C30 compiler and built-in hardware functions, is shown in Example 6-6.

EXAMPLE 6-5: PROGRAMMING A SINGLE WORD OF FLASH PROGRAM MEMORY

```
; Setup a pointer to data Program Memory
    MOV     #tblpage(PROG_ADDR), W0       ;
    MOV     W0, TBLPAG                    ;Initialize PM Page Boundary SFR
    MOV     #tbloffset(PROG_ADDR), W0     ;Initialize a register with program memory address

    MOV     #LOW_WORD_N, W2               ;
    MOV     #HIGH_BYTE_N, W3              ;
    TBLWTL  W2, [W0]                      ; Write PM low word into program latch
    TBLWTH  W3, [W0++]                    ; Write PM high byte into program latch

; Setup NVMCON for programming one word to data Program Memory
    MOV     #0x4003, W0                   ;
    MOV     W0, NVMCON                    ; Set NVMOP bits to 0011

    DISI    #5                            ; Disable interrupts while the KEY sequence is written
    MOV.B   #0x55, W0                     ; Write the key sequence
    MOV     W0, NVMKEY
    MOV.B   #0xAA, W0
    MOV     W0, NVMKEY
    BSET    NVMCON, #WR                   ; Start the write cycle
    NOP                                   ; Required delays
    NOP
```

EXAMPLE 6-6: PROGRAMMING A SINGLE WORD OF FLASH PROGRAM MEMORY ('C' LANGUAGE CODE)

```
// C example using MPLAB C30
unsigned int offset;
unsigned long progAddr = 0xXXXXXX;          // Address of word to program
unsigned int progDataL = 0xXXXX;            // Data to program lower word
unsigned char progDataH = 0xXX;             // Data to program upper byte

//Set up NVMCON for word programming
NVMCON = 0x4003;                            // Initialize NVMCON

//Set up pointer to the first memory location to be written
TBLPAG = progAddr>>16;                      // Initialize PM Page Boundary SFR
offset = progAddr & 0xFFFF;                 // Initialize lower word of address

//Perform TBLWT instructions to write latches
__builtin_tblwtl(offset, progDataL);        // Write to address low word
__builtin_tblwth(offset, progDataH);        // Write to upper byte
asm("DISI #5");                             // Block interrupts with priority <7
                                            // for next 5 instructions
__builtin_write_NVM();                      // C30 function to perform unlock
                                            // sequence and set WR
```

7.0 RESETS

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*", **"Reset"** (DS39712). The information in this data sheet supersedes the information in the FRM.

The Reset module combines all Reset sources and controls the device Master Reset Signal, $\overline{SYSRST}$. The following is a list of device Reset sources:

- POR: Power-on Reset
- $\overline{MCLR}$: Pin Reset
- SWR: RESET Instruction
- WDT: Watchdog Timer Reset
- BOR: Brown-out Reset
- CM: Configuration Mismatch Reset
- TRAPR: Trap Conflict Reset
- IOPUWR: Illegal Opcode Reset
- UWR: Uninitialized W Register Reset

A simplified block diagram of the Reset module is shown in Figure 7-1.

Any active source of Reset will make the $\overline{SYSRST}$ signal active. Many registers associated with the CPU and peripherals are forced to a known Reset state. Most registers are unaffected by a Reset; their status is unknown on POR and unchanged by all other Resets.

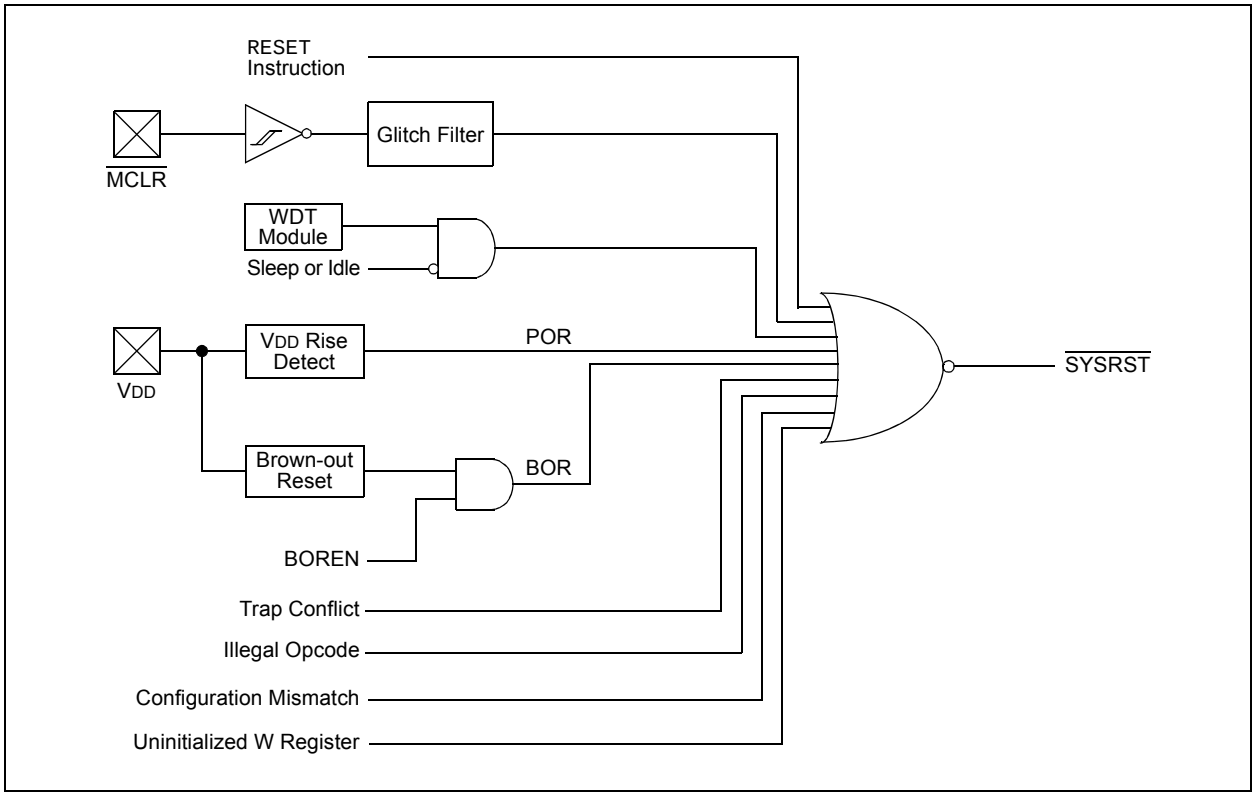
Note: Refer to the specific peripheral or CPU section of this manual for register Reset states.

All types of device Reset will set a corresponding status bit in the RCON register to indicate the type of Reset (see Register 7-1). In addition, Reset events occurring while an extreme power-saving feature is in use (such as VBAT) will set one or more status bits in the RCON2 register (Register 7-2). A POR will clear all bits, except for the BOR and POR (RCON<1:0>) bits, which are set. The user may set or clear any bit at any time during code execution. The RCON bits only serve as status bits. Setting a particular Reset status bit in software will not cause a device Reset to occur.

The RCON register also has other bits associated with the Watchdog Timer and device power-saving states. The function of these bits is discussed in other sections of this data sheet.

Note: The status bits in the RCON registers should be cleared after they are read so that the next RCON register values after a device Reset will be meaningful.

FIGURE 7-1: RESET SYSTEM BLOCK DIAGRAM



REGISTER 7-1: RCON: RESET CONTROL REGISTER

R/W-0	R/W-0	U-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
TRAPR(1)	IOPUWR(1)	—	RETEN(2)	—	DPSLP(1)	CM(1)	PMSLP(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-1	R/W-1
EXTR(1)	SWR(1)	SWDTEN(4)	WDTO(1)	SLEEP(1)	IDLE(1)	BOR(1)	POR(1)
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 **TRAPR:** Trap Reset Flag bit(1)
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)
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 **Unimplemented:** Read as '0'

bit 12 **RETEN:** Retention Mode Enable bit(2)
1 = Retention mode is enabled while device is in Sleep modes (1.2V regulator supplies to the core)
0 = Retention mode is disabled; normal voltage levels are present

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

bit 10 **DPSLP:** Deep Sleep Flag bit(1)
1 = Device has been in Deep Sleep mode
0 = Device has not been in Deep Sleep mode

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

bit 8 **PMSLP:** Program Memory Power During Sleep bit(3)
1 = Program memory bias voltage remains powered during Sleep
0 = Program memory bias voltage is powered down during Sleep

bit 7 **EXTR:** External Reset ($\overline{MCLR}$) Pin bit(1)
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)
1 = A RESET instruction has been executed
0 = A RESET instruction has not been executed

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 $\overline{LPCFG}$ Configuration bit is '1' (unprogrammed), the retention regulator is disabled and the RETEN bit has no effect.

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.

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

REGISTER 7-1: RCON: RESET CONTROL REGISTER (CONTINUED)

bit 5 **SWDTEN:** Software Enable/Disable of WDT bit(4)
1 = WDT is enabled
0 = WDT is disabled

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

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

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

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

bit 0 **POR:** Power-on Reset Flag bit(1)
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 $\overline{\text{LPCFG}}$ Configuration bit is '1' (unprogrammed), the retention regulator is disabled and the RETEN bit has no effect.

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.

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

REGISTER 7-2: RCON2: RESET AND SYSTEM CONTROL REGISTER 2

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	r-0	R/CO-1	R/CO-1	R/CO-1	R/CO-0
—	—	—	r	VDDBOR(1)	VDDPOR(1,2)	VBPOR(1,3)	VBAT(1)
bit 7							bit 0

Legend:	CO = Clearable Only bit	r = Reserved 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 15-5 **Unimplemented:** Read as '0'

bit 4 **Reserved:** Maintain as '0'

bit 3 **VDDBOR:** VDD Brown-out Reset Flag bit(1)
- 1 = A VDD Brown-out Reset has occurred (set by hardware)
- 0 = A VDD Brown-out Reset has not occurred

bit 2 **VDDPOR:** VDD Power-on Reset Flag bit(1,2)
- 1 = A VDD Power-on Reset has occurred (set by hardware)
- 0 = A VDD Power-on Reset has not occurred

bit 1 **VBPOR:** VBAT Power-on Reset Flag bit(1,3)
- 1 = A VBAT POR has occurred (no battery connected to VBAT pin or VBAT power is below Deep Sleep semaphore retention level, set by hardware)
- 0 = A VBAT POR has not occurred

bit 0 **VBAT:** VBAT Flag bit(1)
- 1 = A POR exit has occurred while power was applied to VBAT pin (set by hardware)
- 0 = A POR exit from VBAT has not occurred

Note 1: This bit is set in hardware only; it can only be cleared in software.

2: Indicates a VDD POR. Setting the POR bit (RCON<0>) indicates a VCORE POR.

3: This bit is set when the device is originally powered up, even if power is present on VBAT.

TABLE 7-1: 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>)	$\overline{\text{MCLR}}$ 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
DPSLP (RCON<10>)	PWRSAV #0 Instruction while DSEN bit set	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.

7.1 Special Function Register Reset States

Most of the Special Function Registers (SFRs) associated with the PIC24F CPU and peripherals are reset to a particular value at a device Reset. The SFRs are grouped by their peripheral or CPU function and their Reset values are specified in each section of this manual.

The Reset value for each SFR does not depend on the type of Reset, with the exception of four registers. The Reset value for the Reset Control register, RCON, will depend on the type of device Reset. The Reset value for the Oscillator Control register, OSCCON, will depend on the type of Reset and the programmed values of the FNOSCx bits in Flash Configuration Word 2 (CW2) (see Table 7-2). The RCFGCAL and NVMCON registers are only affected by a POR.

7.2 Device Reset Times

The Reset times for various types of device Reset are summarized in Table 7-3. Note that the System Reset signal, $\overline{\text{SYSRST}}$, is released after the POR delay time expires.

The time at which the device actually begins to execute code will also depend on the system oscillator delays, which include the Oscillator Start-up Timer (OST) and the PLL lock time. The OST and PLL lock times occur in parallel with the applicable $\overline{\text{SYSRST}}$ delay times.

The Fail-Safe Clock Monitor (FSCM) delay determines the time at which the FSCM begins to monitor the system clock source after the $\overline{\text{SYSRST}}$ signal is released.

7.3 Brown-out Reset (BOR)

PIC24FJ128GC010 family devices implement a BOR circuit that provides the user with several configuration and power-saving options. The BOR is controlled by the BOREN (CW3<12>) Configuration bit.

When BOR is enabled, any drop of V_{DD} below the BOR threshold results in a device BOR. Threshold levels are described in **Section 37.1 "DC Characteristics"** (Parameter DC17B).

7.4 Clock Source Selection at Reset

If clock switching is enabled, the system clock source at device Reset is chosen, as shown in Table 7-2. If clock switching is disabled, the system clock source is always selected according to the Oscillator Configuration bits. Refer to the "*PIC24F Family Reference Manual*", **Section 6.0 "Oscillator"** (DS39700) for further details.

TABLE 7-2: OSCILLATOR SELECTION vs. TYPE OF RESET (CLOCK SWITCHING ENABLED)

Reset Type	Clock Source Determinant
POR	FNOSC<2:0> Configuration bits (CW2<10:8>)
BOR	
$\overline{\text{MCLR}}$	COSC<2:0> Control bits (OSCCON<14:12>)
WDTO	
SWR	

TABLE 7-3: RESET DELAY TIMES FOR VARIOUS DEVICE RESETS

Reset Type	Clock Source	$\overline{\text{SYSRST}}$ Delay	System Clock Delay	Notes
POR	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.

3: TRST = Internal State Reset time (2 μs nominal).

4: TOST = Oscillator Start-up Timer (OST). 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.

7.4.1 POR AND LONG OSCILLATOR START-UP TIMES

The oscillator start-up circuitry and its associated delay timers are not linked to the device Reset delays that occur at power-up. Some crystal circuits (especially low-frequency crystals) will have a relatively long start-up time. Therefore, one or more of the following conditions is possible after $\overline{\text{SYSRST}}$ is released:

- The oscillator circuit has not begun to oscillate.
- The Oscillator Start-up Timer has not expired (if a crystal oscillator is used).
- The PLL has not achieved a lock (if PLL is used).

The device will not begin to execute code until a valid clock source has been released to the system. Therefore, the oscillator and PLL start-up delays must be considered when the Reset delay time must be known.

7.4.2 FAIL-SAFE CLOCK MONITOR (FSCM) AND DEVICE RESETS

If the FSCM is enabled, it will begin to monitor the system clock source when $\overline{\text{SYSRST}}$ is released. If a valid clock source is not available at this time, the device will automatically switch to the FRC oscillator and the user can switch to the desired crystal oscillator in the Trap Service Routine (TSR).

8.0 INTERRUPT CONTROLLER

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*", **"Interrupts"** (DS39707). The information in this data sheet supersedes the information in the FRM.

The PIC24F interrupt controller reduces the numerous peripheral interrupt request signals to a single interrupt request signal to the PIC24F CPU. It has the following features:

- Up to 8 processor exceptions and software traps
- Seven user-selectable priority levels
- Interrupt Vector Table (IVT) with up to 118 vectors
- Unique vector for each interrupt or exception source
- Fixed priority within a specified user priority level
- Alternate Interrupt Vector Table (AIVT) for debug support
- Fixed interrupt entry and return latencies

8.1 Interrupt Vector Table

The Interrupt Vector Table (IVT) is shown in Figure 8-1. The IVT resides in program memory, starting at location, 000004h. The IVT contains 126 vectors, consisting of 8 non-maskable trap vectors, plus up to 118 source interrupts. In general, each interrupt source has its own vector. Each interrupt vector contains a 24-bit wide address. The value programmed into each interrupt vector location is the starting address of the associated Interrupt Service Routine (ISR).

Interrupt vectors are prioritized in terms of their natural priority; this is linked to their position in the vector table. All other things being equal, lower addresses have a higher natural priority. For example, the interrupt associated with Vector 0 will take priority over interrupts at any other vector address.

PIC24FJ128GC010 family devices implement non-maskable traps and unique interrupts. These are summarized in Table 8-1 and Table 8-2.

8.1.1 ALTERNATE INTERRUPT VECTOR TABLE

The Alternate Interrupt Vector Table (AIVT) is located after the IVT, as shown in Figure 8-1. The ALTIVT (INTCON2<15>) control bit provides access to the AIVT. If the ALTIVT bit is set, all interrupt and exception processes will use the alternate vectors instead of the default vectors. The alternate vectors are organized in the same manner as the default vectors.

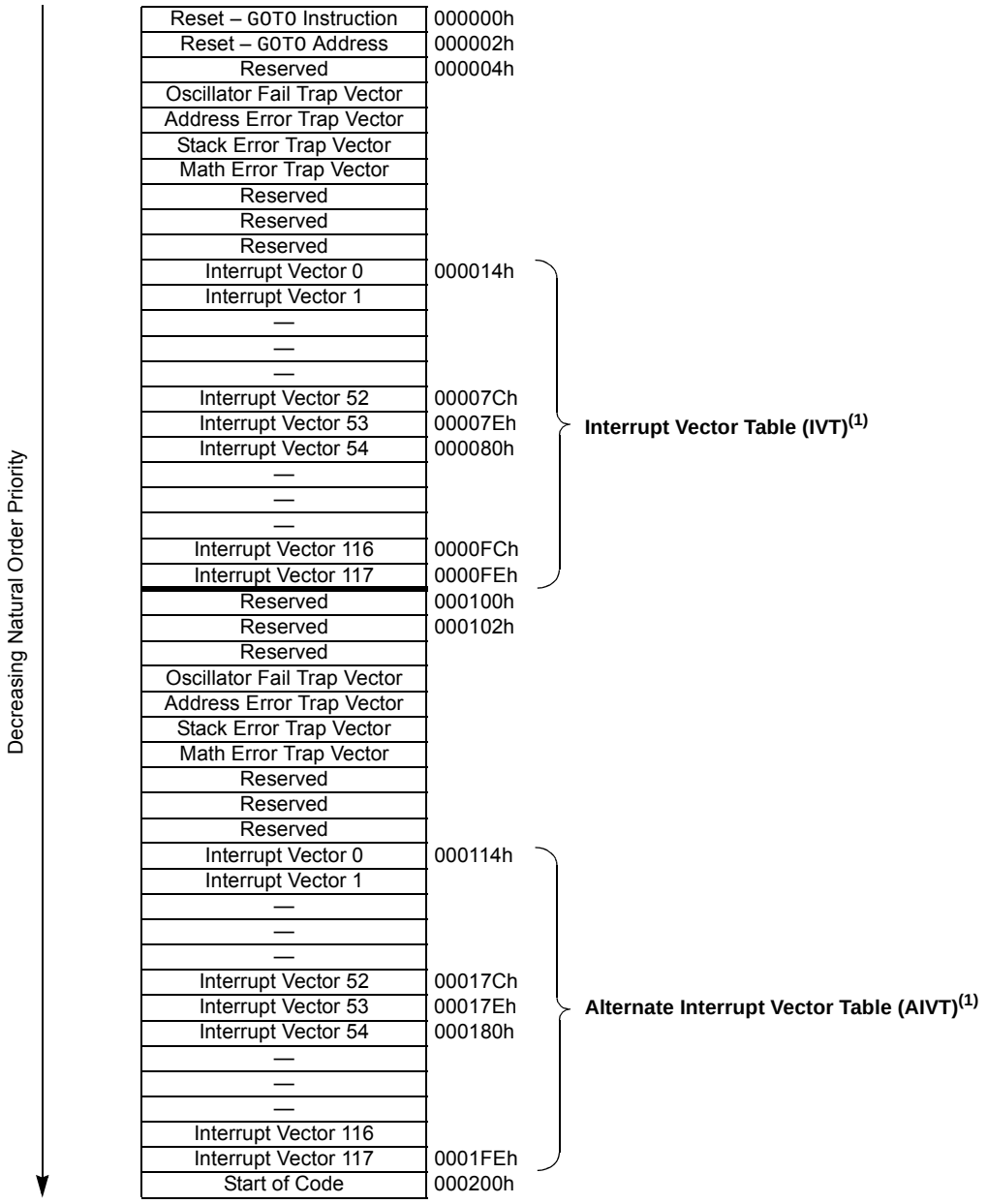
The AIVT supports emulation and debugging efforts by providing a means to switch between an application, and a support environment, without requiring the interrupt vectors to be reprogrammed. This feature also enables switching between applications for evaluation of different software algorithms at run time. If the AIVT is not needed, the AIVT should be programmed with the same addresses used in the IVT.

8.2 Reset Sequence

A device Reset is not a true exception because the interrupt controller is not involved in the Reset process. The PIC24F devices clear their registers in response to a Reset, which forces the PC to zero. The microcontroller then begins program execution at location, 000000h. The user programs a `GOTO` instruction at the Reset address, which redirects program execution to the appropriate start-up routine.

Note: Any unimplemented or unused vector locations in the IVT and AIVT should be programmed with the address of a default interrupt handler routine that contains a `RESET` instruction.

FIGURE 8-1: PIC24F INTERRUPT VECTOR TABLE



Note 1: See Table 8-2 for the interrupt vector list.

TABLE 8-1: TRAP VECTOR DETAILS

Vector Number	IVT Address	AIVT Address	Trap Source
0	000004h	000104h	Reserved
1	000006h	000106h	Oscillator Failure
2	000008h	000108h	Address Error
3	00000Ah	00010Ah	Stack Error
4	00000Ch	00010Ch	Math Error
5	00000Eh	00010Eh	Reserved
6	000010h	000110h	Reserved
7	000012h	000112h	Reserved

TABLE 8-2: IMPLEMENTED INTERRUPT VECTORS

Interrupt Source	Vector Number	IVT Address	AIVT Address	Interrupt Bit Locations		
				Flag	Enable	Priority
A/D (12-Bit Pipeline)	13	00002Eh	00012Eh	IFS0<13>	IEC0<13>	IPC3<6:4>
A/D (Sigma-Delta)	105	0000E6h	0001E6h	IFS6<9>	IEC6<9>	IPC26<6:4>
Comparator Event	18	000038h	000138h	IFS1<2>	IEC1<2>	IPC4<10:8>
CRC Generator	67	00009Ah	00019Ah	IFS4<3>	IEC4<3>	IPC16<14:12>
CTMU Event	77	0000AEh	0001AEh	IFS4<13>	IEC4<13>	IPC19<6:4>
DAC1	78	000080h	000180h	IFS4<14>	IEC4<14>	IPC19<10:8>
DAC2	79	000082h	000182h	IFS4<15>	IEC4<15>	IPC19<14:12>
DMA Channel 0	4	00001Ch	00011Ch	IFS0<4>	IEC0<4>	IPC1<2:0>
DMA Channel 1	14	000030h	000130h	IFS0<14>	IEC0<14>	IPC3<10:8>
DMA Channel 2	24	000044h	000144h	IFS1<8>	IEC1<8>	IPC6<2:0>
DMA Channel 3	36	00005Ch	00015Ch	IFS2<4>	IEC2<4>	IPC9<2:0>
DMA Channel 4	46	000070h	000170h	IFS2<14>	IEC2<14>	IPC11<10:8>
DMA Channel 5	61	00008Eh	00018Eh	IFS3<13>	IEC3<13>	IPC15<6:4>
External Interrupt 0	0	000014h	000114h	IFS0<0>	IEC0<0>	IPC0<2:0>
External Interrupt 1	20	00003Ch	00013Ch	IFS1<4>	IEC1<4>	IPC5<2:0>
External Interrupt 2	29	00004Eh	00014Eh	IFS1<13>	IEC1<13>	IPC7<6:4>
External Interrupt 3	53	00007Eh	00017Eh	IFS3<5>	IEC3<5>	IPC13<6:4>
External Interrupt 4	54	000080h	000180h	IFS3<6>	IEC3<6>	IPC13<10:8>
FRC Self-Tune	106	0000E8h	0001E8h	IFS6<10>	IEC6<10>	IPC26<10:8>
I2C1 Master Event	17	000036h	000136h	IFS1<1>	IEC1<1>	IPC4<6:4>
I2C1 Slave Event	16	000034h	000134h	IFS1<0>	IEC1<0>	IPC4<2:0>
I2C2 Master Event	50	000078h	000178h	IFS3<2>	IEC3<2>	IPC12<10:8>
I2C2 Slave Event	49	000076h	000176h	IFS3<1>	IEC3<1>	IPC12<6:4>
Input Capture 1	1	000016h	000116h	IFS0<1>	IEC0<1>	IPC0<6:4>
Input Capture 2	5	00001Eh	00011Eh	IFS0<5>	IEC0<5>	IPC1<6:4>
Input Capture 3	37	00005Eh	00015Eh	IFS2<5>	IEC2<5>	IPC9<6:4>
Input Capture 4	38	000060h	000160h	IFS2<6>	IEC2<6>	IPC9<10:8>
Input Capture 5	39	000062h	000162h	IFS2<7>	IEC2<7>	IPC9<14:12>
Input Capture 6	40	000064h	000164h	IFS2<8>	IEC2<8>	IPC10<2:0>
Input Capture 7	22	000040h	000140h	IFS1<6>	IEC1<6>	IPC5<10:8>
Input Capture 8	23	000042h	000142h	IFS1<7>	IEC1<7>	IPC5<14:12>
Input Capture 9	93	0000CEh	0001CEh	IFS5<13>	IEC5<13>	IPC23<6:4>
JTAG	117	0000FEh	0001FEh	IFS7<5>	IEC7<5>	IPC29<6:4>
Input Change Notification (ICN)	19	00003Ah	00013Ah	IFS1<3>	IEC1<3>	IPC4<14:12>
LCD Controller	100	0000DCh	0001DCh	IFS6<4>	IEC6<4>	IPC25<2:0>
High/Low-Voltage Detect (HLVD)	72	0000A4h	0001A4h	IFS4<8>	IEC4<8>	IPC18<2:0>
Op Amp 1	103	0000E2h	0001E2h	IFS6<7>	IEC6<7>	IPC25<14:12>
Op Amp 2	104	0000E4h	0001E4h	IFS6<8>	IEC6<8>	IPC26<2:0>
Output Compare 1	2	000018h	000118h	IFS0<2>	IEC0<2>	IPC0<10:8>
Output Compare 2	6	000020h	000120h	IFS0<6>	IEC0<6>	IPC1<10:8>
Output Compare 3	25	000046h	000146h	IFS1<9>	IEC1<9>	IPC6<6:4>
Output Compare 4	26	000048h	000148h	IFS1<10>	IEC1<10>	IPC6<10:8>
Output Compare 5	41	000066h	000166h	IFS2<9>	IEC2<9>	IPC10<6:4>
Output Compare 6	42	000068h	000168h	IFS2<10>	IEC2<10>	IPC10<10:8>
Output Compare 7	43	00006Ah	00016Ah	IFS2<11>	IEC2<11>	IPC10<14:12>
Output Compare 8	44	00006Ch	00016Ch	IFS2<12>	IEC2<12>	IPC11<2:0>
Output Compare 9	92	0000CCh	0001CCh	IFS5<12>	IEC5<12>	IPC23<2:0>

TABLE 8-2: IMPLEMENTED INTERRUPT VECTORS (CONTINUED)

Interrupt Source	Vector Number	IVT Address	AIVT Address	Interrupt Bit Locations		
				Flag	Enable	Priority
Enhanced Parallel Master Port (EPMP)	45	00006Eh	00016Eh	IFS2<13>	IEC2<13>	IPC11<6:4>
Real-Time Clock and Calendar (RTCC)	62	000090h	000190h	IFS3<14>	IEC3<14>	IPC15<10:8>
SPI1 Error	9	000026h	000126h	IFS0<9>	IEC0<9>	IPC2<6:4>
SPI1 Event	10	000028h	000128h	IFS0<10>	IEC0<10>	IPC2<10:8>
SPI2 Error	32	000054h	000154h	IFS2<0>	IEC2<0>	IPC8<2:0>
SPI2 Event	33	000056h	000156h	IFS2<1>	IEC2<1>	IPC8<6:4>
Timer1	3	00001Ah	00011Ah	IFS0<3>	IEC0<3>	IPC0<14:12>
Timer2	7	000022h	000122h	IFS0<7>	IEC0<7>	IPC1<14:12>
Timer3	8	000024h	000124h	IFS0<8>	IEC0<8>	IPC2<2:0>
Timer4	27	00004Ah	00014Ah	IFS1<11>	IEC1<11>	IPC6<14:12>
Timer5	28	00004Ch	00014Ch	IFS1<12>	IEC1<12>	IPC7<2:0>
UART1 Error	65	000096h	000196h	IFS4<1>	IEC4<1>	IPC16<6:4>
UART1 Receiver	11	00002Ah	00012Ah	IFS0<11>	IEC0<11>	IPC2<14:12>
UART1 Transmitter	12	00002Ch	00012Ch	IFS0<12>	IEC0<12>	IPC3<2:0>
UART2 Error	66	000098h	000198h	IFS4<2>	IEC4<2>	IPC16<10:8>
UART2 Receiver	30	000050h	000150h	IFS1<14>	IEC1<14>	IPC7<10:8>
UART2 Transmitter	31	000052h	000152h	IFS1<15>	IEC1<15>	IPC7<14:12>
UART3 Error	81	0000B6h	0001B6h	IFS5<1>	IEC5<1>	IPC20<6:4>
UART3 Receiver	82	0000B8h	0001B8h	IFS5<2>	IEC5<2>	IPC20<10:8>
UART3 Transmitter	83	0000BAh	0001BAh	IFS5<3>	IEC5<3>	IPC20<14:12>
UART4 Error	87	0000C2h	0001C2h	IFS5<7>	IEC5<7>	IPC21<14:12>
UART4 Receiver	88	0000C4h	0001C4h	IFS5<8>	IEC5<8>	IPC22<2:0>
UART4 Transmitter	89	0000C6h	0001C6h	IFS5<9>	IEC5<9>	IPC22<6:4>
USB	86	0000C0h	0001C0h	IFS5<6>	IEC5<6>	IPC21<10:8>

8.3 Interrupt Control and Status Registers

The PIC24FJ128GC010 family of devices implements a total of 44 registers for the interrupt controller:

- INTCON1
- INTCON2
- IFS0 through IFS7
- IEC0 through IEC7
- IPC0 through IPC13, ICP15, ICP16, ICP18 through ICP23, ICP25, ICP26 and ICP29
- INTTREG

Global interrupt control functions are controlled from INTCON1 and INTCON2. INTCON1 contains the Interrupt Nesting Disable (NSTDIS) bit, as well as the control and status flags for the processor trap sources. The INTCON2 register controls the external interrupt request signal behavior and the use of the Alternate Interrupt Vector Table (AIVT).

The IFSx registers maintain all of the interrupt request flags. Each source of interrupt has a status bit, which is set by the respective peripherals or an external signal and is cleared via software.

The IECx registers maintain all of the interrupt enable bits. These control bits are used to individually enable interrupts from the peripherals or external signals.

The IPCx registers are used to set the Interrupt Priority Level (IPL) for each source of interrupt. Each user interrupt source can be assigned to one of eight priority levels.

The INTTREG register contains the associated interrupt vector number and the new CPU Interrupt Priority Level, which are latched into the Vector Number (VECNUM<6:0>) and the Interrupt Level (ILR<3:0>) bit fields in the INTTREG register. The new Interrupt Priority Level is the priority of the pending interrupt.

The interrupt sources are assigned to the IFSx, IECx and IPCx registers in the order of their vector numbers, as shown in Table 8-2. For example, the INT0 (External Interrupt 0) is shown as having a vector number and a natural order priority of 0. Thus, the INT0IF status bit is found in IFS0<0>, the INT0IE enable bit in IEC0<0> and the INT0IP<2:0> priority bits in the first position of IPC0 (IPC0<2:0>).

Although they are not specifically part of the interrupt control hardware, two of the CPU Control registers contain bits that control interrupt functionality. The ALU STATUS Register (SR) contains the IPL<2:0> bits (SR<7:5>). These indicate the current CPU Interrupt Priority Level. The user can change the current CPU priority level by writing to the IPLx bits.

The CORCON register contains the IPL3 bit, which together with the IPL<2:0> bits, indicates the current CPU priority level. IPL3 is a read-only bit so that trap events cannot be masked by the user software.

The interrupt controller has the Interrupt Controller Test register, INTTREG, which displays the status of the interrupt controller. When an interrupt request occurs, its associated vector number and the new Interrupt Priority Level are latched into INTTREG. This information can be used to determine a specific interrupt source if a generic ISR is used for multiple vectors (such as when ISR remapping is used in bootloader applications) or to check if another interrupt is pending while in an ISR.

All Interrupt registers are described in Register 8-1 through Register 8-46 in the succeeding pages.

REGISTER 8-1: SR: ALU STATUS REGISTER (IN CPU)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	DC(1)
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
IPL2(2,3)	IPL1(2,3)	IPL0(2,3)	RA(1)	N(1)	OV(1)	Z(1)	C(1)
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 15-9 **Unimplemented:** Read as '0'

bit 7-5 **IPL<2:0>:** CPU Interrupt Priority Level Status bits(2,3)

111 = CPU Interrupt Priority Level is 7 (15); user interrupts are disabled
110 = CPU Interrupt Priority Level is 6 (14)
101 = CPU Interrupt Priority Level is 5 (13)
100 = CPU Interrupt Priority Level is 4 (12)
011 = CPU Interrupt Priority Level is 3 (11)
010 = CPU Interrupt Priority Level is 2 (10)
001 = CPU Interrupt Priority Level is 1 (9)
000 = CPU Interrupt Priority Level is 0 (8)

Note 1: See Register 3-1 for the description of the remaining bits (bits 8, 4, 3, 2, 1 and 0) that are not dedicated to interrupt control functions.

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

3: The IPLx Status bits are read-only when NSTDIS (INTCON1<15>) = 1.

REGISTER 8-2: CORCON: CPU CONTROL REGISTER

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

U-0	U-0	U-0	U-0	R/C-0	r-1	U-0	U-0
—	—	—	—	IPL3(1)	r	—	—
bit 7							bit 0

Legend:	r = Reserved bit	C = 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 15-4 **Unimplemented:** Read as '0'

bit 3 **IPL3:** CPU Interrupt Priority Level Status bit(1)
1 = CPU Interrupt Priority Level is greater than 7
0 = CPU Interrupt Priority Level is 7 or less

bit 2 **Reserved:** Read as '1'

bit 1-0 **Unimplemented:** Read as '0'

Note 1: The IPL3 bit is concatenated with the IPL<2:0> bits (SR<7:5>) to form the CPU Interrupt Priority Level; see Register 3-2 for bit description.

REGISTER 8-3: INTCON1: INTERRUPT CONTROL REGISTER 1

R/W-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
NSTDIS	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0
—	—	—	MATHERR	ADDRERR	STKERR	OSCFAIL	—
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 **NSTDIS:** Interrupt Nesting Disable bit
1 = Interrupt nesting is disabled
0 = Interrupt nesting is enabled

bit 14-5 **Unimplemented:** Read as '0'

bit 4 **MATHERR:** Arithmetic Error Trap Status bit
1 = Overflow trap has occurred
0 = Overflow trap has not occurred

bit 3 **ADDRERR:** Address Error Trap Status bit
1 = Address error trap has occurred
0 = Address error trap has not occurred

bit 2 **STKERR:** Stack Error Trap Status bit
1 = Stack error trap has occurred
0 = Stack error trap has not occurred

bit 1 **OSCFAIL:** Oscillator Failure Trap Status bit
1 = Oscillator failure trap has occurred
0 = Oscillator failure trap has not occurred

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

REGISTER 8-4: INTCON2: INTERRUPT CONTROL REGISTER 2

R/W-0	R-0, HSC	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:	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 **ALTIVT:** Enable Alternate Interrupt Vector Table bit
1 = Uses Alternate Interrupt Vector Table
0 = Uses standard (default) Interrupt 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 8-5: IFS0: INTERRUPT FLAG STATUS REGISTER 0

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	DMA1IF	AD1IF	U1TXIF	U1RXIF	SPI1IF	SPF1IF	T3IF
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
T2IF	OC2IF	IC2IF	DMA0IF	T1IF	OC1IF	IC1IF	INT0IF
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 **Unimplemented:** Read as '0'

bit 14 **DMA1IF:** DMA Channel 1 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 13 **AD1IF:** 12-Bit Pipeline A/D Event Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 12 **U1TXIF:** UART1 Transmitter Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 11 **U1RXIF:** UART1 Receiver Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 10 **SPI1IF:** SPI1 Event Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 9 **SPF1IF:** SPI1 Fault Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 8 **T3IF:** Timer3 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 7 **T2IF:** Timer2 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 6 **OC2IF:** Output Compare Channel 2 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 5 **IC2IF:** Input Capture Channel 2 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 4 **DMA0IF:** DMA Channel 0 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 3 **T1IF:** Timer1 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

REGISTER 8-5: IFS0: INTERRUPT FLAG STATUS REGISTER 0 (CONTINUED)

bit 2 **OC1IF:** Output Compare Channel 1 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 1 **IC1IF:** Input Capture Channel 1 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 0 **INT0IF:** External Interrupt 0 Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

REGISTER 8-6: IFS1: INTERRUPT FLAG STATUS REGISTER 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
U2TXIF	U2RXIF	INT2IF	T5IF	T4IF	OC4IF	OC3IF	DMA2IF
bit 15							bit 8

R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IC8IF	IC7IF	—	INT1IF	CNIF	CMIF	MI2C1IF	SI2C1IF
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 **U2TXIF:** UART2 Transmitter Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 14 **U2RXIF:** UART2 Receiver Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 13 **INT2IF:** External Interrupt 2 Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 12 **T5IF:** Timer5 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 11 **T4IF:** Timer4 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 10 **OC4IF:** Output Compare Channel 4 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 9 **OC3IF:** Output Compare Channel 3 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 8 **DMA2IF:** DMA Channel 2 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 7 **IC8IF:** Input Capture Channel 8 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 6 **IC7IF:** Input Capture Channel 7 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 5 **Unimplemented:** Read as '0'

bit 4 **INT1IF:** External Interrupt 1 Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 3 **CNIF:** Input Change Notification Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

REGISTER 8-6: IFS1: INTERRUPT FLAG STATUS REGISTER 1 (CONTINUED)

bit 2 **CMIF:** Comparator Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 1 **MI2C1IF:** Master I2C1 Event Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 0 **SI2C1IF:** Slave I2C1 Event Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

REGISTER 8-7: IFS2: INTERRUPT FLAG STATUS REGISTER 2

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	DMA4IF	PMPIF	OC8IF	OC7IF	OC6IF	OC5IF	IC6IF
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
IC5IF	IC4IF	IC3IF	DMA3IF	—	—	SPI2IF	SPF2IF
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 **Unimplemented:** Read as '0'

bit 14 **DMA4IF:** DMA Channel 4 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 13 **PMPIF:** Parallel Master Port Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 12 **OC8IF:** Output Compare Channel 8 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 11 **OC7IF:** Output Compare Channel 7 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 10 **OC6IF:** Output Compare Channel 6 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 9 **OC5IF:** Output Compare Channel 5 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 8 **IC6IF:** Input Capture Channel 6 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 7 **IC5IF:** Input Capture Channel 5 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 6 **IC4IF:** Input Capture Channel 4 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 5 **IC3IF:** Input Capture Channel 3 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 4 **DMA3IF:** DMA Channel 3 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 3-2 **Unimplemented:** Read as '0'

REGISTER 8-7: IFS2: INTERRUPT FLAG STATUS REGISTER 2 (CONTINUED)

bit 1 **SPI2IF:** SPI2 Event Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 0 **SPF2IF:** SPI2 Fault Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

REGISTER 8-8: IFS3: INTERRUPT FLAG STATUS REGISTER 3

U-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0
—	RTCIF	DMA5IF	—	—	—	—	—
bit 15							bit 8

U-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	U-0
—	INT4IF	INT3IF	—	—	MI2C2IF	SI2C2IF	—
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 **Unimplemented:** Read as '0'

bit 14 **RTCIF:** Real-Time Clock/Calendar Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 13 **DMA5IF:** DMA Channel 5 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 12-7 **Unimplemented:** Read as '0'

bit 6 **INT4IF:** External Interrupt 4 Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 5 **INT3IF:** External Interrupt 3 Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 4-3 **Unimplemented:** Read as '0'

bit 2 **MI2C2IF:** Master I2C2 Event Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 1 **SI2C2IF:** Slave I2C2 Event Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

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

REGISTER 8-9: IFS4: INTERRUPT FLAG STATUS REGISTER 4

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0
DAC2IF	DAC1IF	CTMUIF	—	—	—	—	HLVDIF
bit 15							bit 8

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	U-0
—	—	—	—	CRCIF	U2ERIF	U1ERIF	—
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 **DAC2IF:** DAC Converter 2 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 14 **DAC1IF:** DAC Converter 1 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 13 **CTMUIF:** CTMU Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 12-9 **Unimplemented:** Read as '0'

bit 8 **HLVDIF:** High/Low-Voltage Detect Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 7-4 **Unimplemented:** Read as '0'

bit 3 **CRCIF:** CRC Generator Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 2 **U2ERIF:** UART2 Error Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 1 **U1ERIF:** UART1 Error Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

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

REGISTER 8-10: IFS5: INTERRUPT FLAG STATUS REGISTER 5

U-0	U-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
—	—	IC9IF	OC9IF	—	—	U4TXIF	U4RXIF
bit 15							bit 8

R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	U-0
U4ERIF	USB1IF	—	—	U3TXIF	U3RXIF	U3ERIF	—
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 15-14 **Unimplemented:** Read as '0'

bit 13 **IC9IF:** Input Capture Channel 9 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 12 **OC9IF:** Output Compare Channel 9 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 11-10 **Unimplemented:** Read as '0'

bit 9 **U4TXIF:** UART4 Transmitter Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 8 **U4RXIF:** UART4 Receiver Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 7 **U4ERIF:** UART4 Error Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 6 **USB1IF:** USB1 (USB OTG) Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 5-4 **Unimplemented:** Read as '0'

bit 3 **U3TXIF:** UART3 Transmitter Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 2 **U3RXIF:** UART3 Receiver Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 1 **U3ERIF:** UART3 Error Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

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

REGISTER 8-11: IFS6: INTERRUPT FLAG STATUS REGISTER 6

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	FSTIF	SDA1IF	AMP2IF
bit 15							bit 8

R/W-0	U-0	U-0	R/W-0	U-0	U-0	U-0	U-0
AMP1IF	—	—	LCDIF	—	—	—	—
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 **FSTIF:** FRC Self-Tune Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 9 **SDA1IF:** Sigma-Delta A/D Converter Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 8 **AMP2IF:** Op Amp 2 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 7 **AMP1IF:** Op Amp 1 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 6-5 **Unimplemented:** Read as '0'

bit 4 **LCDIF:** LCD Controller Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

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

REGISTER 8-12: IFS7: INTERRUPT FLAG STATUS REGISTER 7

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

U-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
—	—	JTAGIF	—	—	—	—	—
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 15-6 **Unimplemented:** Read as ‘0’

bit 5 **JTAGIF:** JTAG Controller Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 4-0 **Unimplemented:** Read as ‘0’

REGISTER 8-13: IEC0: INTERRUPT ENABLE CONTROL REGISTER 0

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	DMA1IE	AD1IE	U1TXIE	U1RXIE	SPI1IE	SPF1IE	T3IE
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
T2IE	OC2IE	IC2IE	DMA0IE	T1IE	OC1IE	IC1IE	INT0IE
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 **Unimplemented:** Read as '0'

bit 14 **DMA1IE:** DMA Channel 1 Interrupt Flag Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 13 **AD1IE:** 12-Bit Pipeline A/D Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 12 **U1TXIE:** UART1 Transmitter Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 11 **U1RXIE:** UART1 Receiver Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 10 **SPI1IE:** SPI1 Transfer Complete Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 9 **SPF1IE:** SPI1 Fault Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 8 **T3IE:** Timer3 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 7 **T2IE:** Timer2 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 6 **OC2IE:** Output Compare Channel 2 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 5 **IC2IE:** Input Capture Channel 2 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 4 **DMA0IE:** DMA Channel 0 Interrupt Flag Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 3 **T1IE:** Timer1 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

REGISTER 8-13: IEC0: INTERRUPT ENABLE CONTROL REGISTER 0 (CONTINUED)

bit 2 **OC1IE:** Output Compare Channel 1 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 1 **IC1IE:** Input Capture Channel 1 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 0 **INT0IE:** External Interrupt 0 Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

REGISTER 8-14: IEC1: INTERRUPT ENABLE CONTROL REGISTER 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
U2TXIE	U2RXIE	INT2IE(1)	T5IE	T4IE	OC4IE	OC3IE	DMA2IE
bit 15							bit 8

R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IC8IE	IC7IE	—	INT1IE(1)	CNIE	CMIE	MI2C1IE	SI2C1IE
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 **U2TXIE:** UART2 Transmitter Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 14 **U2RXIE:** UART2 Receiver Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 13 **INT2IE:** External Interrupt 2 Enable bit(1)
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 12 **T5IE:** Timer5 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 11 **T4IE:** Timer4 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 10 **OC4IE:** Output Compare Channel 4 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 9 **OC3IE:** Output Compare Channel 3 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 8 **DMA2IE:** DMA Channel 2 Interrupt Flag Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 7 **IC8IE:** Input Capture Channel 8 Interrupt Flag Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 6 **IC7IE:** Input Capture Channel 7 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 5 **Unimplemented:** Read as '0'

Note 1: If an external interrupt is enabled, the interrupt input must also be configured to an available RPx or RPIx pin. See **Section 11.4 "Peripheral Pin Select (PPS)"** for more information.

REGISTER 8-14: IEC1: INTERRUPT ENABLE CONTROL REGISTER 1 (CONTINUED)

bit 4 **INT1IE:** External Interrupt 1 Enable bit(1)
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 3 **CNIE:** Input Change Notification Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 2 **CMIE:** Comparator Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 1 **MI2C1IE:** Master I2C1 Event Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 0 **SI2C1IE:** Slave I2C1 Event Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

Note 1: If an external interrupt is enabled, the interrupt input must also be configured to an available RPx or RPIx pin. See **Section 11.4 "Peripheral Pin Select (PPS)"** for more information.

REGISTER 8-15: IEC2: INTERRUPT ENABLE CONTROL REGISTER 2

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	DMA4IE	PMPIE	OC8IE	OC7IE	OC6IE	OC5IE	IC6IE
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
IC5IE	IC4IE	IC3IE	DMA3IE	—	—	SPI2IE	SPF2IE
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 **Unimplemented:** Read as '0'

bit 14 **DMA4IE:** DMA Channel 4 Interrupt Flag Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 13 **PMPIE:** Parallel Master Port Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 12 **OC8IE:** Output Compare Channel 8 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 11 **OC7IE:** Output Compare Channel 7 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 10 **OC6IE:** Output Compare Channel 6 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 9 **OC5IE:** Output Compare Channel 5 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 8 **IC6IE:** Input Capture Channel 6 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 7 **IC5IE:** Input Capture Channel 5 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 6 **IC4IE:** Input Capture Channel 4 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 5 **IC3IE:** Input Capture Channel 3 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 4 **DMA3IF:** DMA Channel 3 Interrupt Flag Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 3-2 **Unimplemented:** Read as '0'

REGISTER 8-15: IEC2: INTERRUPT ENABLE CONTROL REGISTER 2 (CONTINUED)

bit 1 **SPI2IE:** SPI2 Event Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 0 **SPF2IE:** SPI2 Fault Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

REGISTER 8-16: IEC3: INTERRUPT ENABLE CONTROL REGISTER 3

U-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0
—	RTCIE	DMA5IE	—	—	—	—	—
bit 15							bit 8

U-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	U-0
—	INT4IE(1)	INT3IE(1)	—	—	MI2C2IE	SI2C2IE	—
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 **Unimplemented:** Read as '0'

bit 14 **RTCIE:** Real-Time Clock/Calendar Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 13 **DMA5IE:** DMA Channel 5 Interrupt Flag Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 12-7 **Unimplemented:** Read as '0'

bit 6 **INT4IE:** External Interrupt 4 Enable bit(1)
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 5 **INT3IE:** External Interrupt 3 Enable bit(1)
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 4-3 **Unimplemented:** Read as '0'

bit 2 **MI2C2IE:** Master I2C2 Event Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 1 **SI2C2IE:** Slave I2C2 Event Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

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

Note 1: If an external interrupt is enabled, the interrupt input must also be configured to an available RPx or RPIx pin. See **Section 11.4 "Peripheral Pin Select (PPS)"** for more information.

REGISTER 8-17: IEC4: INTERRUPT ENABLE CONTROL REGISTER 4

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0
DAC2IE	DAC1IE	CTMUIE	—	—	—	—	HLVDIE
bit 15							bit 8

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	U-0
—	—	—	—	CRCIE	U2ERIE	U1ERIE	—
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 **DAC2IE:** DAC Converter 2 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 14 **DAC1IE:** DAC Converter 1 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 13 **CTMUIE:** CTMU Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 12-9 **Unimplemented:** Read as '0'

bit 8 **HLVDIE:** High/Low-Voltage Detect Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 7-4 **Unimplemented:** Read as '0'

bit 3 **CRCIE:** CRC Generator Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 2 **U2ERIE:** UART2 Error Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 1 **U1ERIE:** UART1 Error Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

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

REGISTER 8-18: IEC5: INTERRUPT ENABLE CONTROL REGISTER 5

U-0	U-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
—	—	IC9IE	OC9IE	—	—	U4TXIE	U4RXIE
bit 15							bit 8

R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	U-0
U4ERIE	USBIE	—	—	U3TXIE	U3RXIE	U3ERIE	—
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 15-14 **Unimplemented:** Read as '0'

bit 13 **IC9IE:** Input Capture Channel 9 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 12 **OC9IE:** Output Compare Channel 9 Enable Status bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 11-10 **Unimplemented:** Read as '0'

bit 9 **U4TXIE:** UART4 Transmitter Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 8 **U4RXIE:** UART4 Receiver Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 7 **U4ERIE:** UART4 Error Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 6 **USBIE:** USB1 (USB OTG) Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 5-4 **Unimplemented:** Read as '0'

bit 3 **U3TXIE:** UART3 Transmitter Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 2 **U3RXIE:** UART3 Receiver Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 1 **U3ERIE:** UART3 Error Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

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

REGISTER 8-19: IEC6: INTERRUPT ENABLE CONTROL REGISTER 6

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	FSTIE	SDA1IE	AMP2IE
bit 15							bit 8

R/W-0	U-0	U-0	R/W-0	U-0	U-0	U-0	U-0
AMP1IE	—	—	LCDIE	—	—	—	—
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 **FSTIE:** FRC Self-Tune Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 9 **SDA1IE:** Sigma-Delta A/D Converter Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 8 **AMP2IE:** Op Amp 2 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 7 **AMP1IE:** Op Amp 1 Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 6-5 **Unimplemented:** Read as '0'

bit 4 **LCDIE:** LCD Controller Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

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

REGISTER 8-20: IEC7: INTERRUPT ENABLE CONTROL REGISTER 7

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

U-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
—	—	JTAGIE	—	—	—	—	—
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 15-6 **Unimplemented:** Read as ‘0’

bit 5 **JTAGIE:** JATG Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 4-0 **Unimplemented:** Read as ‘0’

REGISTER 8-21: IPC0: INTERRUPT PRIORITY CONTROL REGISTER 0

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	T1IP2	T1IP1	T1IP0	—	OC1IP2	OC1IP1	OC1IP0
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
—	IC1IP2	IC1IP1	IC1IP0	—	INT0IP2	INT0IP1	INT0IP0
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 **Unimplemented:** Read as ‘0’

bit 14-12 **T1IP<2:0>:** Timer1 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 11 **Unimplemented:** Read as ‘0’

bit 10-8 **OC1IP<2:0>:** Output Compare Channel 1 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 7 **Unimplemented:** Read as ‘0’

bit 6-4 **IC1IP<2:0>:** Input Capture Channel 1 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 **INT0IP<2:0>:** External Interrupt 0 Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

REGISTER 8-22: IPC1: INTERRUPT PRIORITY CONTROL REGISTER 1

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	T2IP2	T2IP1	T2IP0	—	OC2IP2	OC2IP1	OC2IP0
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
—	IC2IP2	IC2IP1	IC2IP0	—	DMA0IP2	DMA0IP1	DMA0IP0
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 **Unimplemented:** Read as ‘0’

bit 14-12 **T2IP<2:0>:** Timer2 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 11 **Unimplemented:** Read as ‘0’

bit 10-8 **OC2IP<2:0>:** Output Compare Channel 2 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 7 **Unimplemented:** Read as ‘0’

bit 6-4 **IC2IP<2:0>:** Input Capture Channel 2 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 **DMA0IP<2:0>:** DMA Channel 0 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

REGISTER 8-23: IPC2: INTERRUPT PRIORITY CONTROL REGISTER 2

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	U1RXIP2	U1RXIP1	U1RXIP0	—	SPI1IP2	SPI1IP1	SPI1IP0
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
—	SPF1IP2	SPF1IP1	SPF1IP0	—	T3IP2	T3IP1	T3IP0
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 **Unimplemented:** Read as ‘0’

bit 14-12 **U1RXIP<2:0>:** UART1 Receiver Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 11 **Unimplemented:** Read as ‘0’

bit 10-8 **SPI1IP<2:0>:** SPI1 Event Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 7 **Unimplemented:** Read as ‘0’

bit 6-4 **SPF1IP<2:0>:** SPI1 Fault 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 **T3IP<2:0>:** Timer3 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

REGISTER 8-24: IPC3: INTERRUPT PRIORITY CONTROL REGISTER 3

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	DMA1IP2	DMA1IP1	DMA1IP0
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 15-11 **Unimplemented:** Read as ‘0’

bit 10-8 **DMA1IP<2:0>:** DMA Channel 1 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 7 **Unimplemented:** Read as ‘0’

bit 6-4 **AD1IP<2:0>:** 12-Bit Pipeline A/D 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 8-25: IPC4: INTERRUPT PRIORITY CONTROL REGISTER 4

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	CNIP2	CNIP1	CNIP0	—	CMIP2	CMIP1	CMIP0
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
—	MI2C1IP2	MI2C1IP1	MI2C1IP0	—	SI2C1IP2	SI2C1IP1	SI2C1IP0
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 **Unimplemented:** Read as ‘0’

bit 14-12 **CNIP<2:0>:** Input Change Notification Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 11 **Unimplemented:** Read as ‘0’

bit 10-8 **CMIP<2:0>:** Comparator Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 7 **Unimplemented:** Read as ‘0’

bit 6-4 **MI2C1IP<2:0>:** Master I2C1 Event 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 **SI2C1IP<2:0>:** Slave I2C1 Event Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

REGISTER 8-26: IPC5: INTERRUPT PRIORITY CONTROL REGISTER 5

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	IC8IP2	IC8IP1	IC8IP0	—	IC7IP2	IC7IP1	IC7IP0
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	INT1IP2	INT1IP1	INT1IP0
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 **Unimplemented:** Read as ‘0’

bit 14-12 **IC8IP<2:0>:** Input Capture Channel 8 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 11 **Unimplemented:** Read as ‘0’

bit 10-8 **IC7IP<2:0>:** Input Capture Channel 7 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 7-3 **Unimplemented:** Read as ‘0’

bit 2-0 **INT1IP<2:0>:** External Interrupt 1 Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

REGISTER 8-27: IPC6: INTERRUPT PRIORITY CONTROL REGISTER 6

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	T4IP2	T4IP1	T4IP0	—	OC4IP2	OC4IP1	OC4IP0
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
—	OC3IP2	OC3IP1	OC3IP0	—	DMA2IP2	DMA2IP1	DMA2IP0
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 **Unimplemented:** Read as '0'

bit 14-12 **T4IP<2:0>:** Timer4 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

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

bit 10-8 **OC4IP<2:0>:** Output Compare Channel 4 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

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

bit 6-4 **OC3IP<2:0>:** Output Compare Channel 3 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 **DMA2IP<2:0>:** DMA Channel 2 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

REGISTER 8-28: IPC7: INTERRUPT PRIORITY CONTROL REGISTER 7

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	U2TXIP2	U2TXIP1	U2TXIP0	—	U2RXIP2	U2RXIP1	U2RXIP0
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
—	INT2IP2	INT2IP1	INT2IP0	—	T5IP2	T5IP1	T5IP0
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 **Unimplemented:** Read as ‘0’

bit 14-12 **U2TXIP<2:0>:** UART2 Transmitter Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 11 **Unimplemented:** Read as ‘0’

bit 10-8 **U2RXIP<2:0>:** UART2 Receiver Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 7 **Unimplemented:** Read as ‘0’

bit 6-4 **INT2IP<2:0>:** External Interrupt 2 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 **T5IP<2:0>:** Timer5 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

REGISTER 8-29: IPC8: INTERRUPT PRIORITY CONTROL REGISTER 8

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
—	SPI2IP2	SPI2IP1	SPI2IP0	—	SPF2IP2	SPF2IP1	SPF2IP0
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 **SPI2IP<2:0>:** SPI2 Event 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 **SPF2IP<2:0>:** SPI2 Fault Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

REGISTER 8-30: IPC9: INTERRUPT PRIORITY CONTROL REGISTER 9

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	IC5IP2	IC5IP1	IC5IP0	—	IC4IP2	IC4IP1	IC4IP0
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
—	IC3IP2	IC3IP1	IC3IP0	—	DMA3IP2	DMA3IP1	DMA3IP0
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 **Unimplemented:** Read as ‘0’

bit 14-12 **IC5IP<2:0>:** Input Capture Channel 5 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 11 **Unimplemented:** Read as ‘0’

bit 10-8 **IC4IP<2:0>:** Input Capture Channel 4 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 7 **Unimplemented:** Read as ‘0’

bit 6-4 **IC3IP<2:0>:** Input Capture Channel 3 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 **DMA3IP<2:0>:** DMA Channel 3 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

REGISTER 8-31: IPC10: INTERRUPT PRIORITY CONTROL REGISTER 10

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	OC7IP2	OC7IP1	OC7IP0	—	OC6IP2	OC6IP1	OC6IP0
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
—	OC5IP2	OC5IP1	OC5IP0	—	IC6IP2	IC6IP1	IC6IP0
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 **Unimplemented:** Read as ‘0’

bit 14-12 **OC7IP<2:0>:** Output Compare Channel 7 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 11 **Unimplemented:** Read as ‘0’

bit 10-8 **OC6IP<2:0>:** Output Compare Channel 6 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 7 **Unimplemented:** Read as ‘0’

bit 6-4 **OC5IP<2:0>:** Output Compare Channel 5 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 **IC6IP<2:0>:** Input Capture Channel 6 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

REGISTER 8-32: IPC11: INTERRUPT PRIORITY CONTROL REGISTER 11

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	DMA4IP2	DMA4IP1	DMA4IP0
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
—	PMPIP2	PMPIP1	PMPIP0	—	OC8IP2	OC8IP1	OC8IP0
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 **DMA4IP<2:0>:** DMA Channel 4 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 7 **Unimplemented:** Read as ‘0’

bit 6-4 **PMPIP<2:0>:** Parallel Master Port 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 **OC8IP<2:0>:** Output Compare Channel 8 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

REGISTER 8-33: IPC12: INTERRUPT PRIORITY CONTROL REGISTER 12

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	MI2C2IP2	MI2C2IP1	MI2C2IP0
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	SI2C2IP2	SI2C2IP1	SI2C2IP0	—	—	—	—
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 **MI2C2IP<2:0>:** Master I2C2 Event Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

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

bit 6-4 **SI2C2IP<2:0>:** Slave I2C2 Event Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

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

REGISTER 8-34: IPC13: INTERRUPT PRIORITY CONTROL REGISTER 13

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	INT4IP2	INT4IP1	INT4IP0
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	INT3IP2	INT3IP1	INT3IP0	—	—	—	—
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 **INT4IP<2:0>:** External Interrupt 4 Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 7 **Unimplemented:** Read as ‘0’

bit 6-4 **INT3IP<2:0>:** External Interrupt 3 Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 3-0 **Unimplemented:** Read as ‘0’

REGISTER 8-35: 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	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	DMA5IP2	DMA5IP1	DMA5IP0	—	—	—	—
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 **Unimplemented:** Read as ‘0’

bit 6-4 **DMA5IP<2:0>:** DMA Channel 5 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 3-0 **Unimplemented:** Read as ‘0’

REGISTER 8-36: IPC16: INTERRUPT PRIORITY CONTROL REGISTER 16

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	CRCIP2	CRCIP1	CRCIP0	—	U2ERIP2	U2ERIP1	U2ERIP0
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	U1ERIP2	U1ERIP1	U1ERIP0	—	—	—	—
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 **Unimplemented:** Read as ‘0’

bit 14-12 **CRCIP<2:0>:** CRC Generator Error Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 11 **Unimplemented:** Read as ‘0’

bit 10-8 **U2ERIP<2:0>:** UART2 Error Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 7 **Unimplemented:** Read as ‘0’

bit 6-4 **U1ERIP<2:0>:** UART1 Error Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 3-0 **Unimplemented:** Read as ‘0’

REGISTER 8-37: IPC18: INTERRUPT PRIORITY CONTROL REGISTER 18

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-1	R/W-0	R/W-0
—	—	—	—	—	HLVDIP2	HLVDIP1	HLVDIP0
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 15-3 **Unimplemented:** Read as ‘0’

bit 2-0 **HLVDIP<2:0>:** High/Low-Voltage Detect Interrupt Priority bits

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

REGISTER 8-38: IPC19: INTERRUPT PRIORITY CONTROL REGISTER 19

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	DAC2IP2	DAC2IP1	DAC2IP0	—	DAC1IP2	DAC1IP1	DAC1IP0
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	CTMUIP2	CTMUIP1	CTMUIP0	—	—	—	—
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 **Unimplemented:** Read as ‘0’

bit 14-12 **DAC2IP<2:0>:** DAC Converter 2 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 11 **Unimplemented:** Read as ‘0’

bit 10-8 **DAC1IP<2:0>:** DAC Converter 1 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 7 **Unimplemented:** Read as ‘0’

bit 6-4 **CTMUIP<2:0>:** CTMU Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 3-0 **Unimplemented:** Read as ‘0’

REGISTER 8-39: IPC20: INTERRUPT PRIORITY CONTROL REGISTER 20

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	U3TXIP2	U3TXIP1	U3TXIP0	—	U3RXIP2	U3RXIP1	U3RXIP0
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	U3ERIP2	U3ERIP1	U3ERIP0	—	—	—	—
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 **Unimplemented:** Read as ‘0’

bit 14-12 **U3TXIP<2:0>:** UART3 Transmitter Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 11 **Unimplemented:** Read as ‘0’

bit 10-8 **U3RXIP<2:0>:** UART3 Receiver Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 7 **Unimplemented:** Read as ‘0’

bit 6-4 **U3ERIP<2:0>:** UART3 Error Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 3-0 **Unimplemented:** Read as ‘0’

REGISTER 8-40: IPC21: INTERRUPT PRIORITY CONTROL REGISTER 21

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	U4ERIP2	U4ERIP1	U4ERIP0	—	USB1IP2	USB1IP1	USB1IP0
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 **Unimplemented:** Read as ‘0’

bit 14-12 **U4ERIP<2:0>:** UART4 Error Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 11 **Unimplemented:** Read as ‘0’

bit 10-8 **USB1IP<2:0>:** USB1 (USB OTG) 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 8-41: IPC22: INTERRUPT PRIORITY CONTROL REGISTER 22

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
—	U4TXIP2	U4TXIP1	U4TXIP0	—	U4RXIP2	U4RXIP1	U4RXIP0
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 **U4TXIP<2:0>:** UART4 Transmitter 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 **U4RXIP<2:0>:** UART4 Receiver Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

REGISTER 8-42: IPC23: INTERRUPT PRIORITY CONTROL REGISTER 23

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
—	IC9IP2	IC9IP1	IC9IP0	—	OC9IP2	OC9IP1	OC9IP0
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 **IC9IP<2:0>:** Input Capture Channel 9 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 **OC9IP<2:0>:** Output Compare Channel 9 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

REGISTER 8-43: IPC25: INTERRUPT PRIORITY CONTROL REGISTER 25

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	AMP1IP2	AMP1IP1	AMP1IP0	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	LCDIP2	LCDIP1	LCDIP0
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 **Unimplemented:** Read as ‘0’

bit 14-12 **AMP1IP<2:0>:** Op Amp 1 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 11-3 **Unimplemented:** Read as ‘0’

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

REGISTER 8-44: IPC26: INTERRUPT PRIORITY CONTROL REGISTER 26

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	FSTIP2	FSTIP1	FSTIP0
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
—	SDA1IP2	SDA1IP1	SDA1IP0	—	AMP2IP2	AMP2IP1	AMP2IP0
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 **FSTIP<2:0>:** FRC Self-Tune Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

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

bit 6-4 **SDA1IP<2:0>:** Sigma-Delta A/D Converter 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 **AMP2IP<2:0>:** Op Amp 2 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

REGISTER 8-45: IPC29: INTERRUPT PRIORITY CONTROL REGISTER 29

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	U-0	U-0	U-0
—	JTAGIP2	JTAGIP1	JTAGIP0	—	—	—	—
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 **JTAGIP<2:0>:** JTAG Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 3-0 **Unimplemented:** Read as ‘0’

REGISTER 8-46: INTTREG: INTERRUPT CONTROLLER TEST REGISTER

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

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

Legend:	r = Reserved 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 **Reserved:** Maintain as '0'

bit 13 **VHOLD:** Vector Number Capture Configuration bit
1 = VECNUM<6:0> contain the value of the highest priority pending interrupt
0 = VECNUM<6:0> 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<6:0>:** Vector Number of Pending Interrupt or Last Acknowledged Interrupt bits
When VHOLD = 1:
Indicates the vector number (from 0 to 118) of the last interrupt to occur.
When VHOLD = 0:
Indicates the vector number (from 0 to 118) of the interrupt request currently being handled.

8.4 Interrupt Setup Procedures

8.4.1 INITIALIZATION

To configure an interrupt source:

1. Set the NSTDIS (INTCON1<15>) control bit if nested interrupts are not desired.
2. Select the user-assigned priority level for the interrupt source by writing the control bits in the appropriate IPCx register. The priority level will depend on the specific application and type of interrupt source. If multiple priority levels are not desired, the IPCx register control bits, for all enabled interrupt sources, may be programmed to the same non-zero value.

Note:	At a device Reset, the IPCx registers are initialized, such that all user interrupt sources are assigned to Priority Level 4.

3. Clear the interrupt flag status bit associated with the peripheral in the associated IFSx register.
4. Enable the interrupt source by setting the interrupt enable control bit associated with the source in the appropriate IECx register.

8.4.2 INTERRUPT SERVICE ROUTINE (ISR)

The method that is used to declare an Interrupt Service Routine (ISR) and initialize the IVT with the correct vector address will depend on the programming language (i.e., 'C' or assembler), and the language development toolsuite that is used to develop the application. In general, the user must clear the interrupt flag in the appropriate IFSx register for the source of the interrupt that the ISR handles; otherwise, the ISR will be re-entered immediately after exiting the routine. If the ISR is coded in assembly language, it must be terminated using a `RETFIE` instruction to unstack the saved PC value, SRL value and old CPU priority level.

8.4.3 TRAP SERVICE ROUTINE (TSR)

A Trap Service Routine (TSR) is coded like an ISR, except that the appropriate trap status flag in the INTCON1 register must be cleared to avoid re-entry into the TSR.

8.4.4 INTERRUPT DISABLE

All user interrupts can be disabled using the following procedure:

1. Push the current SR value onto the software stack using the `PUSH` instruction.
2. Force the CPU to Priority Level 7 by inclusive ORing the value, 0Eh, with SRL.

To enable user interrupts, the `POP` instruction may be used to restore the previous SR value.

Note that only user interrupts with a priority level of 7 or less can be disabled. Trap sources (Levels 8-15) cannot be disabled.

The `DISI` instruction provides a convenient way to disable interrupts of Priority Levels 1-6 for a fixed period of time. Level 7 interrupt sources are not disabled by the `DISI` instruction.

9.0 OSCILLATOR CONFIGURATION

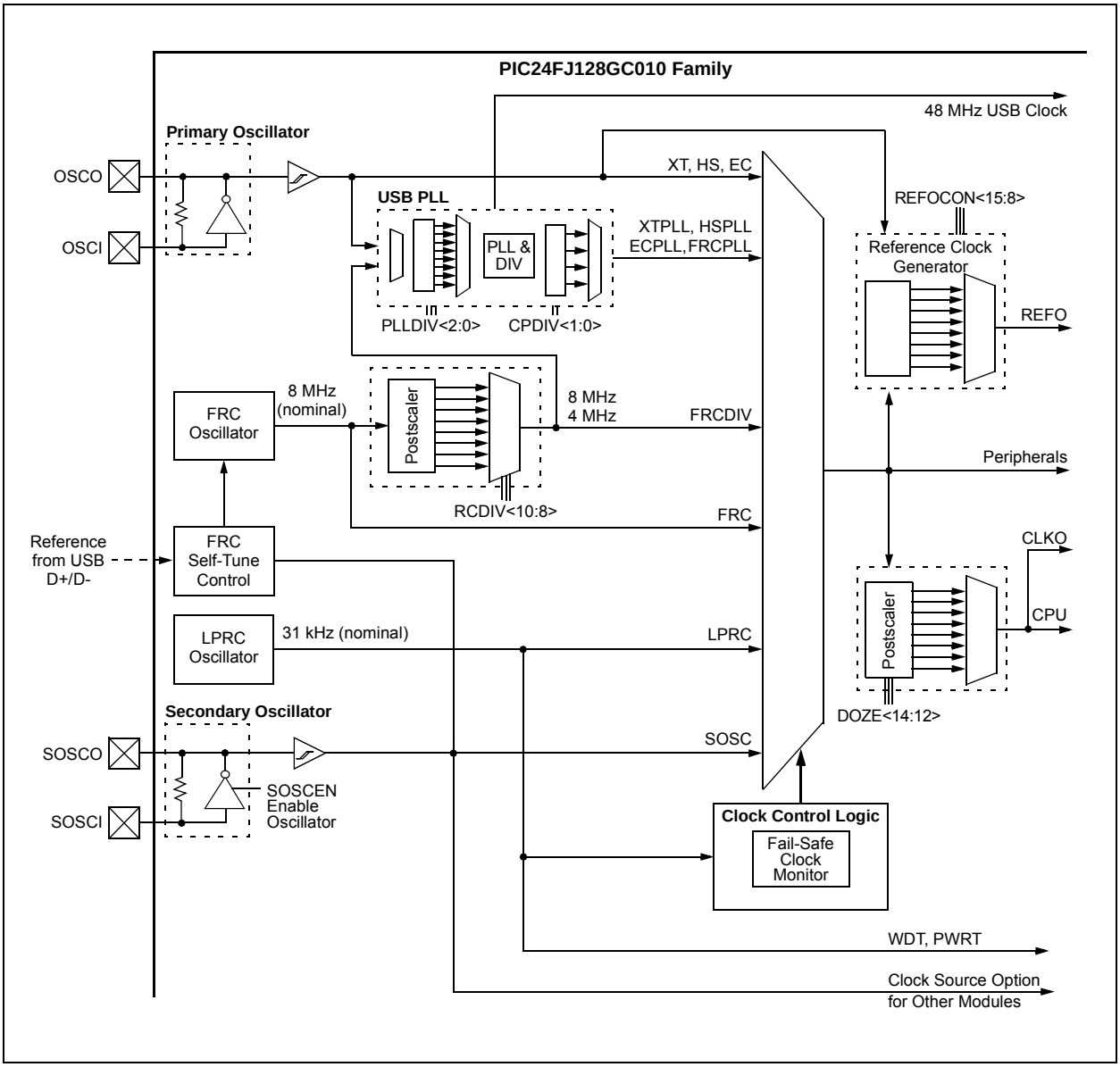
Note: This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the *"PIC24F Family Reference Manual"*, **"Oscillator"** (DS39700).

The oscillator system for PIC24FJ128GC010 family devices has the following features:

- A total of four external and internal oscillator options as clock sources, providing 11 different clock modes
- An on-chip USB PLL block to provide a stable 48 MHz clock for the USB module, as well as a range of frequency options for the system clock
- Software-controllable switching between various clock sources
- Software-controllable postscaler for selective clocking of CPU for system power savings
- A Fail-Safe Clock Monitor (FSCM) that detects clock failure and permits safe application recovery or shutdown
- A separate and independently configurable system clock output for synchronizing external hardware

A simplified diagram of the oscillator system is shown in Figure 9-1.

FIGURE 9-1: PIC24FJ128GC010 FAMILY CLOCK DIAGRAM



9.1 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 USB PLL block, which generates both the USB module clock and a separate system clock from the 96 MHZ PLL. Refer to **Section 9.6 "Oscillator Modes and USB Operation"** 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.

9.2 Initial Configuration on POR

The oscillator source (and operating mode) that is used at a device Power-on Reset 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 34.1 "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 Power-on Reset. The FRC Primary Oscillator with Postscaler (FRCDIV) is the default (unprogrammed) selection. The Secondary Oscillator (SOSC), 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 9-1.

9.2.1 CLOCK SWITCHING MODE CONFIGURATION BITS

The FCKSM<1:0> 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 9-1: 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
(Reserved)	Internal	xx	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	
Primary Oscillator (HS)	Primary	10	010	
Primary Oscillator (XT)	Primary	01	010	
Primary Oscillator (EC)	Primary	00	010	
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.

9.3 Control Registers

The operation of the oscillator is controlled by three Special Function Registers:

- OSCCON
- CLKDIV
- OSCTUN

The OSCCON register (Register 9-1) is the main control register for the oscillator. It controls clock source switching and allows the monitoring of clock sources. OSCCON is protected by a write lock to prevent inadvertent clock switches. See **Section 9.4 "Clock Switching Operation"** for more information.

The CLKDIV register (Register 9-2) controls the features associated with Doze mode, as well as the postscaler for the FRC Oscillator.

The OSCTUN register (Register 9-3) allows the user to fine-tune the FRC Oscillator over a range of approximately ±1.5%. It also controls the FRC self-tuning features, described in **Section 9.5 "FRC Active Clock Tuning"**.

REGISTER 9-1: OSCCON: OSCILLATOR CONTROL REGISTER

U-0	R-0	R-0	R-0	U-0	R/W-x(1)	R/W-x(1)	R/W-x(1)
—	COSC2	COSC1	COSC0	—	NOSC2	NOSC1	NOSC0
bit 15							bit 8

R/SO-0	R/W-0	R-0(3)	U-0	R/CO-0	R/W-0	R/W-0	R/W-0
CLKLOCK	IOLOCK(2)	LOCK	—	CF	POSCEN	SOSCEN	OSWEN
bit 7							bit 0

Legend:	CO = Clearable Only bit	SO = Settable Only 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 **COSC<2:0>:** Current Oscillator Selection bits
111 = Fast RC Oscillator with Postscaler (FRCDIV)
110 = Reserved
101 = Low-Power RC Oscillator (LPRC)
100 = Secondary Oscillator (SOSC)
011 = Primary Oscillator with PLL module (XTPLL, HSPLL, ECPLL)
010 = Primary Oscillator (XT, HS, EC)
001 = Fast RC Oscillator with Postscaler and PLL module (FRCPLL)(4)
000 = Fast RC Oscillator (FRC)

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

bit 10-8 **NOSC<2:0>:** New Oscillator Selection bits(1)
111 = Fast RC Oscillator with Postscaler (FRCDIV)
110 = Reserved
101 = Low-Power RC Oscillator (LPRC)
100 = Secondary Oscillator (SOSC)
011 = Primary Oscillator with PLL module (XTPLL, HSPLL, ECPLL)
010 = Primary Oscillator (XT, HS, EC)
001 = Fast RC Oscillator with Postscaler and PLL module (FRCPLL)(4)
000 = Fast RC Oscillator (FRC)

Note 1: Reset values for these bits are determined by the FNOSCx Configuration bits.
2: The state of the IOLOCK bit can only be changed once an unlocking sequence has been executed. In addition, if the IOL1WAY Configuration bit is '1', once the IOLOCK bit is set, it cannot be cleared.
3: This bit also resets to '0' during any valid clock switch or whenever a non-PLL Clock mode is selected.
4: The default divisor of the postscaler is 2, which will generate a 4 MHz clock to the PLL module.

REGISTER 9-1: OSCCON: OSCILLATOR CONTROL REGISTER (CONTINUED)

bit 7 **CLKLOCK:** Clock Selection Lock Enable bit
If FSCM is enabled (FCKSM1 = 1):
1 = Clock and PLL selections are locked
0 = Clock and PLL selections are not locked and may be modified by setting the OSWEN bit
If FSCM is disabled (FCKSM1 = 0):
Clock and PLL selections are never locked and may be modified by setting the OSWEN bit.

bit 6 **IOLOCK:** I/O Lock Enable bit(2)
1 = I/O lock is active
0 = I/O lock is not active

bit 5 **LOCK:** PLL Lock Status bit(3)
1 = PLL module is in lock or PLL module start-up timer is satisfied
0 = PLL module is out of lock, PLL start-up timer is running or PLL is disabled

bit 4 **Unimplemented:** Read as '0'

bit 3 **CF:** Clock Fail Detect bit
1 = FSCM has detected a clock failure
0 = No clock failure has been detected

bit 2 **POSCEN:** Primary Oscillator Sleep Enable bit
1 = Primary Oscillator continues to operate during Sleep mode
0 = Primary Oscillator is disabled during Sleep mode

bit 1 **SOSCEN:** 32 kHz Secondary Oscillator (SOSC) Enable bit
1 = Enables Secondary Oscillator
0 = Disables Secondary Oscillator

bit 0 **OSWEN:** Oscillator Switch Enable bit
1 = Initiates an oscillator switch to a clock source specified by the NOSC<2:0> bits
0 = Oscillator switch is complete

Note 1: Reset values for these bits are determined by the FNOSCx Configuration bits.
2: The state of the IOLOCK bit can only be changed once an unlocking sequence has been executed. In addition, if the IOL1WAY Configuration bit is '1', once the IOLOCK bit is set, it cannot be cleared.
3: This bit also resets to '0' during any valid clock switch or whenever a non-PLL Clock mode is selected.
4: The default divisor of the postscaler is 2, which will generate a 4 MHz clock to the PLL module.

REGISTER 9-2: CLKDIV: CLOCK DIVIDER REGISTER

R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-1
ROI	DOZE2	DOZE1	DOZE0	DOZEN(1)	RCDIV2	RCDIV1	RCDIV0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0
CPDIV1	CPDIV0	PLLEN	—	—	—	—	—
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 **ROI:** Recover on Interrupt bit
1 = Interrupts clear the DOZEN bit and reset the CPU peripheral clock ratio to 1:1
0 = Interrupts have no effect on the DOZEN bit

bit 14-12 **DOZE<2:0>:** CPU Peripheral Clock Ratio Select bits
111 = 1:128
110 = 1:64
101 = 1:32
100 = 1:16
011 = 1:8 (default)
010 = 1:4
001 = 1:2
000 = 1:1

bit 11 **DOZEN:** Doze Enable bit(1)
1 = DOZE<2:0> bits specify the CPU peripheral clock ratio
0 = CPU peripheral clock ratio is set to 1:1

bit 10-8 **RCDIV<2:0>:** FRC Postscaler Select bits
111 = 31.25 kHz (divide-by-256)
110 = 125 kHz (divide-by-64)
101 = 250 kHz (divide-by-32)
100 = 500 kHz (divide-by-16)
011 = 1 MHz (divide-by-8)
010 = 2 MHz (divide-by-4)
001 = 4 MHz (divide-by-2) (default)
000 = 8 MHz (divide-by-1)

bit 7-6 **CPDIV<1:0>:** System Clock Select bits (postscaler select from 32 MHz clock branch)
11 = 4 MHz (divide-by-8)(2)
10 = 8 MHz (divide-by-4)(2)
01 = 16 MHz (divide-by-2)
00 = 32 MHz (divide-by-1)

bit 5 **PLLEN:** USB PLL Enable bit
1 = PLL is always active
0 = PLL is only active when a PLL Oscillator mode is selected (OSCCON<14:12> = 011 or 001)

bit 4-0 **Unimplemented:** Read as '0'

Note 1: This bit is automatically cleared when the ROI bit is set and an interrupt occurs.
2: This setting is not allowed while the USB module is enabled.

REGISTER 9-3: OSCTUN: FRC OSCILLATOR TUNE REGISTER

R/W-0	U-0	R/W-0	R/W-0	R-0	R/W-0	R-0	R/W-0
STEN	—	STSIDL	STSRC(1)	STLOCK	STLPOL	STOR	STORPOL
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	TUN5(2)	TUN4(2)	TUN3(2)	TUN2(2)	TUN1(2)	TUN0(2)
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 **STEN:** FRC Self-Tune Enable bit
1 = FRC self-tuning is enabled; TUNx bits are controlled by hardware
0 = FRC self-tuning is disabled; application may optionally control TUNx bits

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

bit 13 **STSIDL:** FRC Self-Tune Stop in Idle bit
1 = Self-tuning stops during Idle mode
0 = Self-tuning continues during Idle mode

bit 12 **STSRC:** FRC Self-Tune Reference Clock Source bit(1)
1 = FRC is tuned to approximately match the USB host clock tolerance
0 = FRC is tuned to approximately match the 32.768 kHz SOSC tolerance

bit 11 **STLOCK:** FRC Self-Tune Lock Status bit
1 = FRC accuracy is currently within ±0.2% of the STSRC reference accuracy
0 = FRC accuracy may not be within ±0.2% of the STSRC reference accuracy

bit 10 **STLPOL:** FRC Self-Tune Lock Interrupt Polarity bit
1 = A self-tune lock interrupt is generated when STLOCK is '0'
0 = A self-tune lock interrupt is generated when STLOCK is '1'

bit 9 **STOR:** FRC Self-Tune Out of Range Status bit
1 = STSRC reference clock error is beyond the range of TUN<5:0>; no tuning is performed
0 = STSRC reference clock is within the tunable range; tuning is performed

bit 8 **STORPOL:** FRC Self-Tune Out of Range Interrupt Polarity bit
1 = A self-tune out of range interrupt is generated when STOR is '0'
0 = A self-tune out of range interrupt is generated when STOR is '1'

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

bit 5-0 **TUN<5:0>:** FRC Oscillator Tuning bits(2)
011111 = Maximum frequency deviation
011110 =
• • •
000001 =
000000 = Center frequency, oscillator is running at factory calibrated frequency
111111 =
• • •
100001 =
100000 = Minimum frequency deviation

Note 1: Use of either clock tuning reference source has specific application requirements. See **Section 9.5 "FRC Active Clock Tuning"** for details.

2: These bits are read-only when STEN = 1.

9.4 Clock Switching Operation

With few limitations, applications are free to switch between any of the four clock sources (POSC, SOSC, FRC and LPRC) under software control and at any time. To limit the possible side effects that could result from this flexibility, PIC24F devices have a safeguard lock built into the switching process.

Note:	The Primary Oscillator mode has three different submodes (XT, HS and EC) which are determined by the POSCMDx Configuration bits. While an application can switch to and from Primary Oscillator mode in software, it cannot switch between the different primary submodes without reprogramming the device.

9.4.1 ENABLING CLOCK SWITCHING

To enable clock switching, the FCKSM1 Configuration bit in CW2 must be programmed to '0'. (Refer to **Section 34.1 "Configuration Bits"** for further details.) If the FCKSM1 Configuration bit is unprogrammed ('1'), the clock switching function and Fail-Safe Clock Monitor function are disabled; this is the default setting.

The NOSCx control bits (OSCCON<10:8>) do not control the clock selection when clock switching is disabled. However, the COSC<2:0> bits (OSCCON<14:12>) will reflect the clock source selected by the FNOSCx Configuration bits.

The OSWEN control bit (OSCCON<0>) has no effect when clock switching is disabled; it is held at '0' at all times.

9.4.2 OSCILLATOR SWITCHING SEQUENCE

At a minimum, performing a clock switch requires this basic sequence:

1. If desired, read the COSCx bits (OSCCON<14:12>) to determine the current oscillator source.
2. Perform the unlock sequence to allow a write to the OSCCON register high byte.
3. Write the appropriate value to the NOSCx bits (OSCCON<10:8>) for the new oscillator source.
4. Perform the unlock sequence to allow a write to the OSCCON register low byte.
5. Set the OSWEN bit to initiate the oscillator switch.

Once the basic sequence is completed, the system clock hardware responds automatically as follows:

1. The clock switching hardware compares the COSCx bits with the new value of the NOSCx bits. If they are the same, then the clock switch is a redundant operation. In this case, the OSWEN bit is cleared automatically and the clock switch is aborted.
2. If a valid clock switch has been initiated, the LOCK (OSCCON<5>) and CF (OSCCON<3>) bits are cleared.
3. The new oscillator is turned on by the hardware if it is not currently running. If a crystal oscillator must be turned on, the hardware will wait until the OST expires. If the new source is using the PLL, then the hardware waits until a PLL lock is detected (LOCK = 1).
4. The hardware waits for 10 clock cycles from the new clock source and then performs the clock switch.
5. The hardware clears the OSWEN bit to indicate a successful clock transition. In addition, the NOSCx bit values are transferred to the COSCx bits.
6. The old clock source is turned off at this time, with the exception of LPRC (if WDT or FSCM is enabled) or SOSC (if SOSCEN remains set).

Note 1:	The processor will continue to execute code throughout the clock switching sequence. Timing-sensitive code should not be executed during this time.
2:	Direct clock switches between any Primary Oscillator mode with PLL and FRCPLL mode are not permitted. This applies to clock switches in either direction. In these instances, the application must switch to FRC mode as a transitional clock source between the two PLL modes.

A recommended code sequence for a clock switch includes the following:

1. Disable interrupts during the OSCCON register unlock and write sequence.
2. Execute the unlock sequence for the OSCCON high byte by writing 78h and 9Ah to OSCCON<15:8> in two back-to-back instructions.
3. Write the new oscillator source to the NOSCx bits in the instruction immediately following the unlock sequence.
4. Execute the unlock sequence for the OSCCON low byte by writing 46h and 57h to OSCCON<7:0> in two back-to-back instructions.
5. Set the OSWEN bit in the instruction immediately following the unlock sequence.
6. Continue to execute code that is not clock-sensitive (optional).
7. Invoke an appropriate amount of software delay (cycle counting) to allow the selected oscillator and/or PLL to start and stabilize.
8. Check to see if OSWEN is '0'. If it is, the switch was successful. If OSWEN is still set, then check the LOCK bit to determine the cause of the failure.

The core sequence for unlocking the OSCCON register and initiating a clock switch is shown in Example 9-1.

EXAMPLE 9-1: BASIC CODE SEQUENCE FOR CLOCK SWITCHING

```
;Place the new oscillator selection in W0
;OSCCONH (high byte) Unlock Sequence
MOV         #OSCCONH, w1
MOV         #0x78, w2
MOV         #0x9A, w3
MOV.b       w2, [w1]
MOV.b       w3, [w1]
;Set new oscillator selection
MOV.b       WREG, OSCCONH
;OSCCONL (low byte) unlock sequence
MOV         #OSCCONL, w1
MOV         #0x46, w2
MOV         #0x57, w3
MOV.b       w2, [w1]
MOV.b       w3, [w1]
;Start oscillator switch operation
BSET        OSCCON,#0
```

9.5 FRC Active Clock Tuning

PIC24FJ128GC010 family devices include an automatic mechanism to calibrate the FRC during run time. This system uses active clock tuning from a source of known accuracy to maintain the FRC within a very narrow margin of its nominal 8 MHz frequency. This allows for a frequency accuracy that is well within the requirements of the *"USB 2.0 Specification"*, regarding full-speed USB devices.

Note:	The self-tune feature maintains sufficient accuracy for operation in USB Device mode. For applications that function as a USB host, a high-accuracy clock source (±0.05%) is still required.

The self-tune system is controlled by the bits in the upper half of the OSCTUN register. Setting the STEN bit (OSCTUN<15>) enables the self-tuning feature, allowing the hardware to calibrate to a source selected by the STSRC bit (OSCTUN<12>). When STSRC = 1, the system uses the Start-of-Frame (SOF) packets from an external USB host for its source. When STSRC = 0, the system uses the crystal-controlled SOSC for its calibration source. Regardless of the source, the system uses the TUN<5:0> bits (OSCTUN<5:0>) to change the FRC Oscillator's frequency. Frequency monitoring and adjustment is dynamic, occurring continuously during run time. While the system is active, the TUNx bits cannot be written to by software.

Note:	To use the USB as a reference clock tuning source (STSRC = 1), the microcontroller must be configured for USB device operation and connected to a non-suspended USB host or hub port. If the SOSC is to be used as the reference clock tuning source (STSRC = 0), the SOSC must also be enabled for clock tuning to occur.

The self-tune system can generate a hardware interrupt, FSTIF. The interrupt can result from a drift of the FRC from the reference by greater than 0.2% in either direction, or whenever the frequency deviation is beyond the ability of the TUNx bits to correct (i.e., greater than 1.5%). The STLOCK and STOR status bits (OSCTUN<11,9>) are used to indicate these conditions.

The STLPOL and STORPOL bits (OSCTUN<10,8>) configure the FSTIF interrupt to occur in the presence or the absence of the conditions. It is the user's responsibility to monitor both the STLOCK and STOR bits to determine the exact cause of the interrupt.

Note:	The STLPOL and STORPOL bits should be ignored when the self-tune system is disabled (STEN = 0).

9.6 Oscillator Modes and USB Operation

Because of the timing requirements imposed by USB, an internal clock of 48 MHz is required at all times while the USB module is enabled and not in a suspended operating state. Since this is well beyond the maximum CPU clock speed, a method is provided to internally generate both the USB and system clocks from a single oscillator source. PIC24FJ128GC010 family devices use the same clock structure as most other PIC24FJ devices, but include a two-branch PLL system to generate the two clock signals.

The USB PLL block is shown in Figure 9-2. In this system, the input from the Primary Oscillator is divided down by a PLL prescaler to generate a 4 MHz output. This is used to drive an on-chip, 96 MHz PLL frequency multiplier to drive the two clock branches. One branch uses a fixed, divide-by-2 frequency divider to generate the 48 MHz USB clock. The other branch uses a fixed, divide-by-3 frequency divider and configurable PLL prescaler/divider to generate a range of system clock frequencies. The CPDIV<1:0> bits select the system clock speed; available clock options are listed in Table 9-2.

The USB PLL prescaler does not automatically sense the incoming oscillator frequency. The user must manually configure the PLL divider to generate the required 4 MHz output, using the PLLDIV<3:0> Configuration bits. This limits the choices for Primary Oscillator frequency to a total of 8 possibilities, shown in Table 9-3.

TABLE 9-2: SYSTEM CLOCK OPTIONS DURING USB OPERATION

MCU Clock Division (CPDIV<1:0>)	Microcontroller Clock Frequency
None (00)	32 MHz
÷2 (01)	16 MHz
÷4 (10)(1)	8 MHz
÷8 (11)(1)	4 MHz

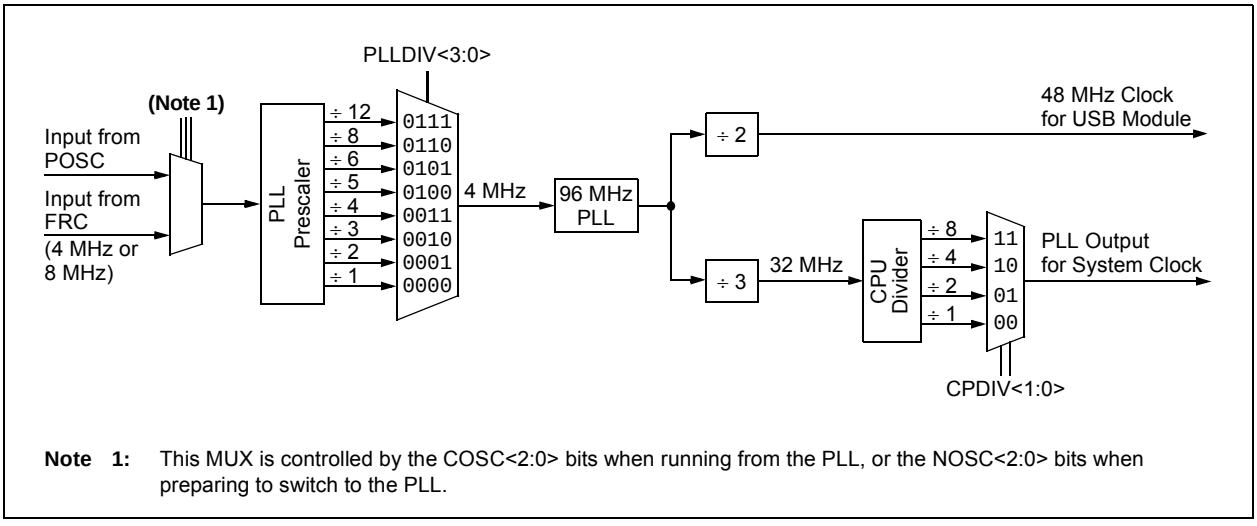
Note 1: This is not compatible with USB operation. The USB module must be disabled to use this system clock option.

TABLE 9-3: VALID PRIMARY OSCILLATOR CONFIGURATIONS FOR USB OPERATIONS

Input Oscillator Frequency	Clock Mode	PLL Division (PLLDIV<3:0>)
48 MHz	ECPLL	÷12 (0111)
32 MHz	HSPLL, ECPLL	÷8 (0110)
24 MHz	HSPLL, ECPLL	÷6 (0101)
20 MHz	HSPLL, ECPLL	÷5 (0100)
16 MHz	HSPLL, ECPLL	÷4 (0011)
12 MHz	HSPLL, ECPLL	÷3 (0010)
8 MHz	ECPLL, XTPLL, FRCPLL(1)	÷2 (0001)
4 MHz	ECPLL, XTPLL, FRCPLL(1)	÷1 (0000)

Note 1: This requires the use of the FRC self-tune feature to maintain required clock accuracy.

FIGURE 9-2: PLL BLOCK



Note 1: This MUX is controlled by the COSC<2:0> bits when running from the PLL, or the NOSC<2:0> bits when preparing to switch to the PLL.

9.6.1 CONSIDERATIONS FOR USB OPERATION

When using the USB On-The-Go module in PIC24FJ128GC010 family devices, users must always observe these rules in configuring the system clock:

- The oscillator modes listed in Table 9-3 are the only oscillator configurations that permit USB operation. There is no provision to provide a separate external clock source to the USB module.
- For USB operation, the selected clock source (EC, HS or XT) must meet the USB clock tolerance requirements.
- When the FRCPLL Oscillator mode is used for USB applications, the FRC self-tune system should be used as well. While the FRC is accurate, the only two ways to ensure the level of accuracy required by the *"USB 2.0 Specification"*, throughout the application's operating range, are either the self-tune system or manually changing the TUN<5:0> bits.
- The user must always ensure that the FRC source is configured to provide a frequency of 4 MHz or 8 MHz (RCDIV<2:0> = 001 or 000) and that the USB PLL prescaler is configured appropriately.
- All other oscillator modes are available; however, USB operation is not possible when these modes are selected. They may still be useful in cases where other power levels of operation are desirable and the USB module is not needed (e.g., the application is Sleeping and waiting for a bus attachment).

9.7 Reference Clock Output

In addition to the CLKO output (FOSC/2) available in certain oscillator modes, the device clock in the PIC24FJ128GC010 family devices can also be configured to provide a reference clock output signal to a port pin. This feature is available in all oscillator configurations and allows the user to select a greater range of clock submultiples to drive external devices in the application.

This reference clock output is controlled by the REFOCON register (Register 9-4). Setting the ROEN bit (REFOCON<15>) makes the clock signal available on the REFO pin. The RODIV<3:0> bits (REFOCON<11:8>) enable the selection of 16 different clock divider options.

The ROSSLP and ROSEL bits (REFOCON<13:12>) control the availability of the reference output during Sleep mode. The ROSEL bit determines if the oscillator on OSC1 and OSC2, or the current system clock source, is used for the reference clock output. The ROSSLP bit determines if the reference source is available on REFO when the device is in Sleep mode.

To use the reference clock output in Sleep mode, both the ROSSLP and ROSEL bits must be set. The device clock must also be configured for one of the primary modes (EC, HS or XT); otherwise, if the POSCEN bit is not also set, the oscillator on OSC1 and OSC2 will be powered down when the device enters Sleep mode. Clearing the ROSEL bit allows the reference output frequency to change as the system clock changes during any clock switches.

9.8 Secondary Oscillator

9.8.1 BASIC SOSC OPERATION

PIC24FJ128GC010 family devices do not have to set the SOSCEN bit to use the Secondary Oscillator. Any module requiring the SOSC (such as RTCC, Timer1 or DSWDT) will automatically turn on the SOSC when the clock signal is needed. The SOSC, however, has a long start-up time (as long as 1 second).To avoid delays for peripheral start-up, the SOSC can be manually started using the SOSCEN bit.

To use the Secondary Oscillator, the SOSCSEL bit (CW3<8>) must be set to '1'. Programming the SOSCSEL bit to '0' configures the SOSC pins for Digital mode, enabling digital I/O functionality on the pins.

9.8.2 CRYSTAL SELECTION

The 32.768 kHz crystal used for the SOSC must have the following specifications in order to properly start up and run at the correct frequency:

- 12.5 pF loading capacitance
- 1.0 pF shunt capacitance
- A typical ESR of 50K; 70K maximum

In addition, the two external crystal loading capacitors should be in the range of 22-27 pF, which will be based on the PC board layout. The capacitors should be C0G, 5% tolerance and rated 25V or greater.

The accuracy and duty cycle of the SOSC can be measured on the REFO pin and is recommended to be in the range of 40-60% and accurate to ±0.65Hz.

Note:	Do not enable the LCD segment pin, SEG17, on RD0 when using the 64-pin package if the SOSC is used for time-sensitive applications. Avoid high-frequency traces adjacent to the SOSCO and SOSCI pins as this can cause errors in the SOSC frequency and/or duty cycle.

REGISTER 9-4: REFOCON: REFERENCE OSCILLATOR CONTROL REGISTER

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ROEN	—	ROSSLP	ROSEL	RODIV3	RODIV2	RODIV1	RODIV0
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 **ROEN:** Reference Oscillator Output Enable bit
1 = Reference oscillator is enabled on the REFO pin
0 = Reference oscillator is disabled

bit 14 **Unimplemented:** Read as ‘0’

bit 13 **ROSSLP:** Reference Oscillator Output Stop in Sleep bit
1 = Reference oscillator continues to run in Sleep
0 = Reference oscillator is disabled in Sleep

bit 12 **ROSEL:** Reference Oscillator Source Select bit
1 = Primary Oscillator is used as the base clock. Note that the crystal oscillator must be enabled using the FOSC<2:0> bits; crystal maintains the operation in Sleep mode.
0 = System clock is used as the base clock; base clock reflects any clock switching of the device

bit 11-8 **RODIV<3:0>:** Reference Oscillator Divisor Select bits
1111 = Base clock value divided by 32,768
1110 = Base clock value divided by 16,384
1101 = Base clock value divided by 8,192
1100 = Base clock value divided by 4,096
1011 = Base clock value divided by 2,048
1010 = Base clock value divided by 1,024
1001 = Base clock value divided by 512
1000 = Base clock value divided by 256
0111 = Base clock value divided by 128
0110 = Base clock value divided by 64
0101 = Base clock value divided by 32
0100 = Base clock value divided by 16
0011 = Base clock value divided by 8
0010 = Base clock value divided by 4
0001 = Base clock value divided by 2
0000 = Base clock value

bit 7-0 **Unimplemented:** Read as ‘0’

NOTES:

10.0 POWER-SAVING FEATURES

Note: This data sheet summarizes the features of this group of PIC24FJ devices. It is not intended to be a comprehensive reference source. For more information, refer to the *"PIC24F Family Reference Manual"*, **"Power-Saving Features with Deep Sleep"** (DS39727).

The PIC24FJ128GC010 family of devices provides the ability to manage power consumption by selectively managing clocking to the CPU and the peripherals. In general, a lower clock frequency and a reduction in the number of circuits being clocked reduces consumed power.

PIC24FJ128GC010 family devices manage power consumption with five strategies:

- Instruction-Based Power Reduction Modes
- Hardware-Based Power Reduction Features
- Clock Frequency Control
- Software Controlled Doze Mode
- Selective Peripheral Control in Software

Combinations of these methods can be used to selectively tailor an application's power consumption, while still maintaining critical application features, such as timing-sensitive communications.

10.1 Overview of Power-Saving Modes

In addition to full-power operation, otherwise known as Run mode, the PIC24FJ128GC010 family of devices offers three Instruction-Based Power-Saving modes and one Hardware-Based mode:

- Idle
- Sleep (Sleep and Low-Voltage Sleep)
- Deep Sleep (with and without retention)
- VBAT (with and without RTCC)

All four modes can be activated by powering down different functional areas of the microcontroller, allowing progressive reductions of operating and Idle power consumption. In addition, three of the modes can be tailored for more power reduction, at a trade-off of some operating features. Table 10-1 lists all of the operating modes, in order of increasing power savings. Table 10-2 summarizes how the microcontroller exits the different modes. Specific information is provided in the following sections.

TABLE 10-1: OPERATING MODES FOR PIC24FJ128GC010 FAMILY DEVICES

Mode	Entry	Active Systems				
		Core	Peripherals	Data RAM Retention	RTCC(1)	DSGPR0/ DSGPR1 Retention
Run (default)	N/A	Y	Y	Y	Y	Y
Idle	Instruction	N	Y	Y	Y	Y
Sleep:						
Sleep	Instruction	N	S(2)	Y	Y	Y
Low-Voltage Sleep	Instruction + RETEN bit	N	S(2)	Y	Y	Y
Deep Sleep:						
Retention Deep Sleep	Instruction + DSEN bit + RETEN bit	N	N	Y	Y	Y
Deep Sleep	Instruction + DSEN bit	N	N	N	Y	Y
VBAT:						
with RTCC	Hardware	N	N	N	Y	Y
w/o RTCC	Hardware + RTCBAT Config. bit	N	N	N	N	Y

Note 1: If RTCC is otherwise enabled in firmware.

2: A select peripheral can operate during this mode from LPRC or some external clock.

TABLE 10-2: EXITING POWER SAVING MODES

Mode	Exit Conditions								Code Execution Resumes
	Interrupts		Resets			RTCC Alarm	WDT	VDD Restore(2)	
	All	INT0	All	POR	$\overline{MCLR}$				
Idle	Y	Y	Y	Y	Y	Y	Y	N/A	Next instruction
Sleep (all modes)	Y	Y	Y	Y	Y	Y	Y	N/A	
Deep Sleep	N	Y	N	Y	Y	Y	Y(1)	N/A	Reset vector
Retention Deep Sleep	N	Y	N	Y	Y	Y	Y(1)	N/A	Next instruction
VBAT	N	N	N	N	N	N	N	Y	Reset vector

Note 1: Deep Sleep WDT.

2: A POR or POR-like Reset results whenever VDD is removed and restored in any mode except for Retention Deep Sleep.

10.1.1 INSTRUCTION-BASED POWER-SAVING MODES

Three of the power-saving modes are entered through the execution of the PWRSAV instruction. Sleep mode stops clock operation and halts all code execution. Idle mode halts the CPU and code execution, but allows peripheral modules to continue operation. Deep Sleep mode stops clock operation, code execution and all peripherals, except RTCC and DSWDT. It also freezes I/O states and removes power to Flash memory, and may remove power to SRAM.

The assembly syntax of the PWRSAV instruction is shown in Example 10-1. Sleep and Idle modes are entered directly with a single assembler command. Deep Sleep requires an additional sequence to unlock and enable the entry into Deep Sleep, which is described in **Section 10.4.2 "Entering Deep Sleep Mode"**.

Note:	SLEEP_MODE and IDLE_MODE are constants defined in the assembler include file for the selected device.

Sleep and Idle modes can be exited as a result of an enabled interrupt, WDT time-out or a device Reset. When the device exits these modes, it is said to "wake-up".

The features enabled with the low-voltage/retention regulator result in some changes to the way that Sleep and Deep Sleep modes behave. See **Section 10.3 "Sleep Mode"** and **Section 10.4 "Deep Sleep Mode"** for additional information.

10.1.1.1 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.

For Deep Sleep mode, interrupts that coincide with the execution of the PWRSAV instruction may be lost. If the low-voltage/retention regulator is not enabled, the microcontroller resets on leaving Deep Sleep and the interrupt will be lost. If the low-voltage/retention regulator is enabled, the microcontroller will exit Deep Sleep and the interrupt will then be handled.

Interrupts that occur during the Deep Sleep unlock sequence will interrupt the mandatory five-instruction cycle sequence timing and cause a failure to enter Deep Sleep. For this reason, it is recommended to disable all interrupts during the Deep Sleep unlock sequence.

EXAMPLE 10-1: PWRSAV INSTRUCTION SYNTAX

```
// Syntax to enter Sleep mode:
PWRSAV     #SLEEP_MODE        ; Put the device into SLEEP mode
//
//Synatx to enter Idle mode:
PWRSAV     #IDLE_MODE         ; Put the device into IDLE mode
//
// Syntax to enter Deep Sleep mode:
// First use the unlock sequence to set the DSEN bit (see Example 10-2)
BSET       DSCON, #DSEN       ;Enable Deep Sleep
BSET       DSCON, #DSEN       ; Enable Deep Sleep(repeat the command)
PWRSAV     #SLEEP_MODE        ; Put the device into Deep SLEEP mode
```

10.1.2 HARDWARE-BASED POWER-SAVING MODE

The hardware-based VBAT mode does not require any action by the user during code development. Instead, it is a hardware design feature that allows the microcontroller to retain critical data (using the DSGPRx registers) and maintain the RTCC when VDD is removed from the application. This is accomplished by supplying a backup power source to a specific power pin. VBAT mode is described in more detail in **Section 10.5 "Vbat Mode"**.

10.1.3 LOW-VOLTAGE/RETENTION REGULATOR

PIC24FJ128GC010 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.

The low-voltage/retention regulator is only available when Sleep or Deep Sleep modes are invoked. It is controlled by the LPCFG Configuration bit (CW1<10>) 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 Idle Mode

Idle mode provides 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 10.8 "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 re-applied to the CPU and instruction execution begins immediately, starting with the instruction following the PWRSAV instruction or the first instruction in the Interrupt Service Routine (ISR).

10.3 Sleep Mode

Sleep mode includes these features:

- The system clock source is shut down. If an on-chip oscillator is used, it is turned off.
- The device current consumption will be reduced to a minimum provided that no I/O pin is sourcing current.
- The I/O pin directions and states are frozen.
- The Fail-Safe Clock Monitor does not operate during Sleep mode since the system clock source is disabled.
- The LPRC clock will continue to run in Sleep mode if the WDT or RTCC, with LPRC as the clock source, is enabled.
- The WDT, if enabled, is automatically cleared prior to entering Sleep mode.
- Some device features or peripherals may continue to operate in Sleep mode. This includes items, such as the Input Change Notification (ICN) on the I/O ports, or peripherals that use an external clock input. Any peripheral that requires the system clock source for its operation will be disabled in Sleep mode.

The device will wake-up from Sleep mode on any of these events:

- On any interrupt source that is individually enabled
- On any form of device Reset
- On a WDT time-out

On wake-up from Sleep, the processor will restart with the same clock source that was active when Sleep mode was entered.

10.3.1 LOW-VOLTAGE/RETENTION SLEEP MODE

Low-Voltage/Retention Sleep mode functions as Sleep mode with the same features and wake-up triggers. The difference is that the low-voltage/retention regulator allows Core Digital Logic Voltage (VCORE) to drop to 1.2V nominal. This permits an incremental reduction of power consumption over what would be required if VCORE was maintained at a 1.8V (minimum) level.

Low-Voltage Sleep mode requires a longer wake-up time than Sleep mode, due to the additional time required to bring VCORE back to 1.8V (known as TREG). In addition, the use of the low-voltage/retention regulator limits the amount of current that can be sourced to any active peripherals, such as the RTCC/LCD, etc.

10.4 Deep Sleep Mode

Deep Sleep mode provides the lowest levels of power consumption available from the Instruction-Based modes. PIC24FJ128GC010 family devices have two Deep Sleep modes: Legacy Deep Sleep, found in other PIC24F devices, and Retention Deep Sleep, described below.

Deep Sleep modes have these features:

- The system clock source is shut down. If an on-chip oscillator is used, it is turned off.
- The device current consumption will be reduced to a minimum.
- The I/O pin directions and states are frozen.
- The Fail-Safe Clock Monitor does not operate during Sleep mode since the system clock source is disabled.
- The LPRC clock will continue to run in Deep Sleep mode if the WDT, or RTCC with LPRC as the clock source, is enabled.
- The dedicated Deep Sleep WDT and BOR systems, if enabled, are used.
- The RTCC and its clock source continue to run, if enabled. All other peripherals are disabled.

Entry into Deep Sleep mode is completely under software control. Exit from the Deep Sleep modes can be triggered from any of the following events:

- POR event
- $\overline{\text{MCLR}}$ event
- RTCC alarm (if the RTCC is present)
- External Interrupt 0
- Deep Sleep Watchdog Timer (DSWDT) time-out

10.4.1 RETENTION DEEP SLEEP

Unlike Deep Sleep mode, Retention Deep Sleep mode represents an incremental increase in power consumption. Although it also allows the device to operate at a V_{CORE} of 1.2V, the low-voltage/retention regulator is used in this mode to maintain the contents of the data RAM, which slightly increases current consumption. Maintaining data RAM (including the SFRs) has several effects that make Retention Deep Sleep different form Deep Sleep:

- The wake-up sources are the same as those for Deep Sleep mode.
- Wake-up from Retention Deep Sleep allows the device to resume its previous state and start code execution where it left off, instead of restarting at the Reset vector (as with Deep Sleep).

10.4.2 ENTERING DEEP SLEEP MODE

Deep Sleep mode is entered by setting the DSEN bit in the DSCON register and then executing a Sleep command (`PWRSAV #SLEEP_MODE`) within one instruction cycle to minimize the chance that Deep Sleep will be spuriously entered. If the low-voltage/retention regulator is already enabled, prior to setting the DSEN bit, the device will enter Retention Deep Sleep.

If the `PWRSAV` command is not given within one instruction cycle, the DSEN bit will be cleared by the hardware and must be set again by the software before entering Deep Sleep mode. The DSEN bit is also automatically cleared when exiting Deep Sleep mode.

Note:	To re-enter Deep Sleep after a Deep Sleep wake-up, allow a delay of at least 3 T_{CY} after clearing the RELEASE bit.

The sequence to enter Deep Sleep mode is:

1. If the application requires the Deep Sleep WDT, enable it and configure its clock source. For more information on Deep Sleep WDT, see **Section 10.4.6 “Deep Sleep WDT”**.
2. If the application requires Deep Sleep BOR, enable it by programming the DSBOREN Configuration bit (CW4<6>).
3. If the application requires wake-up from Deep Sleep on RTCC alarm, enable and configure the RTCC module. For more information on RTCC, see **Section 23.0 “Real-Time Clock and Calendar (RTCC)”**.
4. If needed, save any critical application context data by writing it to the DSGPR0 and DSGPR1 registers (optional).
5. Enable Deep Sleep mode by setting the DSEN bit (DSCON<15>).

Note:	A repeat sequence is required to set the DSEN bit. The repeat sequence (repeating the instruction twice) is required to write to any of the Deep Sleep registers (DSCON, DSWAKE, DSGPR0, DSGPR1). This is required to prevent the user from entering Deep Sleep by mistake. Any write to these registers has to be done twice to actually complete the write (see Example 10-2).

6. Enter Deep Sleep mode by issuing 3 `NOP` commands and then a `PWRSAV #0` instruction.

Any time the DSEN bit is set, all bits in the DSWAKE register will be automatically cleared.

EXAMPLE 10-2: THE REPEAT SEQUENCE

Example 1:

```
mov    #8000, w2   ; enable DS
mov    w2, DSCON
mov    w2, DSCON   ; second write required to
                     actually write to DSCON
```

Example 2:

```
bset   DSCON, #15
nop
nop
nop
bset   DSCON, #15  ; enable DS (two writes required)
```

10.4.3 EXITING DEEP SLEEP MODES

Deep Sleep modes exit on any one of the following events:

- POR event on VDD supply. If there is no DSBOR circuit to re-arm the VDD supply POR circuit, the external VDD supply must be lowered to the natural arming voltage of the POR circuit.
- DSWDT time-out. When the DSWDT timer times out, the device exits Deep Sleep.
- RTCC alarm (if RTCEN = 1).
- Assertion ('0') of the $\overline{\text{MCLR}}$ pin.
- Assertion of the INT0 pin (if the interrupt was enabled before Deep Sleep mode was entered). The polarity configuration is used to determine the assertion level ('0' or '1') of the pin that will cause an exit from Deep Sleep mode. Exiting from Deep Sleep mode requires a change on the INT0 pin while in Deep Sleep mode.

Note:	Any interrupt pending when entering Deep Sleep mode is cleared.

Exiting Deep Sleep generally does not retain the state of the device and is equivalent to a Power-on Reset (POR) of the device. Exceptions to this include the RTCC (if present), which remains operational through the wake-up, the DSGPRx registers and DSWDT.

Wake-up events that occur from the time Deep Sleep exits, until the time the POR sequence completes, are not ignored. The DSWAKE register will capture ALL wake-up events, from setting DSEN to clearing RELEASE.

The sequence for exiting Deep Sleep mode is:

1. After a wake-up event, the device exits Deep Sleep and performs a POR. The DSEN bit is cleared automatically. Code execution resumes at the Reset vector.
2. To determine if the device exited Deep Sleep, read the Deep Sleep bit, DPSLP (RCON<10>). This bit will be set if there was an exit from Deep Sleep mode. If the bit is set, clear it.
3. Determine the wake-up source by reading the DSWAKE register.
4. Determine if a DSBOR event occurred during Deep Sleep mode by reading the DSBOR bit (DSCON<1>).
5. If application context data has been saved, read it back from the DSGPR0 and DSGPR1 registers.
6. Clear the RELEASE bit (DSCON<0>).

10.4.4 SAVING CONTEXT DATA WITH THE DSGPRx REGISTERS

As exiting Deep Sleep mode causes a POR, most Special Function Registers reset to their default POR values. In addition, because VCORE power is not supplied in Deep Sleep mode, information in data RAM may be lost when exiting this mode.

Applications which require critical data to be saved, prior to Deep Sleep, may use the Deep Sleep General Purpose registers, DSGPR0 and DSGPR1, or data EEPROM (if available). Unlike other SFRs, the contents of these registers are preserved while the device is in Deep Sleep mode. After exiting Deep Sleep, software can restore the data by reading the registers and clearing the RELEASE bit (DSCON<0>).

10.4.5 I/O PINS IN DEEP SLEEP MODES

During Deep Sleep, the general purpose I/O pins retain their previous states and the Secondary Oscillator (SOSC) will remain running, if enabled. Pins that are configured as inputs (TRISx bit set), prior to entry into Deep Sleep, remain high-impedance during Deep Sleep. Pins that are configured as outputs (TRISx bit clear), prior to entry into Deep Sleep, remain as output pins during Deep Sleep. While in this mode, they continue to drive the output level determined by their corresponding LATx bit at the time of entry into Deep Sleep.

Once the device wakes back up, all I/O pins continue to maintain their previous states, even after the device has finished the POR sequence and is executing application code again. Pins configured as inputs during Deep Sleep remain high-impedance and pins configured as outputs continue to drive their previous value. After waking up, the TRIS and LAT registers, and the SOSCEN bit (OSCCON<1>), are reset. If firmware modifies any of these bits or registers, the I/O will not immediately go to the newly configured states. Once the firmware clears the RELEASE bit (DSCON<0>), the I/O pins are “released”. This causes the I/O pins to take the states configured by their respective TRISx and LATx bit values.

This means that keeping the SOSC running after waking up requires the SOSCEN bit to be set before clearing RELEASE.

If the Deep Sleep BOR (DSBOR) is enabled, and a DSBOR or a true POR event occurs during Deep Sleep, the I/O pins will be immediately released, similar to clearing the RELEASE bit. All previous state information will be lost, including the general purpose DSGPR0 and DSGPR1 contents.

If a $\overline{\text{MCLR}}$ Reset event occurs during Deep Sleep, the DSGPRx, DSCON and DSWAKE registers will remain valid, and the RELEASE bit will remain set. The state of the SOSC will also be retained. The I/O pins, however, will be reset to their $\overline{\text{MCLR}}$ Reset state. Since RELEASE is still set, changes to the SOSCEN bit (OSCCON<1>) cannot take effect until the RELEASE bit is cleared.

In all other Deep Sleep wake-up cases, application firmware must clear the RELEASE bit in order to reconfigure the I/O pins.

10.4.6 DEEP SLEEP WDT

To enable the DSWDT in Deep Sleep mode, program the Configuration bit, DSWDTEN (CW4<7>). The device WDT need not be enabled for the DSWDT to function. Entry into Deep Sleep modes automatically resets the DSWDT.

The DSWDT clock source is selected by the DSWDTOSC Configuration bit (CW4<5>). The posts-caler options are programmed by the DSWDPS<4:0> Configuration bits (CW4<4:0>). The minimum time-out period that can be achieved is 1 ms and the maximum is 25.7 days. For more details on DSWDT configuration options, refer to **Section 34.0 “Special Features”**.

10.4.6.1 Switching Clocks in Deep Sleep Mode

Both the RTCC and the DSWDT may run from either SOSC or the LPRC clock source. This allows both the RTCC and DSWDT to run without requiring both the LPRC and SOSC to be enabled together, reducing power consumption.

Running the RTCC from LPRC will result in a loss of accuracy in the RTCC, of approximately 5 to 10%. If a more accurate RTCC is required, it must be run from the SOSC clock source. The RTCC clock source is selected with the RTCLK<1:0> bits (RTCPWC<11:10>).

Under certain circumstances, it is possible for the DSWDT clock source to be off when entering Deep Sleep mode. In this case, the clock source is turned on automatically (if DSWDT is enabled) without the need for software intervention. However, this can cause a delay in the start of the DSWDT counters. In order to avoid this delay when using SOSC as a clock source, the application can activate SOSC prior to entering Deep Sleep mode.

10.4.7 CHECKING AND CLEARING THE STATUS OF DEEP SLEEP

Upon entry into Deep Sleep mode, the status bit, DPSLP (RCON<10>), becomes set and must be cleared by the software.

On power-up, the software should read this status bit to determine if the Reset was due to an exit from Deep Sleep mode and clear the bit if it is set. Of the four possible combinations of DPSLP and POR bit states, three cases can be considered:

- Both the DPSLP and POR bits are cleared. In this case, the Reset was due to some event other than a Deep Sleep mode exit.
- The DPSLP bit is clear, but the POR bit is set; this is a normal POR.
- Both the DPSLP and POR bits are set. This means that Deep Sleep mode was entered, the device was powered down and Deep Sleep mode was exited.

10.4.8 POWER-ON RESETS (PORs)

V_{DD} voltage is monitored to produce PORs. Since exiting from Deep Sleep mode functionally looks like a POR, the technique described in **Section 10.4.7 “Checking and Clearing the Status of Deep Sleep”** should be used to distinguish between Deep Sleep and a true POR event. When a true POR occurs, the entire device, including all Deep Sleep logic (Deep Sleep registers, RTCC, DSWDT, etc.), is reset.

10.5 VBAT Mode

This mode represents the lowest power state that the microcontroller can achieve and still resume operation. VBAT mode is automatically triggered when the microcontroller's main power supply on VDD fails. When this happens, the microcontroller's on-chip power switch connects to a backup power source, such as a battery, supplied to the VBAT pin. This maintains a few key systems at an extremely low-power draw until VDD is restored.

The power supplied on VBAT only runs two systems: the RTCC and the Deep Sleep Semaphore registers (DSGPR0 and DSGPR1). To maintain these systems during a sudden loss of VDD, it is essential to connect a power source, other than VDD or AVDD, to the VBAT pin.

When the RTCC is enabled, it continues to operate with the same clock source (SOSC or LPRC) that was selected prior to entering VBAT mode. There is no provision to switch to a lower power clock source after the mode switch.

Since the loss of VDD is usually an unforeseen event, it is recommended that the contents of the Deep Sleep Semaphore registers be loaded with the data to be retained at an early point in code execution.

10.5.1 VBAT MODE WITH NO RTCC

By disabling RTCC operation during VBAT mode, power consumption is reduced to the lowest of all power-saving modes. This is done by programming the RTCBAT Configuration bit (CW4<9>) to '0'. In this mode, only the Deep Sleep Semaphore registers are maintained.

10.5.2 WAKE-UP FROM VBAT MODES

When VDD is restored to a device in VBAT mode, it automatically wakes. Wake-up occurs with a POR, after which, the device starts executing code from the Reset vector. All SFRs, except the Deep Sleep Semaphore registers, are reset to their POR values. IF the RTCC was not configured to run during VBAT mode, it will remain disabled and RTCC will not run. Wake-up timing is similar to that for a normal POR.

To differentiate a wake-up from VBAT mode from other POR states, check the VBAT status bit (RCON2<0>). If this bit is set while the device is starting to execute the code from the Reset vector, it indicates that there has been an exit from VBAT mode. The application must clear the VBAT bit to ensure that future VBAT wake-up events are captured.

If a POR occurs without a power source connected to the VBAT pin, the VBPOR bit (RCON2<1>) is set. If this bit is set on a POR, it indicates that a battery needs to be connected to the VBAT pin.

In addition, if the VBAT power source falls below the level needed for Deep Sleep semaphore operation while in VBAT mode (e.g., the battery has been drained), the VBPOR bit will be set. VBPOR is also set when the microcontroller is powered up the very first time, even if power is supplied to VBAT.

10.5.3 I/O PINS DURING VBAT MODES

All I/O pins switch to Input mode during VBAT mode. The only exceptions are the SOSCI and SOSCO pins, which maintain their states if the Secondary Oscillator is being used as the RTCC clock source. It is the user's responsibility to restore the I/O pins to their proper states, using the TRISx and LATx bits, once VDD has been restored.

10.5.4 SAVING CONTEXT DATA WITH THE DSGPRx REGISTERS

As with Deep Sleep mode (i.e., without the low-voltage/retention regulator), all SFRs are reset to their POR values after VDD has been restored. Only the Deep Sleep Semaphore registers are preserved. Applications which require critical data to be saved should save it in DSGPR0 and DSGPR1.

Note:	If the VBAT mode is not used, it is recommended to connect the VBAT pin to VDD.

The POR should be enabled for the reliable operation of the VBAT.

REGISTER 10-1: DSCON: DEEP SLEEP CONTROL REGISTER(1)

R/W-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
DSEN	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	r-0	R/W-0	R/C-0, HS
—	—	—	—	—	r	DSBOR(2)	RELEASE
bit 7							bit 0

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

bit 15 **DSEN:** Deep Sleep Enable bit
1 = Enters Deep Sleep on execution of PWRSAV #0
0 = Enters normal Sleep on execution of PWRSAV #0

bit 14-3 **Unimplemented:** Read as '0'

bit 2 **Reserved:** Maintain as '0'

bit 1 **DSBOR:** Deep Sleep BOR Event bit(2)
1 = The DSBOR was active and a BOR event was detected during Deep Sleep
0 = The DSBOR was not active or was active but did not detect a BOR event during Deep Sleep

bit 0 **RELEASE:** I/O Pin State Release bit
1 = Upon waking from Deep Sleep, I/O pins maintain their states previous to Deep Sleep entry
0 = Releases I/O pins from their state previous to Deep Sleep entry, and allows their respective TRISx and LATx bits to control their states

Note 1: All register bits are reset only in the case of a POR event outside of Deep Sleep mode.
2: Unlike all other events, a Deep Sleep BOR event will NOT cause a wake-up from Deep Sleep; this re-arms the POR.

REGISTER 10-2: DSWAKE: DEEP SLEEP WAKE-UP SOURCE REGISTER(1)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0, HS
—	—	—	—	—	—	—	DSINT0
bit 15							bit 8

R/W-0, HS	U-0	U-0	R/W-0, HS	R/W-0, HS	R/W-0, HS	U-0	U-0
DSFLT	—	—	DSWDT	DSRTCC	DSMCLR	—	—
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 15-9 **Unimplemented:** Read as '0'

bit 8 **DSINT0:** Deep Sleep Interrupt-on-Change bit
1 = Interrupt-on-change was asserted during Deep Sleep
0 = Interrupt-on-change was not asserted during Deep Sleep

bit 7 **DSFLT:** Deep Sleep Fault Detect bit
1 = A Fault occurred during Deep Sleep and some Deep Sleep configuration settings may have been corrupted
0 = No Fault was detected during Deep Sleep

bit 6-5 **Unimplemented:** Read as '0'

bit 4 **DSWDT:** Deep Sleep Watchdog Timer Time-out bit
1 = The Deep Sleep Watchdog Timer timed out during Deep Sleep
0 = The Deep Sleep Watchdog Timer did not time out during Deep Sleep

bit 3 **DSRTCC:** Deep Sleep Real-Time Clock and Calendar Alarm bit
1 = The Real-Time Clock and Calendar triggered an alarm during Deep Sleep
0 = The Real-Time Clock and Calendar did not trigger an alarm during Deep Sleep

bit 2 **DSMCLR:** Deep Sleep $\overline{\text{MCLR}}$ Event bit
1 = The $\overline{\text{MCLR}}$ pin was active and was asserted during Deep Sleep
0 = The $\overline{\text{MCLR}}$ pin was not active or was active, but not asserted during Deep Sleep

bit 1-0 **Unimplemented:** Read as '0'

Note 1: All register bits are cleared when the DSEN (DSCON<15>) bit is set.

REGISTER 10-3: RCON2: RESET AND SYSTEM CONTROL REGISTER 2

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	r-0	R/CO-1	R/CO-1	R/CO-1	R/CO-0
—	—	—	r	VDDBOR(1)	VDDPOR(1,2)	VBPOR(1,3)	VBAT(1)
bit 7							bit 0

Legend:	CO = Clearable Only bit	r = Reserved 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 15-5 **Unimplemented:** Read as '0'

bit 4 **Reserved:** Maintain as '0'

bit 3 **VDDBOR:** VDD Brown-out Reset Flag bit(1)
- 1 = A VDD Brown-out Reset has occurred (set by hardware)
- 0 = A VDD Brown-out Reset has not occurred

bit 2 **VDDPOR:** VDD Power-on Reset Flag bit(1,2)
- 1 = A VDD Power-on Reset has occurred (set by hardware)
- 0 = A VDD Power-on Reset has not occurred

bit 1 **VBPOR:** VBAT Power-on Reset Flag bit(1,3)
- 1 = A VBAT POR has occurred (no battery connected to the VBAT pin or VBAT power is below Deep Sleep semaphore retention level, set by hardware)
- 0 = A VBAT POR has not occurred

bit 0 **VBAT:** VBAT Flag bit(1)
- 1 = A POR exit has occurred while power was applied to the VBAT pin (set by hardware)
- 0 = A POR exit from VBAT has not occurred

Note 1: This bit is set in hardware only; it can only be cleared in software.

2: Indicates a VDD POR. Setting the POR bit (RCON<0>) indicates a VCORE POR.

3: This bit is set when the device is originally powered up, even if power is present on VBAT.

10.6 Clock Frequency and Clock Switching

In Run and Idle modes, all PIC24FJ devices allow for a wide range of clock frequencies to be selected under application control. If the system clock configuration is not locked, users can choose low-power or high-precision oscillators by simply changing the NOSCx bits. The process of changing a system clock during operation, as well as limitations to the process, are discussed in more detail in **Section 9.0 "Oscillator Configuration"**.

10.7 Doze Mode

Generally, changing clock speed and invoking one of the power-saving modes are the preferred strategies for reducing power consumption. There may be circumstances, however, where this is not practical. For example, it may be necessary for an application to maintain uninterrupted synchronous communication, even while it is doing nothing else. Reducing system clock speed may introduce communication errors, while using a power-saving mode may stop communications completely.

Doze mode is a simple and effective alternative method to reduce power consumption while the device is still executing code. In this mode, the system clock continues to operate from the same source and at the same speed. Peripheral modules continue to be clocked at the same speed, while the CPU clock speed is reduced. Synchronization between the two clock domains is maintained, allowing the peripherals to access the SFRs while the CPU executes code at a slower rate.

Doze mode is enabled by setting the DOZEN bit (CLKDIV<11>). The ratio between peripheral and core clock speed is determined by the DOZE<2:0> bits (CLKDIV<14:12>). There are eight possible configurations, from 1:1 to 1:128, with 1:8 being the default.

It is also possible to use Doze mode to selectively reduce power consumption in event driven applications. This allows clock-sensitive functions, such as synchronous communications, to continue without interruption while the CPU Idles, waiting for something to invoke an interrupt routine. Enabling the automatic return to full-speed CPU operation on interrupts is enabled by setting the ROI bit (CLKDIV<15>). By default, interrupt events have no effect on Doze mode operation.

10.8 Selective Peripheral Module Control

Idle and Doze modes allow users to substantially reduce power consumption by slowing or stopping the CPU clock. Even so, peripheral modules still remain clocked, and thus, consume power. There may be cases where the application needs what these modes do not provide: the allocation of power resources to the CPU processing with minimal power consumption from the peripherals.

PIC24F devices address this requirement by allowing peripheral modules to be selectively disabled, reducing or eliminating their power consumption. This can be done with two control bits:

- The Peripheral Enable bit, generically named, "XXXEN", located in the module's main control SFR.
- The Peripheral Module Disable (PMD) bit, generically named, "XXXMD", located in one of the PMDn Control registers (XXXMD bits are in PMDn registers, shown in Table 4-39).

Both bits have similar functions in enabling or disabling its associated module. Setting the PMD bit for a module disables all clock sources to that module, reducing its power consumption to an absolute minimum. In this state, the control and status registers associated with the peripheral will also be disabled, so writes to those registers will have no effect and read values will be invalid. Many peripheral modules have a corresponding PMD bit.

In contrast, disabling a module by clearing its XXXEN bit disables its functionality, but leaves its registers available to be read and written to. Power consumption is reduced, but not by as much as when the PMD bits are used. Most peripheral modules have an enable bit; exceptions include capture, compare and RTCC.

To achieve more selective power savings, peripheral modules can also be selectively disabled when the device enters Idle mode. This is done through the control bit of the generic name format, "XXXIDL". By default, all modules that can operate during Idle mode will do so. Using the disable on Idle feature disables the module while in Idle mode, allowing further reduction of power consumption during Idle mode, enhancing power savings for extremely critical power applications.

NOTES:

11.0 I/O PORTS

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"*, **"I/O Ports with Peripheral Pin Select (PPS)"** (DS39711). The information in this data sheet supersedes the information in the FRM.

All of the device pins (except VDD, VSS, MCLR and OSCI/CLKI) are shared between the peripherals and the Parallel I/O ports. All I/O input ports feature Schmitt Trigger (ST) inputs for improved noise immunity.

11.1 Parallel I/O (PIO) Ports

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

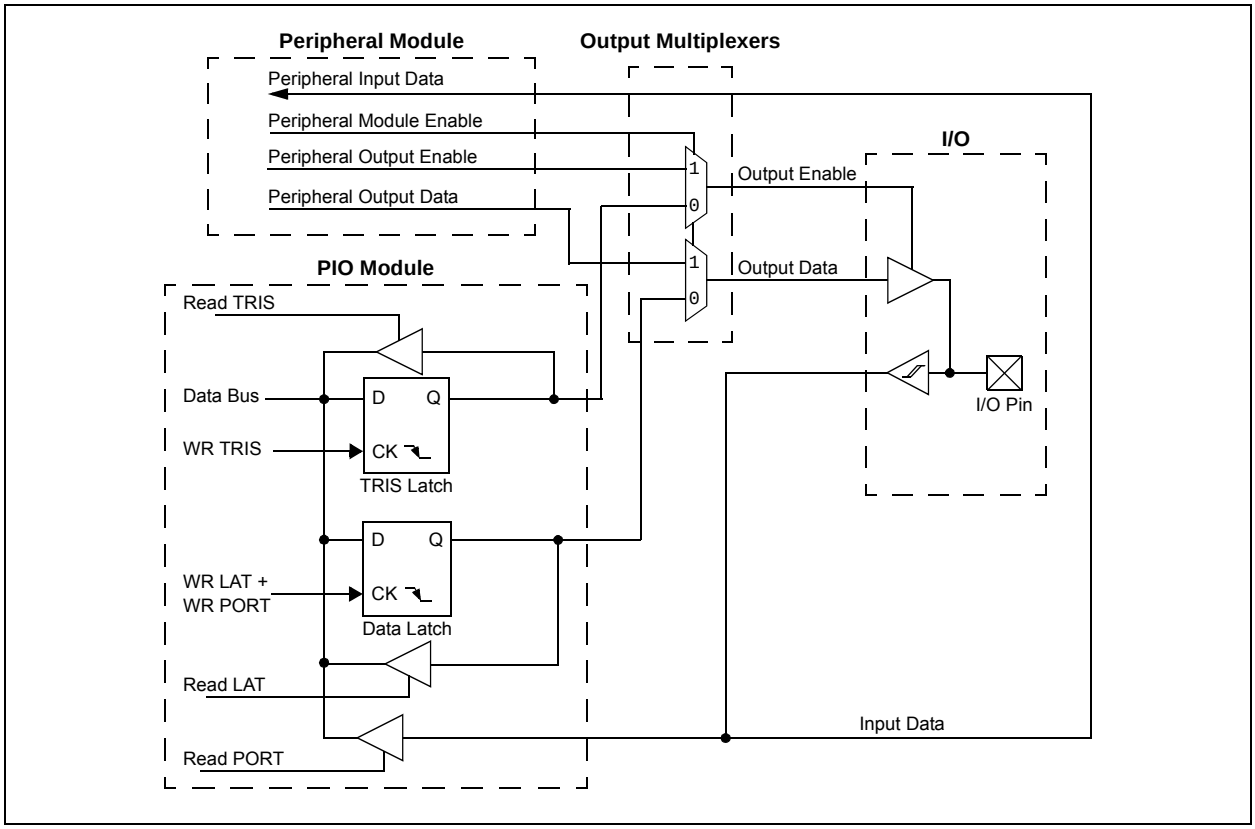
When a peripheral is enabled and the peripheral is actively driving an associated pin, the use of the pin as a general purpose output pin is disabled. The I/O pin may be read, but the output driver for the parallel port bit will be disabled. If a peripheral is enabled, but the peripheral is not actively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

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

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

When a pin is shared with another peripheral or function that is defined as an input only, it is regarded as a dedicated port because there is no other competing source of inputs. RC13 and RC14 can be input ports only; they cannot be configured as outputs.

FIGURE 11-1: BLOCK DIAGRAM OF A TYPICAL SHARED PORT STRUCTURE



11.1.1 I/O PORT WRITE/READ TIMING

One instruction cycle is required between a port direction change or port write operation and a read operation of the same port. Typically, this instruction would be a NOP.

11.1.2 OPEN-DRAIN CONFIGURATION

In addition to the PORTx, LATx and TRISx registers for data control, each port pin can also be individually configured for either a digital or open-drain output. This is controlled by the Open-Drain Control register, ODCx, associated with each port. Setting any of the bits configures the corresponding pin to act as an open-drain output.

The open-drain feature allows the generation of outputs higher than VDD (e.g., 5V) on any desired digital only pins by using external pull-up resistors. The maximum open-drain voltage allowed is the same as the maximum VIH specification.

11.1.3 GPIO FUNCTIONS ON THE USB PINS

When the USB module is enabled, the USB module controls the RG2/RG3/RF7 port pins. General purpose input/output and related interrupt-on-change functionality can be made available on the RG2/RG3/RF7 pins when the USB module is disabled and the UTRDIS (U1CNFG2<0>) bit is set. Additionally, for general purpose digital input function on RF7/VBUS, the ANSF7 bit must be cleared.

11.2 Configuring Analog Port Pins (ANSx)

The ANSx and TRISx registers control the operation of the pins with analog function. Each port pin with analog function is associated with one of the ANSx bits (see Register 11-1 through Register 11-7), which decides if the pin function should be analog or digital. Refer to Table 11-1 for detailed behavior of the pin for different ANSx and TRISx bit settings.

When reading the PORTx register, all pins configured as analog input channels will read as cleared (a low level).

11.2.1 ANALOG INPUT PINS AND VOLTAGE CONSIDERATIONS

The voltage tolerance of pins used as device inputs is dependent on the pin's input function. Most input pins are able to handle DC voltages of up to 5.5V, a level typical for digital logic circuits. However, several pins can only tolerate voltages up to VDD. Voltage excursions beyond VDD on these pins should always be avoided.

Table 11-2 summarizes the different voltage tolerances. Refer to **Section 37.0 "Electrical Characteristics"** for more details.

TABLE 11-1: CONFIGURING ANALOG/DIGITAL FUNCTION OF AN I/O PIN

Pin Function	ANSx Setting	TRISx Setting	Comments
Analog Input	1	1	It is recommended to keep ANSx = 1.
Analog Output	1	1	It is recommended to keep ANSx = 1.
Digital Input	0	1	Firmware must wait at least one instruction cycle after configuring a pin as a digital input before a valid input value can be read.
Digital Output	0	0	Make sure to disable the analog output function on the pin if any is present.

TABLE 11-2: INPUT VOLTAGE LEVELS FOR PORT OR PIN TOLERATED DESCRIPTION INPUT

Port or Pin	Tolerated Input	Description
PORTA<15:14,7:0>(1)	5.5V	Tolerates input levels above VDD; useful for most standard logic.
PORTB<15:14,12,7,4,2>		
PORTC<4:1>(1)		
PORTD<15:0>(1)		
PORTE<9:8,4:0>(1)		
PORTF<13:12,8:7,5:0>(1)		
PORTG<15:12,1:0>(1)		
PORTA<10:9>	VDD	Only VDD input levels are tolerated.
PORTB<13,6:5,3,1:0>		
PORTC<15:12>(1)		
PORTE<7:5>		
PORTG<9:6,3:2>		

Note 1: Not all of these pins are implemented in 64-pin devices. Refer to **Section 1.0 "Device Overview"** for a complete description of port pin implementation.

REGISTER 11-1: ANSA: PORTA ANALOG FUNCTION SELECTION REGISTER

R/W-1	R/W-1	U-0	U-0	U-0	R/W-1	R/W-1	U-0
ANSA15(1)	ANSA14(1)	—	—	—	ANSA10(1)	ANSA9(1)	—
bit 15							bit 8

R/W-1	R/W-1	R/W-1	R/W-1	U-0	U-0	R/W-1	U-0
ANSA7(1)	ANSA6(1)	ANSA5(1)	ANSA4(1)	—	—	ANSA1(1)	—
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 15-14 **ANSA<15:14>:** Analog Function Selection bits(1)
1 = Pin is configured in Analog mode; I/O port read is disabled
0 = Pin is configured in Digital mode; I/O port read is enabled

bit 13-11 **Unimplemented:** Read as '0'

bit 10-9 **ANSA<10:9>:** Analog Function Selection bits(1)
1 = Pin is configured in Analog mode; I/O port read is disabled
0 = Pin is configured in Digital mode; I/O port read is enabled

bit 8 **Unimplemented:** Read as '0'

bit 7-4 **ANSA<7:4>:** Analog Function Selection bits(1)
1 = Pin is configured in Analog mode; I/O port read is disabled
0 = Pin is configured in Digital mode; I/O port read is enabled

bit 3-2 **Unimplemented:** Read as '0'

bit 1 **ANSA1:** Analog Function Selection bit(1)
1 = Pin is configured in Analog mode; I/O port read is disabled
0 = Pin is configured in Digital mode; I/O port read is enabled

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

Note 1: These bits are not available in 64-pin devices.

REGISTER 11-2: ANSB: PORTB ANALOG FUNCTION SELECTION REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	U-0	U-0	U-0	U-0
ANSB15	ANSB14	ANSB13	ANSB12	—	—	—	—
bit 15							bit 8

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
ANSB7	ANSB6	ANSB5	ANSB4	ANSB3	ANSB2	ANSB1	ANSB0
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 **ANSB<15:12>:** Analog Function Selection bits
1 = Pin is configured in Analog mode; I/O port read is disabled
0 = Pin is configured in Digital mode; I/O port read is enabled

bit 11-8 **Unimplemented:** Read as '0'

bit 7-0 **ANSB<7:0>:** Analog Function Selection bits
1 = Pin is configured in Analog mode; I/O port read is disabled
0 = Pin is configured in Digital mode; I/O port read is enabled

REGISTER 11-3: ANSC: PORTC ANALOG FUNCTION SELECTION REGISTER

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

U-0	U-0	U-0	R/W-1	R/W-1	U-0	R/W-1	U-0
—	—	—	ANSC4(1)	ANSC3(1)	—	ANSC1(1)	—
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 15-5 **Unimplemented:** Read as '0'

bit 4-3 **ANSC<4:3>:** Analog Function Selection bits(1)
1 = Pin is configured in Analog mode; I/O port read is disabled
0 = Pin is configured in Digital mode; I/O port read is enabled

bit 2 **Unimplemented:** Read as '0'

bit 1 **ANSC1:** Analog Function Selection bit(1)
1 = Pin is configured in Analog mode; I/O port read is disabled
0 = Pin is configured in Digital mode; I/O port read is enabled

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

Note 1: These bits are not available in 64-pin devices.

REGISTER 11-4: ANSD: PORTD ANALOG FUNCTION SELECTION REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
ANSD15(1)	ANSD14(1)	ANSD13(1)	ANSD12(1)	ANSD11	ANSD10	ANSD9	ANSD8
bit 15							bit 8

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	U-0	R/W-1
ANSD7	ANSD6	ANSD5	ANSD4	ANSD3	ANSD2	—	ANSD0
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 15-2 **ANSD<15:2>:** Analog Function Selection bits(1)
1 = Pin is configured in Analog mode; I/O port read is disabled
0 = Pin is configured in Digital mode; I/O port read is enabled

bit 1 **Unimplemented:** Read as '0'

bit 0 **ANSD0:** Analog Function Selection bit
1 = Pin is configured in Analog mode; I/O port read is disabled
0 = Pin is configured in Digital mode; I/O port read is enabled

Note 1: These bits are not available in 64-pin devices.

REGISTER 11-5: ANSE: PORTE ANALOG FUNCTION SELECTION REGISTER(1)

U-0	U-0	U-0	U-0	U-0	U-0	R/W-1	U-0
—	—	—	—	—	—	ANSE9	—
bit 15							bit 8

R/W-1	R/W-1	R/W-1	R/W-1	U-0	U-0	U-0	U-0
ANSE7	ANSE6	ANSE5	ANSE4	—	—	—	—
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 15-10 **Unimplemented:** Read as '0'

bit 9 **ANSE9:** Analog Function Selection bit
1 = Pin is configured in Analog mode; I/O port read is disabled
0 = Pin is configured in Digital mode; I/O port read is enabled

bit 8 **Unimplemented:** Read as '0'

bit 7-4 **ANSE<7:4>:** Analog Function Selection bits
1 = Pin is configured in Analog mode; I/O port read is disabled
0 = Pin is configured in Digital mode; I/O port read is enabled

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

Note 1: This register is not available in 64-pin devices.

REGISTER 11-6: ANSF: PORTF ANALOG FUNCTION SELECTION REGISTER

U-0	U-0	R/W-1	U-0	U-0	U-0	U-0	R/W-1
—	—	ANSF13(1)	—	—	—	—	ANSF8(1)
bit 15							bit 8

R/W-1	U-0	R/W-1	R/W-1	R/W-1	R/W-1	U-0	R/W-1
ANSF7	—	ANSF5	ANSF4	ANSF3	ANSF2(1)	—	ANSF0
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 15-14 **Unimplemented:** Read as '0'

bit 13 **ANSF13:** Analog Function Selection bit(1)
1 = Pin is configured in Analog mode; I/O port read is disabled
0 = Pin is configured in Digital mode; I/O port read is enabled

bit 12-9 **Unimplemented:** Read as '0'

bit 8-7 **ANSF<8:7>:** Analog Function Selection bits(1)
1 = Pin is configured in Analog mode; I/O port read is disabled
0 = Pin is configured in Digital mode; I/O port read is enabled

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

bit 5-2 **ANSF<5:2>:** Analog Function Selection bits(1)
1 = Pin is configured in Analog mode; I/O port read is disabled
0 = Pin is configured in Digital mode; I/O port read is enabled

bit 1 **Unimplemented:** Read as '0'

bit 0 **ANSF0:** Analog Function Selection bit
1 = Pin is configured in Analog mode; I/O port read is disabled
0 = Pin is configured in Digital mode; I/O port read is enabled

Note 1: These bits are not available in 64-pin devices.

REGISTER 11-7: ANSG: PORTG ANALOG FUNCTION SELECTION REGISTER

R/W-1	U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-1
ANSG15(1)	—	—	—	—	—	ANSG9	ANSG8
bit 15							bit 8

R/W-1	R/W-1	U-0	U-0	U-0	U-0	U-0	U-0
ANSG7	ANSG6	—	—	—	—	—	—
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 **ANSG15:** Analog Function Selection bit(1)
1 = Pin is configured in Analog mode; I/O port read is disabled
0 = Pin is configured in Digital mode; I/O port read is enabled

bit 14-10 **Unimplemented:** Read as '0'

bit 9-6 **ANSG<9:6>:** Analog Function Selection bits
1 = Pin is configured in Analog mode; I/O port read is disabled
0 = Pin is configured in Digital mode; I/O port read is enabled

bit 5-0 **Unimplemented:** Read as '0'

Note 1: This bit is not available in 64-pin devices.

11.3 Input Change Notification (ICN)

The Input Change Notification function of the I/O ports allows the PIC24FJ128GC010 family of devices to generate interrupt requests to the processor in response to a Change-of-State (COS) on selected input pins. This feature is capable of detecting input Change-of-States, even in Sleep mode, when the clocks are disabled. Depending on the device pin count, there are up to 82 external inputs that may be selected (enabled) for generating an interrupt request on a Change-of-State.

Registers, CNEN1 through CNEN6, contain the interrupt enable control bits for each of the Change Notification (CN) input pins. Setting any of these bits enables a CN interrupt for the corresponding pins.

Each CN pin has both a weak pull-up and a weak pull-down connected to it. The pull-ups act as a current source that is connected to the pin, while the pull-downs act as a current sink that is connected to the pin. These eliminate the need for external resistors when push button or keypad devices are connected. The pull-ups and pull-downs are separately enabled using the CNPU1 through CNPU6 registers (for pull-ups) and the CNPD1 through CNPD6 registers (for pull-downs). Each CN pin has individual control bits for its pull-up and pull-down. Setting a control bit enables the weak pull-up or pull-down for the corresponding pin.

When the internal pull-up is selected, the pin pulls up to V_{DD} – 1.1V (typical). When the internal pull-down is selected, the pin pulls down to V_{SS}.

Note:	Pull-ups on Change Notification pins should always be disabled whenever the port pin is configured as a digital output.

EXAMPLE 11-1: PORT WRITE/READ IN ASSEMBLY

```
MOV   0xFF00, W0   ; Configure PORTB<15:8> as inputs
MOV   W0, TRISB    ; and PORTB<7:0> as outputs
NOP                ; Delay 1 cycle
BTSS  PORTB, #13   ; Next Instruction
```

EXAMPLE 11-2: PORT WRITE/READ IN 'C'

```
TRISB = 0xFF00;              // Configure PORTB<15:8> as inputs and PORTB<7:0> as outputs
Nop();                       // Delay 1 cycle
If (PORTBbits.RB13){ };      // Next Instruction
```

11.4 Peripheral Pin Select (PPS)

A major challenge in general purpose devices is providing the largest possible set of peripheral features while minimizing the conflict of features on I/O pins. In an application that needs to use more than one peripheral multiplexed on a single pin, inconvenient work arounds in application code, or a complete redesign, may be the only option.

The Peripheral Pin Select (PPS) feature provides an alternative to these choices by enabling the user's peripheral set selection and its placement on a wide range of I/O pins. By increasing the pinout options available on a particular device, users can better tailor the microcontroller to their entire application, rather than trimming the application to fit the device.

The Peripheral Pin Select feature operates over a fixed subset of digital I/O pins. Users may independently map the input and/or output of any one of many digital peripherals to any one of these I/O pins. PPS is performed in software and generally does not require the device to be reprogrammed. Hardware safeguards are included that prevent accidental or spurious changes to the peripheral mapping once it has been established.

11.4.1 AVAILABLE PINS

The PPS feature is used with a range of up to 44 pins, depending on the particular device and its pin count. Pins that support the Peripheral Pin Select feature include the designation, "RPn" or "RPIn", in their full pin designation, where "n" is the remappable pin number. "RP" is used to designate pins that support both remappable input and output functions, while "RPI" indicates pins that support remappable input functions only.

PIC24FJ128GC010 family devices support a larger number of remappable input only pins than remappable input/output pins. In this device family, there are up to 32 remappable input/output pins, depending on the pin count of the particular device selected. These pins are numbered, RP0 through RP31. Remappable input only pins are numbered above this range, from RPI32 to RPI43 (or the upper limit for that particular device).

See Table 1-3 for a summary of pinout options in each package offering.

11.4.2 AVAILABLE PERIPHERALS

The peripherals managed by the PPS are all digital only peripherals. These include general serial communications (UART and SPI), general purpose timer clock inputs, timer related peripherals (input capture and output compare) and external interrupt inputs. Also included are the outputs of the comparator module, since these are discrete digital signals.

PPS is not available for these peripherals:

- I^2C™ (input and output)
- USB (all module inputs and outputs)
- Change Notification inputs
- RTCC alarm output(s)
- EPMP signals (input and output)
- LCD signals
- Analog (inputs and outputs)
- INT0

A key difference between pin select and non-pin select peripherals is that pin select peripherals are not associated with a default I/O pin. The peripheral must always be assigned to a specific I/O pin before it can be used. In contrast, non-pin select peripherals are always available on a default pin, assuming that the peripheral is active and not conflicting with another peripheral.

11.4.2.1 Peripheral Pin Select Function Priority

Pin-selectable peripheral outputs (e.g., OC, UART transmit) will take priority over general purpose digital functions on a pin, such as EPMP and port I/O. Specialized digital outputs (e.g., USB on USB-enabled devices) will take priority over PPS outputs on the same pin. The pin diagrams list peripheral outputs in the order of priority. Refer to them for priority concerns on a particular pin.

Unlike PIC24F devices with fixed peripherals, pin-selectable peripheral inputs will never take ownership of a pin. The pin's output buffer will be controlled by the TRISx setting or by a fixed peripheral on the pin. If the pin is configured in Digital mode, then the PPS input will operate correctly. If an analog function is enabled on the pin, the PPS input will be disabled.

11.4.3 CONTROLLING PERIPHERAL PIN SELECT

PPS features are controlled through two sets of Special Function Registers (SFRs): one to map peripheral inputs and one to map the outputs. Because they are separately controlled, a particular peripheral's input and output (if the peripheral has both) can be placed on any selectable function pin without constraint.

The association of a peripheral to a peripheral-selectable pin is handled in two different ways, depending on if an input or an output is being mapped.

11.4.3.1 Input Mapping

The inputs of the Peripheral Pin Select options are mapped on the basis of the peripheral; that is, a control register associated with a peripheral dictates the pin it will be mapped to. The RPINRx registers are used to configure peripheral input mapping (see Register 11-8 through Register 11-26).

Each register contains two sets of 6-bit fields, with each set associated with one of the pin-selectable peripherals. Programming a given peripheral's bit field, with an appropriate 6-bit value, maps the RPn/RPIn pin with that value to that peripheral. For any given device, the valid range of values for any of the bit fields corresponds to the maximum number of Peripheral Pin Selections supported by the device.

TABLE 11-3: SELECTABLE INPUT SOURCES (MAPS INPUT TO FUNCTION)(1)

Input Name	Function Name	Register	Function Mapping Bits
DSM Modulation Input	MDMIN	RPINR30	MDMIR<5:0>
DSM Carrier 1 Input	MDCIN1	RPINR31	MDC1R<5:0>
DSM Carrier 2 Input	MDCIN2	RPINR31	MDC2R<5:0>
External Interrupt 1	INT1	RPINR0	INT1R<5:0>
External Interrupt 2	INT2	RPINR1	INT2R<5:0>
External Interrupt 3	INT3	RPINR1	INT3R<5:0>
External Interrupt 4	INT4	RPINR2	INT4R<5:0>
Input Capture 1	IC1	RPINR7	IC1R<5:0>
Input Capture 2	IC2	RPINR7	IC2R<5:0>
Input Capture 3	IC3	RPINR8	IC3R<5:0>
Input Capture 4	IC4	RPINR8	IC4R<5:0>
Input Capture 5	IC5	RPINR9	IC5R<5:0>
Input Capture 6	IC6	RPINR9	IC6R<5:0>
Input Capture 7	IC7	RPINR10	IC7R<5:0>
Input Capture 8	IC8	RPINR10	IC8R<5:0>
Input Capture 9	IC9	RPINR15	IC9R<5:0>
Output Compare Fault A	OCFA	RPINR11	OCFAR<5:0>
Output Compare Fault B	OCFB	RPINR11	OCFBR<5:0>
SPI1 Clock Input	SCK1IN	RPINR20	SCK1R<5:0>
SPI1 Data Input	SDI1	RPINR20	SDI1R<5:0>
SPI1 Slave Select Input	SS1IN	RPINR21	SS1R<5:0>
SPI2 Clock Input	SCK2IN	RPINR22	SCK2R<5:0>
SPI2 Data Input	SDI2	RPINR22	SDI2R<5:0>
SPI2 Slave Select Input	SS2IN	RPINR23	SS2R<5:0>
Generic Timer External Clock	TMRCK	RPINR23	TMRCK<5:0>
UART1 Clear-to-Send	$\overline{\text{U1CTS}}$	RPINR18	U1CTSR<5:0>
UART1 Receive	U1RX	RPINR18	U1RXR<5:0>
UART2 Clear-to-Send	$\overline{\text{U2CTS}}$	RPINR19	U2CTSR<5:0>
UART2 Receive	U2RX	RPINR19	U2RXR<5:0>
UART3 Clear-to-Send	$\overline{\text{U3CTS}}$	RPINR21	U3CTSR<5:0>
UART3 Receive	U3RX	RPINR17	U3RXR<5:0>
UART4 Clear-to-Send	$\overline{\text{U4CTS}}$	RPINR27	U4CTSR<5:0>
UART4 Receive	U4RX	RPINR27	U4RXR<5:0>

Note 1: Unless otherwise noted, all inputs use the Schmitt Trigger (ST) input buffers.

11.4.3.2 Output Mapping

In contrast to inputs, the outputs of the Peripheral Pin Select options are mapped on the basis of the pin. In this case, a control register associated with a particular pin dictates the peripheral output to be mapped. The RPORx registers are used to control output mapping. Each register contains two 6-bit fields, with each field being associated with one RPn pin (see Register 11-27 through Register 11-42). The value of the bit field corresponds to one of the peripherals and that peripheral's output is mapped to the pin (see Table 11-4).

Because of the mapping technique, the list of peripherals for output mapping also includes a null value of '000000'. This permits any given pin to remain disconnected from the output of any of the pin-selectable peripherals.

TABLE 11-4: SELECTABLE OUTPUT SOURCES (MAPS FUNCTION TO OUTPUT)

Output Function Number(1)	Function	Output Name
0	NULL(2)	Null
1	C1OUT	Comparator 1 Output
2	C2OUT	Comparator 2 Output
3	U1TX	UART1 Transmit
4	$\overline{\text{U1RTS}}$(3)	UART1 Request-to-Send
5	U2TX	UART2 Transmit
6	$\overline{\text{U2RTS}}$(3)	UART2 Request-to-Send
7	SDO1	SPI1 Data Output
8	SCK1OUT	SPI1 Clock Output
9	SS1OUT	SPI1 Slave Select Output
10	SDO2	SPI2 Data Output
11	SCK2OUT	SPI2 Clock Output
12	SS2OUT	SPI2 Slave Select Output
18	OC1	Output Compare 1
19	OC2	Output Compare 2
20	OC3	Output Compare 3
21	OC4	Output Compare 4
22	OC5	Output Compare 5
23	OC6	Output Compare 6
24	OC7	Output Compare 7
25	OC8	Output Compare 8
28	U3TX	UART3 Transmit
29	$\overline{\text{U3RTS}}$(3)	UART3 Request-to-Send
30	U4TX	UART4 Transmit
31	$\overline{\text{U4RTS}}$(3)	UART4 Request-to-Send
35	OC9	Output Compare 9
36	C3OUT	Comparator 3 Output
37	MDOUT	DSM Modulator Output
38-63	(unused)	NC

Note 1: Setting the RPORx register with the listed value assigns that output function to the associated RPn pin.

2: The NULL function is assigned to all RPn outputs at device Reset and disables the RPn output function.

3: IrDA® BCLKx functionality uses this output.

11.4.3.3 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.

11.4.3.4 Mapping Exceptions for PIC24FJ128GC010 Family Devices

Although the PPS registers theoretically allow for up to 64 remappable I/O pins, not all of these are implemented in all devices. For PIC24FJ128GC010 family devices, the maximum number of remappable pins available is 44, which includes 12 input only pins. In addition, some pins in the RP and RPI sequences are unimplemented in lower pin count devices. The differences in available remappable pins are summarized in Table 11-5.

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 all PIC24FJ128GC010 family devices, this includes all values greater than 43 (‘101011’).
- For RPORx registers, the bit fields corresponding to an unimplemented pin will also be unimplemented. Writing to these fields will have no effect.

11.4.4 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

11.4.4.1 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 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.

11.4.4.2 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.

11.4.4.3 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 (CW4<15>) 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.

TABLE 11-5: REMAPPABLE PIN EXCEPTIONS FOR PIC24FJ128GC010 FAMILY DEVICES

Device	RP Pins (I/O)		RPI Pins	
	Total	Unimplemented	Total	Unimplemented
PIC24FJXXXGC006	28	RP5, RP15, RP30, RP31	1	RPI32-36, RPI38-43
PIC24FJXXXGC010	32	—	12	—

11.4.5 CONSIDERATIONS FOR PERIPHERAL PIN SELECTION

The ability to control Peripheral Pin Selection introduces several considerations into application design that could be overlooked. This is particularly true for several common peripherals that are available only as remappable peripherals.

The main consideration is that the Peripheral Pin Selects are not available on default pins in the device's default (Reset) state. Since all RPINRx registers reset to '111111' and all RPORx registers reset to '000000', all Peripheral Pin Select inputs are tied to Vss and all Peripheral Pin Select outputs are disconnected.

Note:	In tying Peripheral Pin Select inputs to RP63, the RP63 pin need not exist on a device for the registers to be reset to it.

This situation requires the user to initialize the device with the proper peripheral configuration before any other application code is executed. Since the IOLOCK bit resets in the unlocked state, it is not necessary to execute the unlock sequence after the device has come out of Reset. For application safety, however, it is best to set IOLOCK and lock the configuration after writing to the control registers.

Because the unlock sequence is timing-critical, it must be executed as an assembly language routine in the same manner as changes to the oscillator configuration. If the bulk of the application is written in 'C', or another high-level language, the unlock sequence should be performed by writing in-line assembly.

Choosing the configuration requires a review of all Peripheral Pin Selects and their pin assignments, especially those that will not be used in the application. In all cases, unused pin-selectable peripherals should be disabled completely. Unused peripherals should have their inputs assigned to an unused RPn/RPIn pin function. I/O pins with unused RPn functions should be configured with the null peripheral output.

The assignment of a peripheral to a particular pin does not automatically perform any other configuration of the pin's I/O circuitry. In theory, this means adding a pin-selectable output to a pin may mean inadvertently driving an existing peripheral input when the output is driven. Users must be familiar with the behavior of other fixed peripherals that share a remappable pin and know when to enable or disable them. To be safe, fixed digital peripherals that share the same pin should be disabled when not in use.

Along these lines, configuring a remappable pin for a specific peripheral does not automatically turn that feature on. The peripheral must be specifically configured for operation and enabled as if it were tied to a fixed pin. Where this happens in the application code (immediately following a device Reset and peripheral configuration or inside the main application routine) depends on the peripheral and its use in the application.

A final consideration is that Peripheral Pin Select functions neither override analog inputs nor reconfigure pins with analog functions for digital I/O. If a pin is configured as an analog input on device Reset, it must be explicitly reconfigured as digital I/O when used with a Peripheral Pin Select.

Example 11-3 shows a configuration for bidirectional communication with flow control using UART1. The following input and output functions are used:

- Input Functions: U1RX, $\overline{\text{U1CTS}}$
- Output Functions: U1TX, $\overline{\text{U1RTS}}$

EXAMPLE 11-3: CONFIGURING UART1 INPUT AND OUTPUT FUNCTIONS

```
// Unlock Registers
asm volatile   ("MOV   #OSCCON,   w1   \n"
                "MOV   #0x46,     w2   \n"
                "MOV   #0x57,     w3   \n"
                "MOV.b w2,        [w1] \n"
                "MOV.b w3,        [w1] \n"
                "BCLR  OSCCON,    #6") ;

// or use C30 built-in macro:
// __builtin_write_OSCCONL(OSCCON & 0xbf);

// Configure Input Functions (Table 11-2))
    // Assign U1RX To Pin RP0
    RPINR18bits.U1RXR = 0;

    // Assign U1CTS To Pin RP1
    RPINR18bits.U1CTSR = 1;

// Configure Output Functions (Table 11-4)
    // Assign U1TX To Pin RP2
    RPOR1bits.RP2R = 3;

    // Assign U1RTS To Pin RP3
    RPOR1bits.RP3R = 4;

// Lock Registers
asm volatile   ("MOV   #OSCCON,   w1   \n"
                "MOV   #0x46,     w2   \n"
                "MOV   #0x57,     w3   \n"
                "MOV.b w2,        [w1] \n"
                "MOV.b w3,        [w1] \n"
                "BSET  OSCCON,     #6");

// or use C30 built-in macro:
// __builtin_write_OSCCONL(OSCCON | 0x40);
```

11.4.6 PERIPHERAL PIN SELECT REGISTERS

The PIC24FJ128GC010 family of devices implements a total of 35 registers for remappable peripheral configuration:

- Input Remappable Peripheral Registers (19)
- Output Remappable Peripheral Registers (16)

Note: Input and output register values can only be changed if IOLOCK (OSCCON<6>) = 0. See **Section 11.4.4.1 "Control Register Lock"** for a specific command sequence.

REGISTER 11-8: RPINR0: PERIPHERAL PIN SELECT INPUT REGISTER 0

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	INT1R5	INT1R4	INT1R3	INT1R2	INT1R1	INT1R0
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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **INT1R<5:0>:** Assign External Interrupt 1 (INT1) to Corresponding RPn or RPIn Pin bits

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

REGISTER 11-9: RPINR1: PERIPHERAL PIN SELECT INPUT REGISTER 1

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	INT3R5	INT3R4	INT3R3	INT3R2	INT3R1	INT3R0
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
—	—	INT2R5	INT2R4	INT2R3	INT2R2	INT2R1	INT2R0
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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **INT3R<5:0>:** Assign External Interrupt 3 (INT3) to Corresponding RPn or RPIn Pin bits

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

bit 5-0 **INT2R<5:0>:** Assign External Interrupt 2 (INT2) to Corresponding RPn or RPIn Pin bits

REGISTER 11-10: RPINR2: PERIPHERAL PIN SELECT INPUT REGISTER 2

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
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
—	—	INT4R5	INT4R4	INT4R3	INT4R2	INT4R1	INT4R0
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 15-6 **Unimplemented:** Read as '0'

bit 5-0 **INT4R<5:0>:** Assign External Interrupt 4 (INT4) to Corresponding RPn or RPIn Pin bits

REGISTER 11-11: RPINR7: PERIPHERAL PIN SELECT INPUT REGISTER 7

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	IC2R5	IC2R4	IC2R3	IC2R2	IC2R1	IC2R0
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
—	—	IC1R5	IC1R4	IC1R3	IC1R2	IC1R1	IC1R0
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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **IC2R<5:0>:** Assign Input Capture 2 (IC2) to Corresponding RPn or RPIn Pin bits

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

bit 5-0 **IC1R<5:0>:** Assign Input Capture 1 (IC1) to Corresponding RPn or RPIn Pin bits

REGISTER 11-12: RPINR8: PERIPHERAL PIN SELECT INPUT REGISTER 8

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	IC4R5	IC4R4	IC4R3	IC4R2	IC4R1	IC4R0
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
—	—	IC3R5	IC3R4	IC3R3	IC3R2	IC3R1	IC3R0
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 15-14 **Unimplemented:** Read as ‘0’

bit 13-8 **IC4R<5:0>:** Assign Input Capture 4 (IC4) to Corresponding RPn or RPIn Pin bits

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-0 **IC3R<5:0>:** Assign Input Capture 3 (IC3) to Corresponding RPn or RPIn Pin bits

REGISTER 11-13: RPINR9: PERIPHERAL PIN SELECT INPUT REGISTER 9

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	IC6R5	IC6R4	IC6R3	IC6R2	IC6R1	IC6R0
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
—	—	IC5R5	IC5R4	IC5R3	IC5R2	IC5R1	IC5R0
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 15-14 **Unimplemented:** Read as ‘0’

bit 13-8 **IC6R<5:0>:** Assign Input Capture 6 (IC6) to Corresponding RPn or RPIn Pin bits

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-0 **IC5R<5:0>:** Assign Input Capture 5 (IC5) to Corresponding RPn or RPIn Pin bits

REGISTER 11-14: RPINR10: PERIPHERAL PIN SELECT INPUT REGISTER 10

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	IC8R5	IC8R4	IC8R3	IC8R2	IC8R1	IC8R0
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
—	—	IC7R5	IC7R4	IC7R3	IC7R2	IC7R1	IC7R0
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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **IC8R<5:0>:** Assign Input Capture 8 (IC8) to Corresponding RPn or RPIn Pin bits

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

bit 5-0 **IC7R<5:0>:** Assign Input Capture 7 (IC7) to Corresponding RPn or RPIn Pin bits

REGISTER 11-15: RPINR11: PERIPHERAL PIN SELECT INPUT REGISTER 11

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	OCFBR5	OCFBR4	OCFBR3	OCFBR2	OCFBR1	OCFBR0
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
—	—	OCFAR5	OCFAR4	OCFAR3	OCFAR2	OCFAR1	OCFAR0
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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **OCFBR<5:0>:** Assign Output Compare Fault B (OCFB) to Corresponding RPn or RPIn Pin bits

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

bit 5-0 **OCFAR<5:0>:** Assign Output Compare Fault A (OCFA) to Corresponding RPn or RPIn Pin bits

REGISTER 11-16: RPINR15: PERIPHERAL PIN SELECT INPUT REGISTER 15

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	IC9R5	IC9R4	IC9R3	IC9R2	IC9R1	IC9R0
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 15-14 **Unimplemented:** Read as ‘0’

bit 13-8 **IC9R<5:0>:** Assign Input Capture 9 (IC9) to Corresponding RPn or RPIn Pin bits

bit 7-0 **Unimplemented:** Read as ‘0’

REGISTER 11-17: RPINR17: PERIPHERAL PIN SELECT INPUT REGISTER 17

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	U3RXR5	U3RXR4	U3RXR3	U3RXR2	U3RXR1	U3RXR0
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 15-14 **Unimplemented:** Read as ‘0’

bit 13-8 **U3RXR<5:0>:** Assign UART3 Receive (U3RX) to Corresponding RPn or RPIn Pin bits

bit 7-0 **Unimplemented:** Read as ‘0’

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

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
—	—	U1RXR5	U1RXR4	U1RXR3	U1RXR2	U1RXR1	U1RXR0
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 15-14 **Unimplemented:** Read as ‘0’

bit 13-8 **U1CTSR<5:0>:** Assign UART1 Clear-to-Send ($\overline{\text{U1CTS}}$) 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 11-19: 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 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 **U2CTSR<5:0>:** Assign UART2 Clear-to-Send ($\overline{\text{U2CTS}}$) to Corresponding RPn or RPIn Pin bits

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-0 **U2RXR<5:0>:** Assign UART2 Receive (U2RX) to Corresponding RPn or RPIn Pin bits

REGISTER 11-20: RPINR20: PERIPHERAL PIN SELECT INPUT REGISTER 20

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	SCK1R5	SCK1R4	SCK1R3	SCK1R2	SCK1R1	SCK1R0
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
—	—	SDI1R5	SDI1R4	SDI1R3	SDI1R2	SDI1R1	SDI1R0
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 15-14 **Unimplemented:** Read as ‘0’

bit 13-8 **SCK1R<5:0>:** Assign SPI1 Clock Input (SCK1IN) to Corresponding RPn or RPIn Pin bits

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-0 **SDI1R<5:0>:** Assign SPI1 Data Input (SDI1) to Corresponding RPn or RPIn Pin bits

REGISTER 11-21: RPINR21: PERIPHERAL PIN SELECT INPUT REGISTER 21

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	U3CTSR5	U3CTSR4	U3CTSR3	U3CTSR2	U3CTSR1	U3CTSR0
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
—	—	SS1R5	SS1R4	SS1R3	SS1R2	SS1R1	SS1R0
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 15-14 **Unimplemented:** Read as ‘0’

bit 13-8 **U3CTSR<5:0>:** Assign UART3 Clear-to-Send ($\overline{\text{U3CTS}}$) to Corresponding RPn or RPIn Pin bits

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-0 **SS1R<5:0>:** Assign SPI1 Slave Select Input (SS1IN) to Corresponding RPn or RPIn Pin bits

REGISTER 11-22: RPINR22: PERIPHERAL PIN SELECT INPUT REGISTER 22

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	SCK2R5	SCK2R4	SCK2R3	SCK2R2	SCK2R1	SCK2R0
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
—	—	SDI2R5	SDI2R4	SDI2R3	SDI2R2	SDI2R1	SDI2R0
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 15-14 **Unimplemented:** Read as ‘0’

bit 13-8 **SCK2R<5:0>:** Assign SPI2 Clock Input (SCK2IN) to Corresponding RPn or RPIn Pin bits

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-0 **SDI2R<5:0>:** Assign SPI2 Data Input (SDI2) to Corresponding RPn or RPIn Pin bits

REGISTER 11-23: RPINR23: PERIPHERAL PIN SELECT INPUT REGISTER 23

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	TMRCKR5	TMRCKR4	TMRCKR3	TMRCKR2	TMRCKR1	TMRCKR0
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
—	—	SS2R5	SS2R4	SS2R3	SS2R2	SS2R1	SS2R0
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 15-14 **Unimplemented:** Read as ‘0’

bit 13-8 **TMRCK<5:0>:** Assign General Timer External Input (TMRCK) to Corresponding RPn or RPIn Pin bits

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-0 **SS2R<5:0>:** Assign SPI2 Slave Select Input (SS2IN) to Corresponding RPn or RPIn Pin bits

REGISTER 11-24: RPINR27: PERIPHERAL PIN SELECT INPUT REGISTER 27

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	U4CTSR5	U4CTSR4	U4CTSR3	U4CTSR2	U4CTSR1	U4CTSR0
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
—	—	U4RXR5	U4RXR4	U4RXR3	U4RXR2	U4RXR1	U4RXR0
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 15-14 **Unimplemented:** Read as ‘0’

bit 13-8 **U4CTSR<5:0>:** Assign UART4 Clear-to-Send Input (U4CTS) to Corresponding RPn or RPIn Pin bits

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-0 **U4RXR<5:0>:** Assign UART4 Receive Input (U4RX) to Corresponding RPn or RPIn Pin bits

REGISTER 11-25: RPINR30: PERIPHERAL PIN SELECT INPUT REGISTER 30

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
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
—	—	MDMIR5	MDMIR4	MDMIR3	MDMIR2	MDMIR1	MDMIR0
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 15-6 **Unimplemented:** Read as ‘0’

bit 5-0 **MDMIR<5:0>:** Assign TX Modulation Input (MDMI) to Corresponding RPn or RPIn Pin bits

REGISTER 11-26: RPINR31: PERIPHERAL PIN SELECT INPUT REGISTER 31

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	MDC2R5	MDC2R4	MDC2R3	MDC2R2	MDC2R1	MDC2R0
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
—	—	MDC1R5	MDC1R4	MDC1R3	MDC1R2	MDC21R1	MDC1R0
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 15-14 **Unimplemented:** Read as ‘0’

bit 13-8 **MDC2R<5:0>:** Assign TX Carrier 2 Input (MDCIN2) to Corresponding RPn or RPIn Pin bits

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-0 **MDC1R<5:0>:** Assign TX Carrier 1 Input (MDCIN1) to Corresponding RPn or RPIn Pin bits

REGISTER 11-27: RPOR0: PERIPHERAL PIN SELECT OUTPUT REGISTER 0

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP1R5	RP1R4	RP1R3	RP1R2	RP1R1	RP1R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP0R5	RP0R4	RP0R3	RP0R2	RP0R1	RP0R0
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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **RP1R<5:0>:** RP1 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP1 (see Table 11-4 for peripheral function numbers).

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

bit 5-0 **RP0R<5:0>:** RP0 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP0 (see Table 11-4 for peripheral function numbers).

REGISTER 11-28: RPOR1: PERIPHERAL PIN SELECT OUTPUT REGISTER 1

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP3R5	RP3R4	RP3R3	RP3R2	RP3R1	RP3R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP2R5	RP2R4	RP2R3	RP2R2	RP2R1	RP2R0
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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **RP3R<5:0>:** RP3 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP3 (see Table 11-4 for peripheral function numbers).

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

bit 5-0 **RP2R<5:0>:** RP2 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP2 (see Table 11-4 for peripheral function numbers).

REGISTER 11-29: RPOR2: PERIPHERAL PIN SELECT OUTPUT REGISTER 2

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP5R5(1)	RP5R4(1)	RP5R3(1)	RP5R2(1)	RP5R1(1)	RP5R0(1)
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP4R5	RP4R4	RP4R3	RP4R2	RP4R1	RP4R0
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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **RP5R<5:0>:** RP5 Output Pin Mapping bits(1)
Peripheral Output Number n is assigned to pin, RP5 (see Table 11-4 for peripheral function numbers).

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

bit 5-0 **RP4R<5:0>:** RP4 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP4 (see Table 11-4 for peripheral function numbers).

Note 1: These bits are unimplemented in 64-pin devices; read as '0'.

REGISTER 11-30: RPOR3: PERIPHERAL PIN SELECT OUTPUT REGISTER 3

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP7R5	RP7R4	RP7R3	RP7R2	RP7R1	RP7R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP6R5	RP6R4	RP6R3	RP6R2	RP6R1	RP6R0
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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **RP7R<5:0>:** RP7 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP7 (see Table 11-4 for peripheral function numbers).

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

bit 5-0 **RP6R<5:0>:** RP6 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP6 (see Table 11-4 for peripheral function numbers).

REGISTER 11-31: RPOR4: PERIPHERAL PIN SELECT OUTPUT REGISTER 4

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP9R5	RP9R4	RP9R3	RP9R2	RP9R1	RP9R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP8R5	RP8R4	RP8R3	RP8R2	RP8R1	RP8R0
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 15-14 **Unimplemented:** Read as ‘0’

bit 13-8 **RP9R<5:0>:** RP9 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP9 (see Table 11-4 for peripheral function numbers).

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-0 **RP8R<5:0>:** RP8 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP8 (see Table 11-4 for peripheral function numbers).

REGISTER 11-32: RPOR5: PERIPHERAL PIN SELECT OUTPUT REGISTER 5

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP11R5	RP11R4	RP11R3	RP11R2	RP11R1	RP11R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP10R5	RP10R4	RP10R3	RP10R2	RP10R1	RP10R0
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 15-14 **Unimplemented:** Read as ‘0’

bit 13-8 **RP11R<5:0>:** RP11 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP11 (see Table 11-4 for peripheral function numbers).

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-0 **RP10R<5:0>:** RP10 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP10 (see Table 11-4 for peripheral function numbers).

REGISTER 11-33: RPOR6: PERIPHERAL PIN SELECT OUTPUT REGISTER 6

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP13R5	RP13R4	RP13R3	RP13R2	RP13R1	RP13R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP12R5	RP12R4	RP12R3	RP12R2	RP12R1	RP12R0
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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **RP13R<5:0>:** RP13 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP13 (see Table 11-4 for peripheral function numbers).

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

bit 5-0 **RP12R<5:0>:** RP12 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP12 (see Table 11-4 for peripheral function numbers).

REGISTER 11-34: RPOR7: PERIPHERAL PIN SELECT OUTPUT REGISTER 7

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP15R5(1)	RP15R4(1)	RP15R3(1)	RP15R2(1)	RP15R1(1)	RP15R0(1)
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP14R5	RP14R4	RP14R3	RP14R2	RP14R1	RP14R0
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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **RP15R<5:0>:** RP15 Output Pin Mapping bits(1)
Peripheral Output Number n is assigned to pin, RP15 (see Table 11-4 for peripheral function numbers).

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

bit 5-0 **RP14R<5:0>:** RP14 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP14 (see Table 11-4 for peripheral function numbers).

Note 1: These bits are unimplemented in 64-pin devices; read as '0'.

REGISTER 11-35: RPOR8: PERIPHERAL PIN SELECT OUTPUT REGISTER 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP17R5	RP17R4	RP17R3	RP17R2	RP17R1	RP17R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP16R5	RP16R4	RP16R3	RP16R2	RP16R1	RP16R0
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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **RP17R<5:0>:** RP17 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP17 (see Table 11-4 for peripheral function numbers).

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

bit 5-0 **RP16R<5:0>:** RP16 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP16 (see Table 11-4 for peripheral function numbers).

REGISTER 11-36: RPOR9: PERIPHERAL PIN SELECT OUTPUT REGISTER 9

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP19R5	RP19R4	RP19R3	RP19R2	RP19R1	RP19R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP18R5	RP18R4	RP18R3	RP18R2	RP18R1	RP18R0
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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **RP19R<5:0>:** RP19 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP19 (see Table 11-4 for peripheral function numbers).

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

bit 5-0 **RP18R<5:0>:** RP18 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP18 (see Table 11-4 for peripheral function numbers).

REGISTER 11-37: RPOR10: PERIPHERAL PIN SELECT OUTPUT REGISTER 10

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP21R5	RP21R4	RP21R3	RP21R2	RP21R1	RP21R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP20R5	RP20R4	RP20R3	RP20R2	RP20R1	RP20R0
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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **RP21R<5:0>:** RP21 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP21 (see Table 11-4 for peripheral function numbers).

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

bit 5-0 **RP20R<5:0>:** RP20 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP20 (see Table 11-4 for peripheral function numbers).

REGISTER 11-38: RPOR11: PERIPHERAL PIN SELECT OUTPUT REGISTER 11

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP23R5	RP23R4	RP23R3	RP23R2	RP23R1	RP23R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP22R5	RP22R4	RP22R3	RP22R2	RP22R1	RP22R0
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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **RP23R<5:0>:** RP23 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP23 (see Table 11-4 for peripheral function numbers).

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

bit 5-0 **RP22R<5:0>:** RP22 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP22 (see Table 11-4 for peripheral function numbers).

REGISTER 11-39: RPOR12: PERIPHERAL PIN SELECT OUTPUT REGISTER 12

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP25R5	RP25R4	RP25R3	RP25R2	RP25R1	RP25R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP24R5	RP24R4	RP24R3	RP24R2	RP24R1	RP24R0
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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **RP25R<5:0>:** RP25 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP25 (see Table 11-4 for peripheral function numbers).

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

bit 5-0 **RP24R<5:0>:** RP24 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP24 (see Table 11-4 for peripheral function numbers).

REGISTER 11-40: RPOR13: PERIPHERAL PIN SELECT OUTPUT REGISTER 13

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP27R5	RP27R4	RP27R3	RP27R2	RP27R1	RP27R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP26R5	RP26R4	RP26R3	RP26R2	RP26R1	RP26R0
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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **RP27R<5:0>:** RP27 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP27 (see Table 11-4 for peripheral function numbers).

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

bit 5-0 **RP26R<5:0>:** RP26 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP26 (see Table 11-4 for peripheral function numbers).

REGISTER 11-41: RPOR14: PERIPHERAL PIN SELECT OUTPUT REGISTER 14

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP29R5	RP29R4	RP29R3	RP29R2	RP29R1	RP29R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP28R5	RP28R4	RP28R3	RP28R2	RP28R1	RP28R0
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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **RP29R<5:0>:** RP29 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP29 (see Table 11-4 for peripheral function numbers).

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

bit 5-0 **RP28R<5:0>:** RP28 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP28 (see Table 11-4 for peripheral function numbers).

REGISTER 11-42: RPOR15: PERIPHERAL PIN SELECT OUTPUT REGISTER 15

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP31R5(1)	RP31R4(1)	RP31R3(1)	RP31R2(1)	RP31R1(1)	RP31R0(1)
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP30R5	RP30R4	RP30R3	RP30R2	RP30R1	RP30R0
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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **RP31R<5:0>:** RP31 Output Pin Mapping bits(1)
Peripheral Output Number n is assigned to pin, RP31 (see Table 11-4 for peripheral function numbers).

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

bit 5-0 **RP30R<5:0>:** RP30 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP30 (see Table 11-4 for peripheral function numbers).

Note 1: These bits are unimplemented in 64-pin devices; read as '0'.

12.0 TIMER1

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*", **"Timers"** (DS39704). The information in this data sheet supersedes the information in the FRM.

The Timer1 module is a 16-bit timer, which can serve as the time counter for the Real-Time Clock (RTC) or operate as a free-running, interval timer/counter. Timer1 can operate in three modes:

- 16-Bit Timer
- 16-Bit Synchronous Counter
- 16-Bit Asynchronous Counter

Timer1 also supports these features:

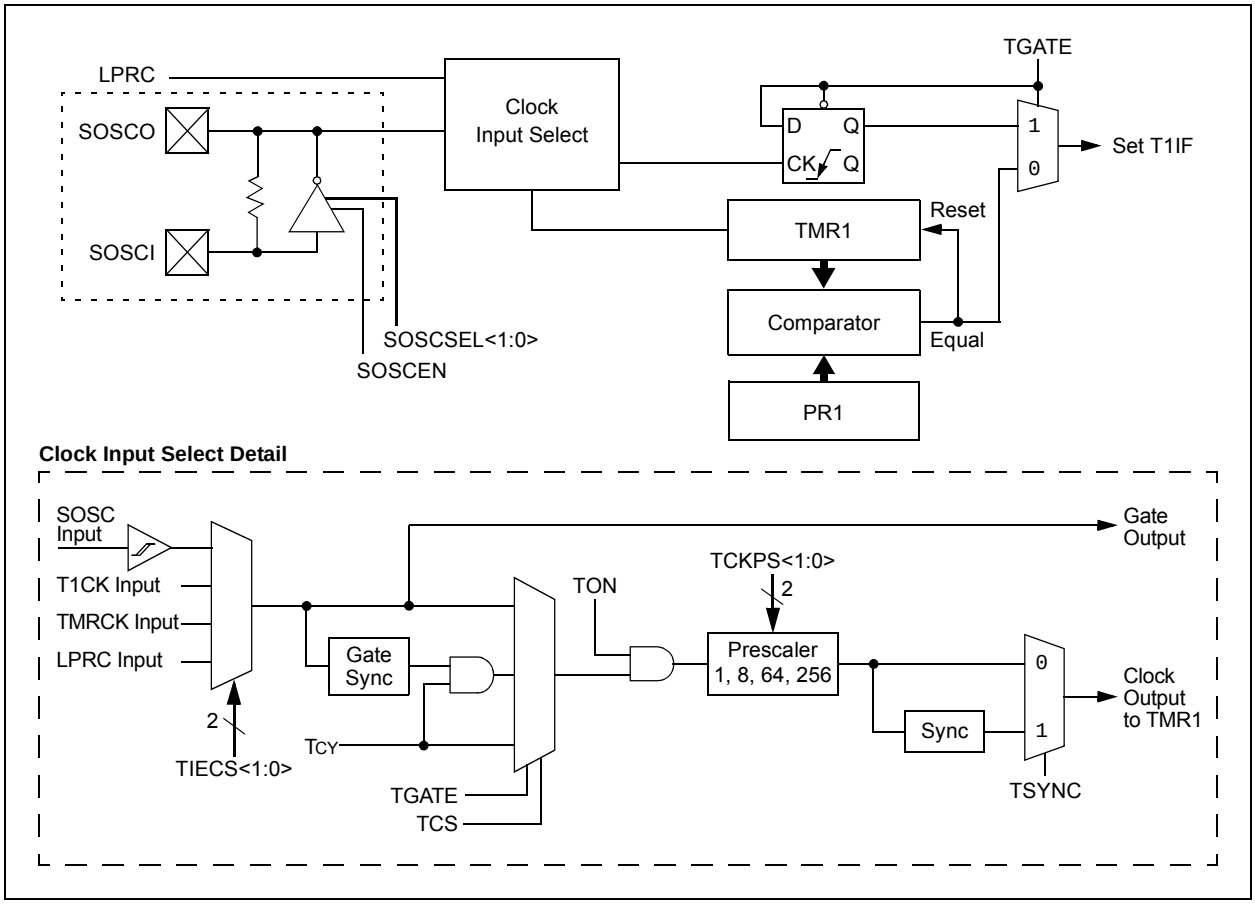
- Timer Gate Operation
- Selectable Prescaler Settings
- Timer Operation During CPU Idle and Sleep modes
- Interrupt on 16-Bit Period Register Match or Falling Edge of External Gate Signal

Figure 12-1 presents a block diagram of the 16-bit timer module.

To configure Timer1 for operation:

1. Set the TON bit (= 1).
2. Select the timer prescaler ratio using the TCKPS<1:0> bits.
3. Set the Clock and Gating modes using the TCS, TIECS<1:0> and TGATE bits.
4. Set or clear the TSYNC bit to configure synchronous or asynchronous operation.
5. Load the timer period value into the PR1 register.
6. If interrupts are required, set the Timer1 Interrupt Enable bit, T1IE. Use the Timer1 Interrupt Priority bits, T1IP<2:0>, to set the interrupt priority.

FIGURE 12-1: 16-BIT TIMER1 MODULE BLOCK DIAGRAM



REGISTER 12-1: T1CON: TIMER1 CONTROL REGISTER(1)

R/W-0	U-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0
TON	—	TSIDL	—	—	—	TIECS1	TIECS0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	U-0
—	TGATE	TCKPS1	TCKPS0	—	TSYNC	TCS	—
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 **TON:** Timer1 On bit
1 = Starts 16-bit Timer1
0 = Stops 16-bit Timer1

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

bit 13 **TSIDL:** Timer1 Stop in Idle Mode bit
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode

bit 12-10 **Unimplemented:** Read as '0'

bit 9-8 **TIECS<1:0>:** Timer1 Extended Clock Source Select bits (selected when TCS = 1)
When TCS = 1:
11 = Generic Timer (TMRCK) External Input
10 = LPRC Oscillator
01 = T1CK External Clock Input
00 = SOSC
When TCS = 0:
These bits are ignored; timer is clocked from internal system clock (FOSC/2).

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

bit 6 **TGATE:** Timer1 Gated Time Accumulation Enable bit
When TCS = 1:
This bit is ignored.
When TCS = 0:
1 = Gated time accumulation is enabled
0 = Gated time accumulation is disabled

bit 5-4 **TCKPS<1:0>:** Timer1 Input Clock Prescale Select bits
11 = 1:256
10 = 1:64
01 = 1:8
00 = 1:1

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

bit 2 **TSYNC:** Timer1 External Clock Input Synchronization Select bit
When TCS = 1:
1 = Synchronizes external clock input
0 = Does not synchronize external clock input
When TCS = 0:
This bit is ignored.

bit 1 **TCS:** Timer1 Clock Source Select bit
1 = Extended clock is selected by the TIECS<1:0> bits
0 = Internal clock (FOSC/2)

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

Note 1: Changing the value of T1CON while the timer is running (TON = 1) causes the timer prescale counter to reset and is not recommended.

13.0 TIMER2/3 AND TIMER4/5

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"*, **"Timers"** (DS39704). The information in this data sheet supersedes the information in the FRM.

The Timer2/3 and Timer4/5 modules are 32-bit timers, which can also be configured as four independent, 16-bit timers with selectable operating modes.

As 32-bit timers, Timer2/3 and Timer4/5 can each operate in three modes:

- Two Independent 16-Bit Timers with All 16-Bit Operating modes (except Asynchronous Counter mode)
- Single 32-Bit Timer
- Single 32-Bit Synchronous Counter

They also support these features:

- Timer Gate Operation
- Selectable Prescaler Settings
- Timer Operation During Idle and Sleep modes
- Interrupt on a 32-Bit Period Register Match
- A/D Event Trigger (only on Timer2/3 in 32-bit mode and Timer3 in 16-bit mode)

Individually, all four of the 16-bit timers can function as synchronous timers or counters. They also offer the features listed above, except for the A/D Event Trigger. This trigger is implemented only on Timer2/3 in 32-bit mode and Timer3 in 16-bit mode. The operating modes and enabled features are determined by setting the appropriate bit(s) in the T2CON, T3CON, T4CON and T5CON registers. T2CON and T4CON are shown in generic form in Register 13-1; T3CON and T5CON are shown in Register 13-2.

For 32-bit timer/counter operation, Timer2 and Timer4 are the least significant word; Timer3 and Timer5 are the most significant word of the 32-bit timers.

Note: For 32-bit operation, T3CON and T5CON control bits are ignored. Only T2CON and T4CON control bits are used for setup and control. Timer2 and Timer4 clock and gate inputs are utilized for the 32-bit timer modules, but an interrupt is generated with the Timer3 or Timer5 interrupt flags.

To configure Timer2/3 or Timer4/5 for 32-bit operation:

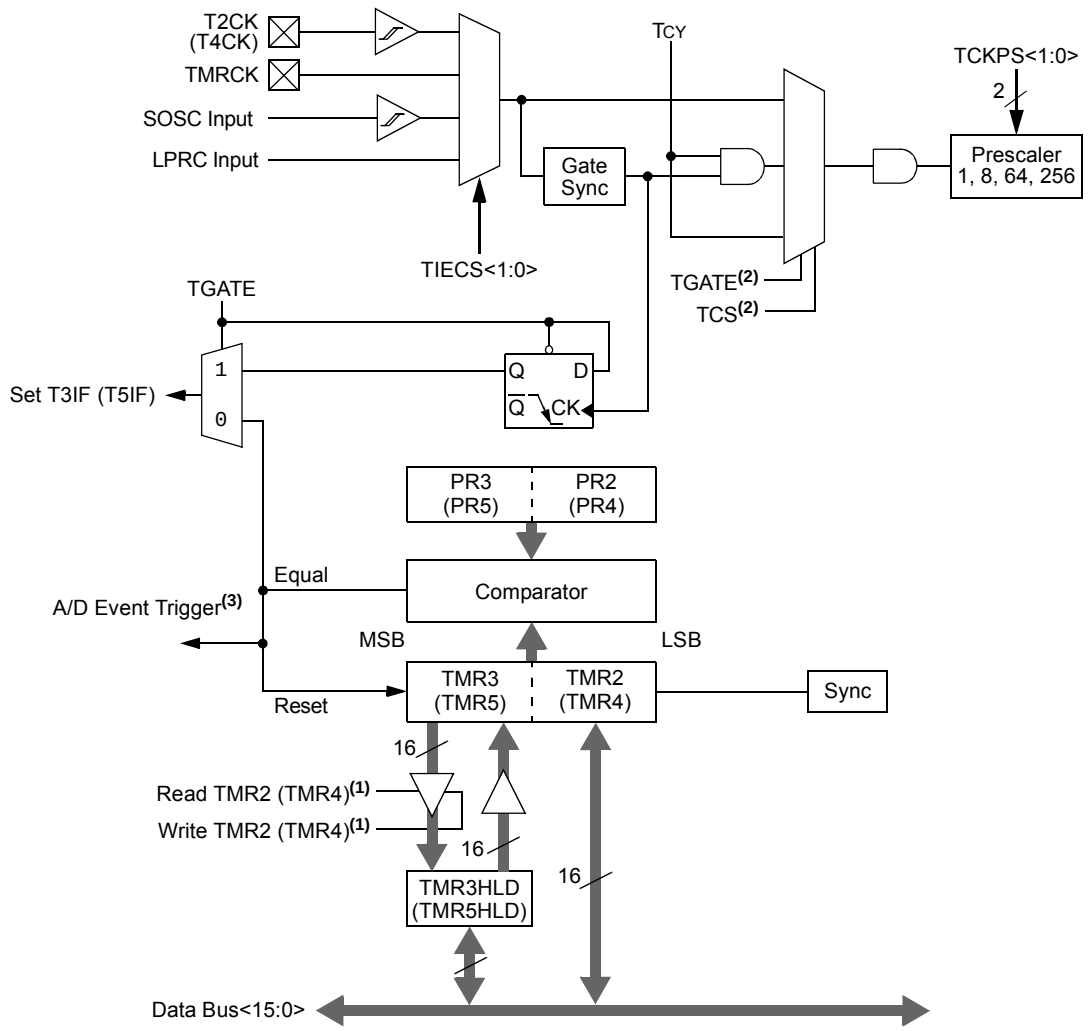
1. Set the T32 or T45 bit (T2CON<3> or T4CON<3> = 1).
2. Select the prescaler ratio for Timer2 or Timer4 using the TCKPS<1:0> bits.
3. Set the Clock and Gating modes using the TCS and TGATE bits. If TCS is set to an external clock, RPINRx (TxCK) must be configured to an available RPn/RPIn pin. For more information, see **Section 11.4 "Peripheral Pin Select (PPS)"**.
4. Load the timer period value. PR3 (or PR5) will contain the most significant word (msw) of the value, while PR2 (or PR4) contains the least significant word (lsw).
5. If interrupts are required, set the interrupt enable bit, T3IE or T5IE. Use the priority bits, T3IP<2:0> or T5IP<2:0>, to set the interrupt priority. Note that while Timer2 or Timer4 controls the timer, the interrupt appears as a Timer3 or Timer5 interrupt.
6. Set the TON bit (= 1).

The timer value, at any point, is stored in the register pair, TMR<3:2> (or TMR<5:4>). TMR3 (TMR5) always contains the most significant word of the count, while TMR2 (TMR4) contains the least significant word.

To configure any of the timers for individual 16-bit operation:

1. Clear the T32 bit corresponding to that timer (T2CON<3> for Timer2 and Timer3 or T4CON<3> (T45) for Timer4 and Timer5).
2. Select the timer prescaler ratio using the TCKPS<1:0> bits.
3. Set the Clock and Gating modes using the TCS and TGATE bits. See **Section 11.4 "Peripheral Pin Select (PPS)"** for more information.
4. Load the timer period value into the PRx register.
5. If interrupts are required, set the interrupt enable bit, TxIE. Use the priority bits, TxIP<2:0>, to set the interrupt priority.
6. Set the TON (TxCON<15> = 1) bit.

FIGURE 13-1: TIMER2/3 AND TIMER4/5 (32-BIT) BLOCK DIAGRAM

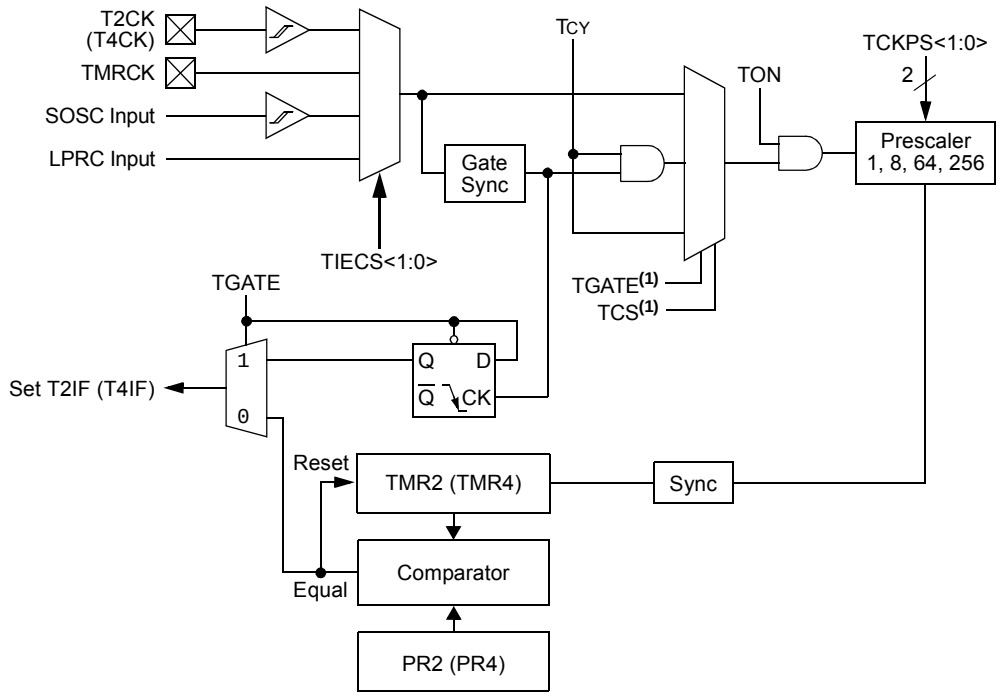


Note 1: The 32-Bit Timer Configuration bit, T32, must be set for 32-bit timer/counter operation. All control bits are respective to the T2CON and T4CON registers.

2: The timer clock input must be assigned to an available RPn/RPIn pin before use. See **Section 11.4 "Peripheral Pin Select (PPS)"** for more information.

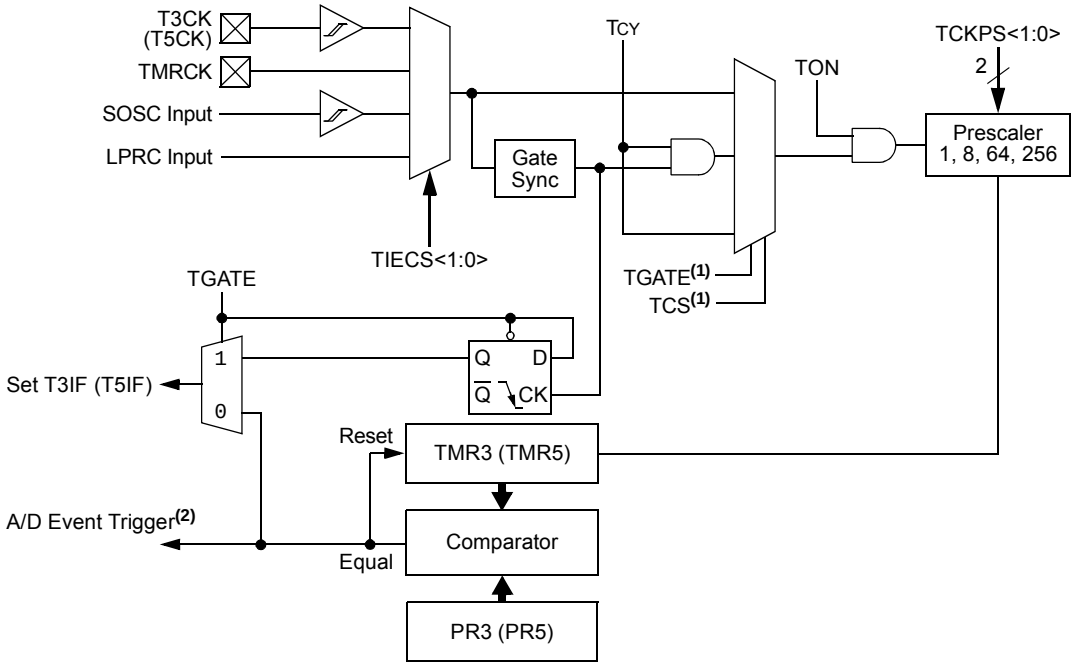
3: The A/D Event Trigger is available only on Timer2/3 in 32-bit mode and Timer3 in 16-bit mode.

FIGURE 13-2: TIMER2 AND TIMER4 (16-BIT SYNCHRONOUS) BLOCK DIAGRAM



Note 1: The timer clock input must be assigned to an available RPn/RPIn pin before use. See **Section 11.4 "Peripheral Pin Select (PPS)"** for more information.

FIGURE 13-3: TIMER3 AND TIMER5 (16-BIT ASYNCHRONOUS) BLOCK DIAGRAM



Note 1: The timer clock input must be assigned to an available RPn/RPIn pin before use. See **Section 11.4 "Peripheral Pin Select (PPS)"** for more information.

2: The A/D Event Trigger is available only on Timer3.

REGISTER 13-1: TxCON: TIMER2 AND TIMER4 CONTROL REGISTER(1)

R/W-0	U-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0
TON	—	TSIDL	—	—	—	TIECS1(2)	TIECS0(2)
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	U-0
—	TGATE	TCKPS1	TCKPS0	T32(3)	—	TCS(2)	—
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 **TON:** Timerx On bit
When TxCON<3> = 1:
1 = Starts 32-bit Timerx/y
0 = Stops 32-bit Timerx/y
When TxCON<3> = 0:
1 = Starts 16-bit Timerx
0 = Stops 16-bit Timerx

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

bit 13 **TSIDL:** Timerx Stop in Idle Mode bit
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode

bit 12-10 **Unimplemented:** Read as '0'

bit 9-8 **TIECS<1:0>:** Timerx Extended Clock Source Select bits (selected when TCS = 1)(2)
When TCS = 1:
11 = Generic Timer (TMRCK) External Input
10 = LPRC Oscillator
01 = TxCK External Clock Input
00 = SOSC
When TCS = 0:
These bits are ignored; timer is clocked from internal system clock (FOSC/2).

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

bit 6 **TGATE:** Timerx Gated Time Accumulation Enable bit
When TCS = 1:
This bit is ignored.
When TCS = 0:
1 = Gated time accumulation is enabled
0 = Gated time accumulation is disabled

bit 5-4 **TCKPS<1:0>:** Timerx Input Clock Prescale Select bits
11 = 1:256
10 = 1:64
01 = 1:8
00 = 1:1

Note 1: Changing the value of TxCON while the timer is running (TON = 1) causes the timer prescale counter to reset and is not recommended.

2: If TCS = 1 and TIECS<1:0> = x1, the selected external timer input (TMRCK or TxCK) must be configured to an available RPn/RPIn pin. For more information, see **Section 11.4 "Peripheral Pin Select (PPS)"**.

3: In T4CON, the T45 bit is implemented instead of T32 to select 32-bit mode. In 32-bit mode, the T3CON or T5CON control bits do not affect 32-bit timer operation.

REGISTER 13-1: TxCON: TIMER2 AND TIMER4 CONTROL REGISTER(1)

bit 3 **T32:** 32-Bit Timer Mode Select bit(3)
1 = Timerx and Timery form a single 32-bit timer
0 = Timerx and Timery act as two 16-bit timers
In 32-bit mode, T3CON control bits do not affect 32-bit timer operation.

bit 2 **Unimplemented:** Read as '0'

bit 1 **TCS:** Timerx Clock Source Select bit(2)
1 = Timer source is selected by TIECS<1:0>
0 = Internal clock (FOSC/2)

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

Note 1: Changing the value of TxCON while the timer is running (TON = 1) causes the timer prescale counter to reset and is not recommended.

2: If TCS = 1 and TIECS<1:0> = x1, the selected external timer input (TMRCK or TxCK) must be configured to an available RPn/RPIn pin. For more information, see **Section 11.4 "Peripheral Pin Select (PPS)"**.

3: In T4CON, the T45 bit is implemented instead of T32 to select 32-bit mode. In 32-bit mode, the T3CON or T5CON control bits do not affect 32-bit timer operation.

REGISTER 13-2: TyCON: TIMER3 AND TIMER5 CONTROL REGISTER(1)

R/W-0	U-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0
TON(2)	—	TSIDL(2)	—	—	—	TIECS1(2,3)	TIECS0(2,3)
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	U-0
—	TGATE(3)	TCKPS1(3)	TCKPS0(3)	—	—	TCS(2,3)	—
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 **TON:** Timery On bit(2)
1 = Starts 16-bit Timery
0 = Stops 16-bit Timery

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

bit 13 **TSIDL:** Timery Stop in Idle Mode bit(2)
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode

bit 12-10 **Unimplemented:** Read as '0'

bit 9-8 **TIECS<1:0>:** Timery Extended Clock Source Select bits (selected when TCS = 1)(2,3)
11 = Generic Timer (TMRCK) External Input
10 = LPRC Oscillator
01 = TxCK External Clock Input
00 = SOSC

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

bit 6 **TGATE:** Timery Gated Time Accumulation Enable bit(2)
When TCS = 1:
This bit is ignored.
When TCS = 0:
1 = Gated time accumulation is enabled
0 = Gated time accumulation is disabled

bit 5-4 **TCKPS<1:0>:** Timery Input Clock Prescale Select bits(2)
11 = 1:256
10 = 1:64
01 = 1:8
00 = 1:1

bit 3-2 **Unimplemented:** Read as '0'

bit 1 **TCS:** Timery Clock Source Select bit(2,3)
1 = External clock from pin, TyCK (on the rising edge)
0 = Internal clock (FOSC/2)

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

Note 1: Changing the value of TyCON while the timer is running (TON = 1) causes the timer prescale counter to reset and is not recommended.

2: When 32-bit operation is enabled (T2CON<3> or T4CON<3> = 1), these bits have no effect on Timery operation; all timer functions are set through T2CON and T4CON.

3: If TCS = 1 and TIECS<1:0> = x1, the selected external timer input (TMRCK or TyCK) must be configured to an available RPn/RPIn pin. For more information, see **Section 11.4 "Peripheral Pin Select (PPS)"**.

14.0 INPUT CAPTURE WITH DEDICATED TIMERS

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"*, **"Input Capture with Dedicated Timer"** (DS39722). The information in this data sheet supersedes the information in the FRM.

Devices in the PIC24FJ128GC010 family contain seven independent input capture modules. Each of the modules offers a wide range of configuration and operating options for capturing external pulse events and generating interrupts.

Key features of the input capture module include:

- Hardware-configurable for 32-bit operation in all modes by cascading two adjacent modules
- Synchronous and Trigger modes of output compare operation, with up to 30 user-selectable sync/trigger sources available
- A 4-level FIFO buffer for capturing and holding timer values for several events
- Configurable interrupt generation
- Up to 6 clock sources available for each module, driving a separate internal 16-bit counter

The module is controlled through two registers: ICxCON1 (Register 14-1) and ICxCON2 (Register 14-2). A general block diagram of the module is shown in Figure 14-1.

14.1 General Operating Modes

14.1.1 SYNCHRONOUS AND TRIGGER MODES

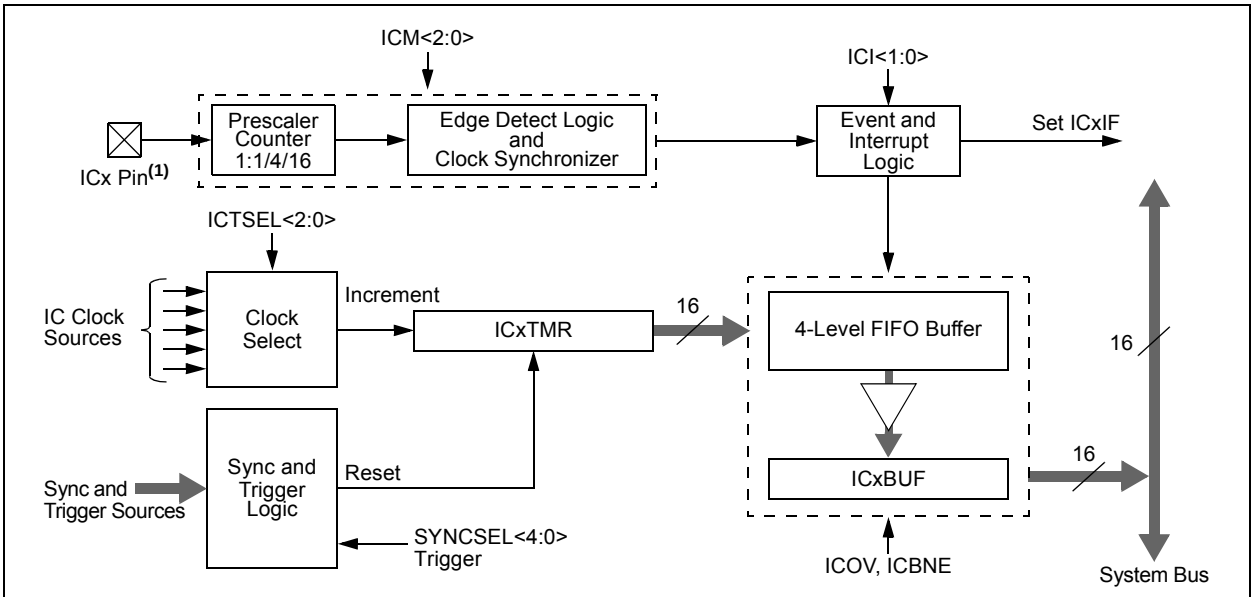
When the input capture module operates in a Free-Running mode, the internal 16-bit counter, ICx-TMR, counts up continuously, wrapping around from FFFFh to 0000h on each overflow. Its period is synchronized to the selected external clock source. When a capture event occurs, the current 16-bit value of the internal counter is written to the FIFO buffer.

In Synchronous mode, the module begins capturing events on the ICx pin as soon as its selected clock source is enabled. Whenever an event occurs on the selected sync source, the internal counter is reset. In Trigger mode, the module waits for a sync event from another internal module to occur before allowing the internal counter to run.

Standard, free-running operation is selected by setting the SYNCSELx bits (ICxCON2<4:0>) to '00000' and clearing the ICTRIG bit (ICxCON2<7>). Synchronous and Trigger modes are selected any time the SYNCSELx bits are set to any value except '00000'. The ICTRIG bit selects either Synchronous or Trigger mode; setting the bit selects Trigger mode operation. In both modes, the SYNCSEL<4:0> bits determine the sync/trigger source.

When the SYNCSELx bits are set to '00000' and ICTRIG is set, the module operates in Software Trigger mode. In this case, capture operations are started by manually setting the TRIGSTAT bit (ICxCON2<6>).

FIGURE 14-1: INPUT CAPTURE x BLOCK DIAGRAM



Note 1: The ICx inputs must be assigned to an available RPn/RPIn pin before use. See **Section 11.4 "Peripheral Pin Select (PPS)"** for more information.

14.1.2 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.

14.2 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<4:0> bits (ICxCON2<4:0>) to the desired sync/trigger source.
5. Set the ICTSEL<2:0> bits (ICxCON1<12:10>) for the desired clock source.
6. Set the ICI<1:0> bits (ICxCON1<6:5>) to the desired interrupt frequency
7. Select Synchronous or Trigger mode operation:
   a) Check that the SYNCSELx 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<2:0> 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 ICTSELx and SYNCSELx 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 ICTSELx and SYNCSELx bits 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 ICIx 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 ICMx 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).

REGISTER 14-1: ICxCON1: INPUT CAPTURE x CONTROL REGISTER 1

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0
—	—	ICSIDL	ICTSEL2	ICTSEL1	ICTSEL0	—	—
bit 15							bit 8

U-0	R/W-0	R/W-0	R-0, HSC	R-0, HSC	R/W-0	R/W-0	R/W-0
—	ICI1	ICI0	ICOV	ICBNE	ICM2(1)	ICM1(1)	ICM0(1)
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 15-14 **Unimplemented:** Read as '0'

bit 13 **ICSIDL:** Input Capture x Module Stop in Idle Control bit
1 = Input capture module Halts in CPU Idle mode
0 = Input capture module continues to operate in CPU Idle mode

bit 12-10 **ICTSEL<2:0>:** Input Capture x Timer Select bits
111 = System clock (FOSC/2)
110 = Reserved
101 = Reserved
100 = Timer1
011 = Timer5
010 = Timer4
001 = Timer2
000 = Timer3

bit 9-7 **Unimplemented:** Read as '0'

bit 6-5 **ICI<1:0>:** Select Number of Captures per Interrupt bits
11 = Interrupt on every fourth capture event
10 = Interrupt on every third capture event
01 = Interrupt on every second capture event
00 = Interrupt on every capture event

bit 4 **ICOV:** Input Capture x Overflow Status Flag bit (read-only)
1 = Input capture overflow has occurred
0 = No input capture overflow has occurred

bit 3 **ICBNE:** Input Capture x Buffer Empty Status bit (read-only)
1 = Input capture buffer is not empty, at least one more capture value can be read
0 = Input capture buffer is empty

bit 2-0 **ICM<2:0>:** Input Capture x Mode Select bits(1)
111 = Interrupt mode: Input capture functions as an interrupt pin only when the device is in Sleep or Idle mode (rising edge detect only, all other control bits are not applicable)
110 = Unused (module is disabled)
101 = Prescaler Capture mode: Capture on every 16th rising edge
100 = Prescaler Capture mode: Capture on every 4th rising edge
011 = Simple Capture mode: Capture on every rising edge
010 = Simple Capture mode: Capture on every falling edge
001 = Edge Detect Capture mode: Capture on every edge (rising and falling); ICI<1:0> bits do not control interrupt generation for this mode
000 = Input capture module is turned off

Note 1: The ICx input must also be configured to an available RPn/RPIn pin. For more information, see **Section 11.4 "Peripheral Pin Select (PPS)"**.

REGISTER 14-2: ICxCON2: INPUT CAPTURE x CONTROL REGISTER 2

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	IC32
bit 15							bit 8

R/W-0	R/W-0, HS	U-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-1
ICTRIG	TRIGSTAT	—	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0
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 15-9 **Unimplemented:** Read as '0'

bit 8 **IC32:** Cascade Two IC Modules Enable bit (32-bit operation)
1 = ICx and ICy operate in cascade as a 32-bit module (this bit must be set in both modules)
0 = ICx functions independently as a 16-bit module

bit 7 **ICTRIG:** ICx Sync/Trigger Select bit
1 = Triggers ICx from the source designated by the SYNCSELx bits
0 = Synchronizes ICx with the source designated by the SYNCSELx bits

bit 6 **TRIGSTAT:** Timer Trigger Status bit
1 = Timer source has been triggered and is running (set in hardware, can be set in software)
0 = Timer source has not been triggered and is being held clear

bit 5 **Unimplemented:** Read as '0'

bit 4-0 **SYNCSEL<4:0>:** Synchronization/Trigger Source Selection bits
11111 = Reserved
11101 = Reserved
11100 = CTMU(1)
11011 = Pipeline A/D(1)
11010 = Comparator 3(1)
11001 = Comparator 2(1)
11000 = Comparator 1(1)
10111 = Input Capture 8(2)
10110 = Input Capture 7(2)
10101 = Input Capture 6(2)
10100 = Input Capture 5(2)
10011 = Input Capture 4(2)
10010 = Input Capture 3(2)
10001 = Input Capture 2(2)
10000 = Input Capture 1(2)
01111 = Timer5
01110 = Timer4
01101 = Timer3
01100 = Timer2
01011 = Timer1
01010 = Input Capture 9(2)
01001 = Output Compare 9
•
•
•
00010 = Output Compare 2
00001 = Output Compare 1
00000 = Not synchronized to any other module

Note 1: Use these inputs as trigger sources only and never as sync sources.
2: Never use an IC module as its own trigger source by selecting this mode.

15.0 OUTPUT COMPARE WITH DEDICATED TIMERS

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"*, **"Output Compare with Dedicated Timer"** (DS39723). The information in this data sheet supersedes the information in the FRM.

Devices in the PIC24FJ128GC010 family all feature seven independent output compare modules. Each of these modules offers a wide range of configuration and operating options for generating pulse trains on internal device events, and can produce Pulse-Width Modulated (PWM) waveforms for driving power applications.

Key features of the output compare module include:

- Hardware-configurable for 32-bit operation in all modes by cascading two adjacent modules
- Synchronous and Trigger modes of output compare operation, with up to 31 user-selectable trigger/sync sources available
- Two separate Period registers (a main register, OCxR, and a secondary register, OCxRS) for greater flexibility in generating pulses of varying widths
- Configurable for single pulse or continuous pulse generation on an output event, or continuous PWM waveform generation
- Up to 6 clock sources available for each module, driving a separate internal 16-bit counter

15.1 General Operating Modes

15.1.1 SYNCHRONOUS AND TRIGGER MODES

When the output compare module operates in a Free-Running mode, the internal 16-bit counter, OCxTMR, runs counts up continuously, wrapping around from 0xFFFF to 0x0000 on each overflow. Its period is synchronized to the selected external clock source. Compare or PWM events are generated each time a match between the internal counter and one of the Period registers occurs.

In Synchronous mode, the module begins performing its compare or PWM operation as soon as its selected clock source is enabled. Whenever an event occurs on the selected sync source, the module's internal counter is reset. In Trigger mode, the module waits for a sync event from another internal module to occur before allowing the counter to run.

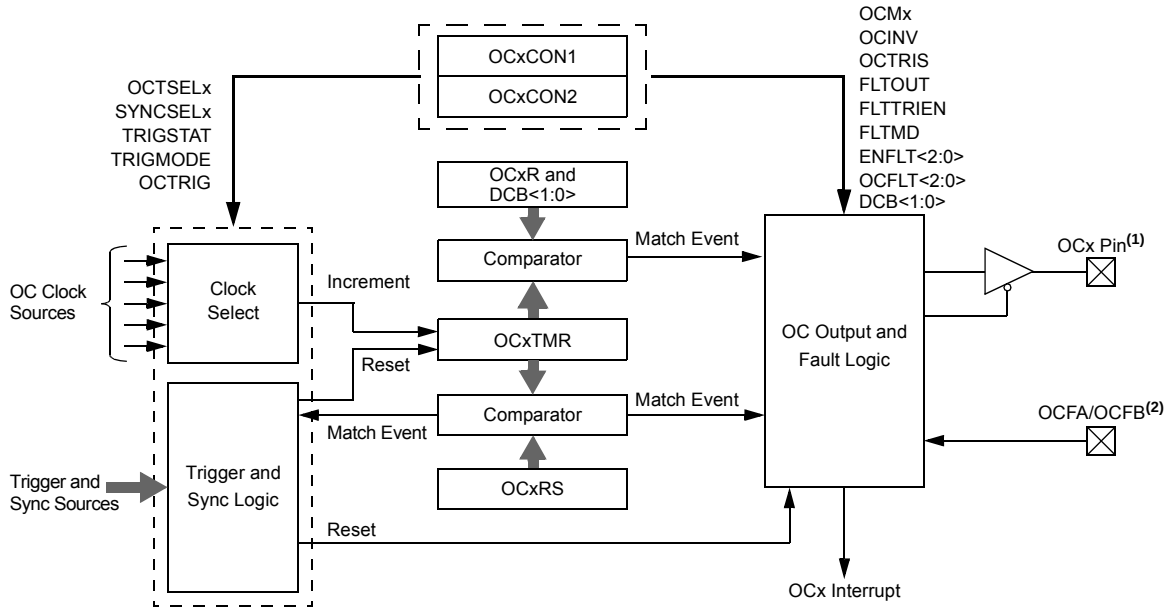
Free-Running mode is selected by default or any time that the SYNCSEL<4:0> bits (OCxCON2<4:0>) are set to '00000'. Synchronous or Trigger modes are selected any time the SYNCSELx bits are set to any value except '00000'. The OCTRIG bit (OCxCON2<7>) selects either Synchronous or Trigger mode; setting the bit selects Trigger mode operation. In both modes, the SYNCSELx bits determine the sync/trigger source.

15.1.2 CASCADED (32-BIT) MODE

By default, each module operates independently with its own set of 16-bit Timer and Duty Cycle registers. 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 (OCx) provides the Least Significant 16 bits of the 32-bit register pairs and the even numbered module (OCy) provides the Most Significant 16 bits. Wrap-arounds of the OCx registers cause an increment of their corresponding OCy registers.

Cascaded operation is configured in hardware by setting the OC32 bit (OCxCON2<8>) for both modules. For more details on cascading, refer to the "*PIC24F Family Reference Manual*", **Section 35. "Output Compare with Dedicated Timer"** (DS39723).

FIGURE 15-1: OUTPUT COMPARE x BLOCK DIAGRAM (16-BIT MODE)



Note 1: The OCx outputs must be assigned to an available RPn pin before use. See **Section 11.4 "Peripheral Pin Select (PPS)"** for more information.

2: The OCFA/OCFB Fault inputs must be assigned to an available RPn/RPIn pin before use. See **Section 11.4 "Peripheral Pin Select (PPS)"** for more information.

15.2 Compare Operations

In Compare mode (Figure 15-1), the output compare module can be configured for single-shot or continuous pulse generation. It can also repeatedly toggle an output pin on each timer event.

To set up the module for compare operations:

1. Configure the OCx output for one of the available Peripheral Pin Select pins.
2. Calculate the required values for the OCxR and (for Double Compare modes) OCxRS Duty Cycle registers:
   a) Determine the instruction clock cycle time. Take into account the frequency of the external clock to the timer source (if one is used) and the timer prescaler settings.
   b) Calculate time to the rising edge of the output pulse relative to the timer start value (0000h).
   c) Calculate the time to the falling edge of the pulse based on the desired pulse width and the time to the rising edge of the pulse.
3. Write the rising edge value to OCxR and the falling edge value to OCxRS.
4. Set the Timer Period register, PRy, to a value equal to or greater than the value in OCxRS.
5. Set the OCM<2:0> bits for the appropriate compare operation (= `0xx`).
6. For Trigger mode operations, set OCTRIG to enable Trigger mode. Set or clear TRIGMODE to configure trigger operation and TRIGSTAT to select a hardware or software trigger. For Synchronous mode, clear OCTRIG.
7. Set the SYNCSEL<4:0> bits to configure the trigger or synchronization source. If free-running timer operation is required, set the SYNCSELx bits to '`00000`' (no sync/trigger source).
8. Select the time base source with the OCTSEL<2:0> bits. If necessary, set the TON bit for the selected timer, which enables the compare time base to count. Synchronous mode operation starts as soon as the time base is enabled; Trigger mode operation starts after a trigger source event occurs.

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<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.
5. If Trigger mode operation is required, configure the trigger options in OCx by using the OCTRIG (OCxCON2<7>), TRIGMODE (OCxCON1<3>) and SYNCSELx (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.

15.3 Pulse-Width Modulation (PWM) Mode

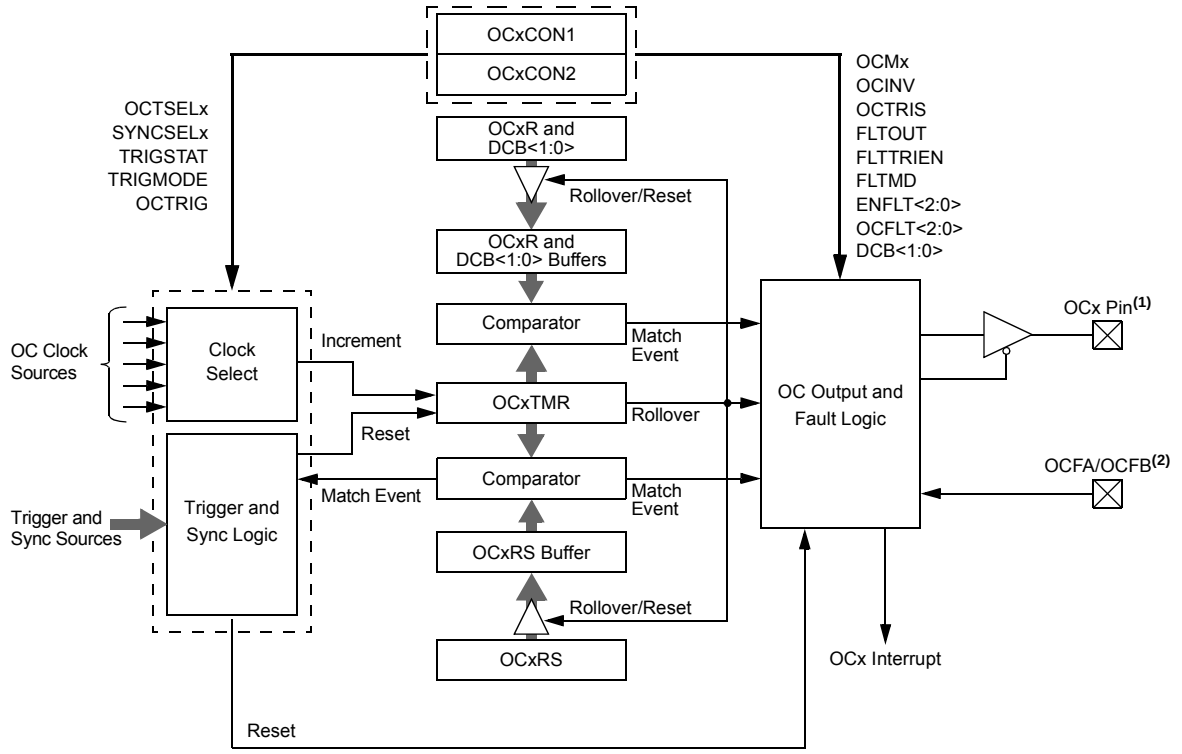
In PWM mode, the output compare module can be configured for edge-aligned or center-aligned pulse waveform generation. All PWM operations are double-buffered (buffer registers are internal to the module and are not mapped into SFR space).

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 (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.
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 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 11.4 "Peripheral Pin Select (PPS)"** for more information.

FIGURE 15-2: OUTPUT COMPARE x BLOCK DIAGRAM (DOUBLE-BUFFERED, 16-BIT PWM MODE)



Note 1: The OCx outputs must be assigned to an available RPn pin before use. See **Section 11.4 "Peripheral Pin Select (PPS)"** for more information.

2: The OCFA/OCFB Fault inputs must be assigned to an available RPn/RPIn pin before use. See **Section 11.4 "Peripheral Pin Select (PPS)"** for more information.

15.3.1 PWM PERIOD

The PWM period is specified by writing to PRy, the Timer Period register. The PWM period can be calculated using Equation 15-1.

EQUATION 15-1: CALCULATING THE PWM PERIOD(1)

$$\text{PWM Period} = [(\text{PRy}) + 1 \cdot T_{CY} \cdot (\text{Timer Prescale Value})$$

Where:

$$\text{PWM Frequency} = 1/[\text{PWM Period}]$$

Note 1: Based on $T_{CY} = T_{OSC} * 2$; Doze mode and PLL are disabled.

Note: A PRy value of N will produce a PWM period of N + 1 time base count cycles. For example, a value of 7, written into the PRy register, will yield a period consisting of 8 time base cycles.

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)

$$\text{Maximum PWM Resolution (bits)} = \frac{\log_{10}\left(\frac{FCY}{FPWM \bullet (\text{Timer Prescale Value})}\right)}{\log_{10}(2)} \text{ bits}$$

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

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

1. Find the Timer Period register value for a desired PWM frequency of 52.08 kHz, where FOSC = 8 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 ms

   PWM Period = (PR2 + 1) • TCY • (Timer2 Prescale Value)

   19.2 ms = PR2 + 1) • 62.5 ns • 1

   PR2 = 306

2. 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 = $\log_{10}(FCY/FPWM)/\log_{10}2)$ bits

   = $(\log_{10}(16\text{ MHz}/52.08\text{ kHz})/\log_{10}2)$ 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.

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

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	OCSIDL	OCTSEL2	OCTSEL1	OCTSEL0	ENFLT2(2)	ENFLT1(2)
bit 15							bit 8

R/W-0	R/W-0, HSC	R/W-0, HSC	R/W-0, HSC	R/W-0	R/W-0	R/W-0	R/W-0
ENFLT0(2)	OCFLT2(2,3)	OCFLT1(2,4)	OCFLT0(2,4)	TRIGMODE	OCM2(1)	OCM1(1)	OCM0(1)
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 15-14 **Unimplemented:** Read as '0'

bit 13 **OCSIDL:** Output Compare x Stop in Idle Mode Control bit
- 1 = Output Compare x Halts in CPU Idle mode
- 0 = Output Compare x continues to operate in CPU Idle mode

bit 12-10 **OCTSEL<2:0>:** Output Compare x Timer Select bits
- 111 = Peripheral clock (FCY)
- 110 = Reserved
- 101 = Reserved
- 100 = Timer1 clock (only the synchronous clock is supported)
- 011 = Timer5 clock
- 010 = Timer4 clock
- 001 = Timer3 clock
- 000 = Timer2 clock

bit 9 **ENFLT2:** Fault Input 2 Enable bit(2)
- 1 = Fault 2 (Comparator 1/2/3 out) is enabled(3)
- 0 = Fault 2 is disabled

bit 8 **ENFLT1:** Fault Input 1 Enable bit(2)
- 1 = Fault 1 (OCFB pin) is enabled(4)
- 0 = Fault 1 is disabled

bit 7 **ENFLT0:** Fault Input 0 Enable bit(2)
- 1 = Fault 0 (OCFA pin) is enabled(4)
- 0 = Fault 0 is disabled

bit 6 **OCFLT2:** PWM Fault 2 (Comparator 1/2/3) Condition Status bit(2,3)
- 1 = PWM Fault 2 has occurred
- 0 = No PWM Fault 2 has occurred

bit 5 **OCFLT1:** PWM Fault 1 (OCFB pin) Condition Status bit(2,4)
- 1 = PWM Fault 1 has occurred
- 0 = No PWM Fault 1 has occurred

Note 1: The OCx output must also be configured to an available RPn pin. For more information, see **Section 11.4 "Peripheral Pin Select (PPS)"**.

2: The Fault input enable and Fault status bits are valid when OCM<2:0> = 111 or 110.

3: The Comparator 1 output controls the OC1-OC3 channels; Comparator 2 output controls the OC4-OC6 channels; Comparator 3 output controls the OC7-OC9 channels.

4: The OCFA/OCFB Fault inputs must also be configured to an available RPn/RPIn pin. For more information, see **Section 11.4 "Peripheral Pin Select (PPS)"**.

REGISTER 15-1: OCxCON1: OUTPUT COMPARE x CONTROL REGISTER 1 (CONTINUED)

bit 4 **OCFLT0:** PWM Fault 0 (OCFA pin) Condition Status bit(2,4)
- 1 = PWM Fault 0 has occurred
- 0 = No PWM Fault 0 has occurred

bit 3 **TRIGMODE:** Trigger Status Mode Select bit
- 1 = TRIGSTAT (OCxCON2<6>) is cleared when OCxRS = OCxTMR or by software
- 0 = TRIGSTAT is only cleared by software

bit 2-0 **OCM<2:0>:** Output Compare x Mode Select bits(1)
- 111 = Center-Aligned PWM mode on OCx(2)
- 110 = Edge-Aligned PWM mode on OCx(2)
- 101 = Double Compare Continuous Pulse mode: Initializes the OCx pin low; toggles the OCx state continuously on alternate matches of OCxR and OCxRS
- 100 = Double Compare Single-Shot mode: Initializes the OCx pin low; toggles the OCx state on matches of OCxR and OCxRS for one cycle
- 011 = Single Compare Continuous Pulse mode: Compare events continuously toggle the OCx pin
- 010 = Single Compare Single-Shot mode: Initializes OCx pin high; compare event forces the OCx pin low
- 001 = Single Compare Single-Shot mode: Initializes OCx pin low; compare event forces the OCx pin high
- 000 = Output compare channel is disabled

Note 1: The OCx output must also be configured to an available RPn pin. For more information, see **Section 11.4 "Peripheral Pin Select (PPS)"**.

2: The Fault input enable and Fault status bits are valid when OCM<2:0> = 111 or 110.

3: The Comparator 1 output controls the OC1-OC3 channels; Comparator 2 output controls the OC4-OC6 channels; Comparator 3 output controls the OC7-OC9 channels.

4: The OCFA/OCFB Fault inputs must also be configured to an available RPn/RPIn pin. For more information, see **Section 11.4 "Peripheral Pin Select (PPS)"**.

REGISTER 15-2: OCxCON2: OUTPUT COMPARE x CONTROL REGISTER 2

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
FLTMD	FLTOUT	FLTTRIEN	OCINV	—	DCB1(3)	DCB0(3)	OC32
bit 15							bit 8

R/W-0	R/W-0, HS	R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0
OCTRIG	TRIGSTAT	OCTRIS	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0
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 **FLTMD:** Fault Mode Select bit
1 = Fault mode is maintained until the Fault source is removed and the corresponding OCFLT0 bit is cleared in software
0 = Fault mode is maintained until the Fault source is removed and a new PWM period starts

bit 14 **FLTOUT:** Fault Out bit
1 = PWM output is driven high on a Fault
0 = PWM output is driven low on a Fault

bit 13 **FLTTRIEN:** Fault Output State Select bit
1 = Pin is forced to an output on a Fault condition
0 = Pin I/O condition is unaffected by a Fault

bit 12 **OCINV:** OCx Invert bit
1 = OCx output is inverted
0 = OCx output is not inverted

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

bit 10-9 **DCB<1:0>:** PWM Duty Cycle Least Significant bits(3)
11 = Delays OCx falling edge by ¾ of the instruction cycle
10 = Delays OCx falling edge by ½ of the instruction cycle
01 = Delays OCx falling edge by ¼ of the instruction cycle
00 = OCx falling edge occurs at the start of the instruction cycle

bit 8 **OC32:** Cascade Two Output Compare Modules Enable bit (32-bit operation)
1 = Cascade module operation is enabled
0 = Cascade module operation is disabled

bit 7 **OCTRIG:** OCx Trigger/Sync Select bit
1 = Triggers OCx from the source designated by the SYNCSELx bits
0 = Synchronizes OCx with the source designated by the SYNCSELx bits

bit 6 **TRIGSTAT:** Timer Trigger Status bit
1 = Timer source has been triggered and is running
0 = Timer source has not been triggered and is being held clear

bit 5 **OCTRIS:** OCx Output Pin Direction Select bit
1 = OCx pin is tri-stated
0 = Output Compare Peripheral x is connected to an OCx pin

Note 1: Never use an OC module as its own trigger source, either by selecting this mode or another equivalent SYNCSELx setting.
2: Use these inputs as trigger sources only and never as sync sources.
3: The DCB<1:0> bits are double-buffered in the PWM modes only (OCM<2:0> (OCxCON1<2:0>) = 111, 110).

REGISTER 15-2: OCxCON2: OUTPUT COMPARE x CONTROL REGISTER 2 (CONTINUED)

bit 4-0 **SYNCSEL<4:0>:** Trigger/Synchronization Source Selection bits

11111 = This OC module(1)
11110 = OCTRIG1 External Input
11101 = OCTRIG2 External Input
11100 = CTMU(2)
11011 = Pipeline A/D(2)
11010 = Comparator 3(2)
11001 = Comparator 2(2)
11000 = Comparator 1(2)
10111 = Input Capture 8(2)
10110 = Input Capture 7(2)
10101 = Input Capture 6(2)
10100 = Input Capture 5(2)
10011 = Input Capture 4(2)
10010 = Input Capture 3(2)
10001 = Input Capture 2(2)
10000 = Input Capture 1(2)
01111 = Timer5
01110 = Timer4
01101 = Timer3
01100 = Timer2
01011 = Timer1
01010 = Input Capture 9(2)
01001 = Output Compare 9(1)
01000 = Output Compare 8(1)
00111 = Output Compare 7(1)
00110 = Output Compare 6(1)
00101 = Output Compare 5(1)
00100 = Output Compare 4(1)
00011 = Output Compare 3(1)
00010 = Output Compare 2(1)
00001 = Output Compare 1(1)
00000 = Not synchronized to any other module

Note 1: Never use an OC module as its own trigger source, either by selecting this mode or another equivalent SYNCSELx setting.

2: Use these inputs as trigger sources only and never as sync sources.

3: The DCB<1:0> bits are double-buffered in the PWM modes only (OCM<2:0> (OCxCON1<2:0>) = 111, 110).

NOTES:

16.0 SERIAL PERIPHERAL INTERFACE (SPI)

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"*, **"Serial Peripheral Interface (SPI)"** (DS39699). The information in this data sheet supersedes the information in the FRM.

The Serial Peripheral Interface (SPI) module is a synchronous serial interface useful for communicating with other peripheral or microcontroller devices. These peripheral devices may be serial EEPROMs, shift registers, display drivers, A/D Converters, etc. The SPI module is compatible with the SPI and SIOP Motorola® interfaces. All devices of the PIC24FJ128GC010 family include two SPI modules.

The module supports operation in two buffer modes. In Standard Buffer mode, data is shifted through a single serial buffer. In Enhanced Buffer mode, data is shifted through an 8-level FIFO buffer.

Note:	Do not perform Read/Modify/Write operations (such as bit-oriented instructions) on the SPIxBUF register in either Standard or Enhanced Buffer mode.

The module also supports a basic framed SPI protocol while operating in either Master or Slave mode. A total of four framed SPI configurations are supported.

The SPI serial interface consists of four pins:

- SDIx: Serial Data Input
- SDOx: Serial Data Output
- SCKx: Shift Clock Input or Output
- $\overline{SS}$x: Active-Low Slave Select or Frame Synchronization I/O Pulse

The SPI module can be configured to operate using 2, 3 or 4 pins. In the 3-pin mode, $\overline{SS}$x is not used; in the 2-pin mode, both SDOx and $\overline{SS}$x are not used.

Block diagrams of the module in Standard and Enhanced modes are shown in Figure 16-1 and Figure 16-2.

Note:	In this section, the SPI modules are referred to together as SPIx, or separately as SPI1 or SPI2. Special Function Registers will follow a similar notation. For example, SPIxCON1 and SPIxCON2 refer to the control registers for any of the two SPI modules.

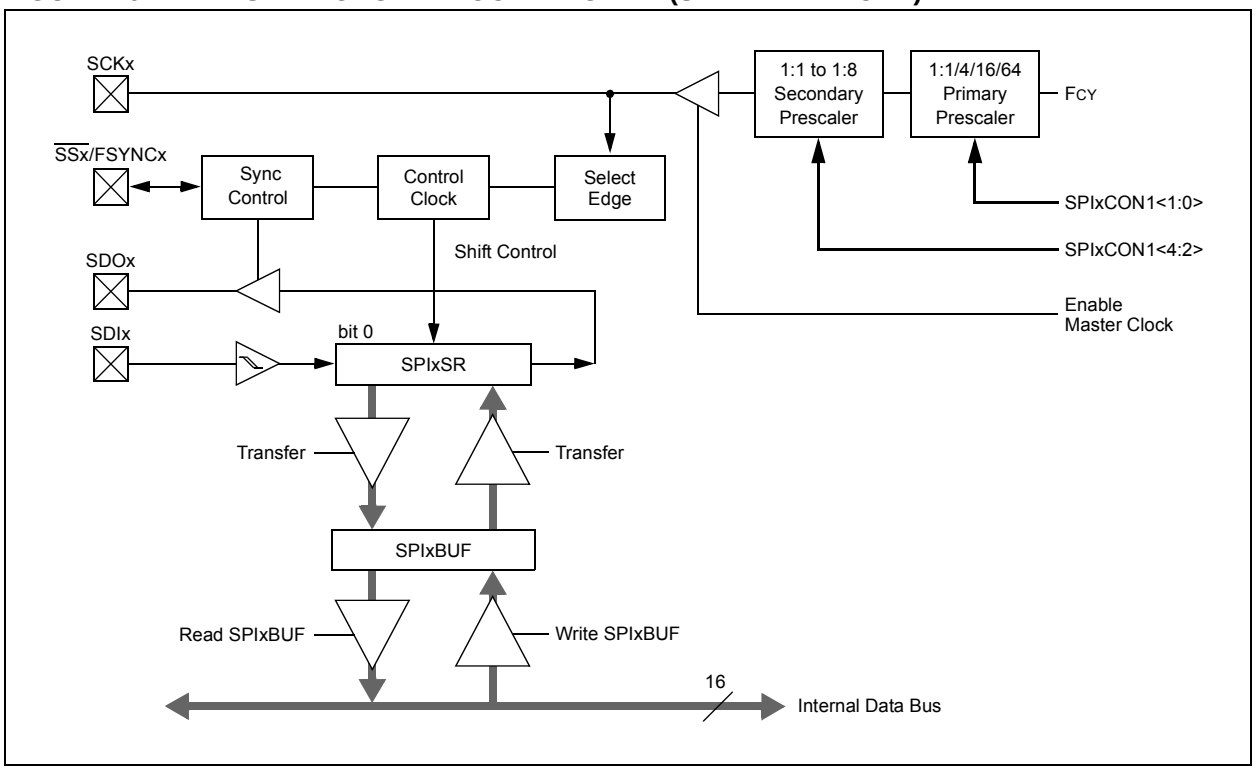
To set up the SPI module for the Standard Master mode of operation:

1. If using interrupts:
   a) Clear the SPIxIF bit in the respective IFSx register.
   b) Set the SPIxIE bit in the respective IECx register.
   c) Write the SPIxIP bits in the respective IPCx register to set the interrupt priority.
2. Write the desired settings to the SPIxCON1 and SPIxCON2 registers with MSTEN (SPIxCON1<5>) = 1.
3. Clear the SPIROV bit (SPIxSTAT<6>).
4. Enable SPI operation by setting the SPIEN bit (SPIxSTAT<15>).
5. 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 Standard Slave mode of operation:

1. Clear the SPIxBUF register.
2. If using interrupts:
   a) Clear the SPIxIF bit in the respective IFSx register.
   b) Set the SPIxIE bit in the respective IECx register.
   c) Write the SPIxIP bits in the respective IPCx 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 (SPIxCON1<8>) is set, then the SSEN bit (SPIxCON1<7>) must be set to enable the $\overline{SSx}$ pin.
6. Clear the SPIROV bit (SPIxSTAT<6>).
7. Enable SPI operation by setting the SPIEN bit (SPIxSTAT<15>).

FIGURE 16-1: SPIx MODULE BLOCK DIAGRAM (STANDARD MODE)



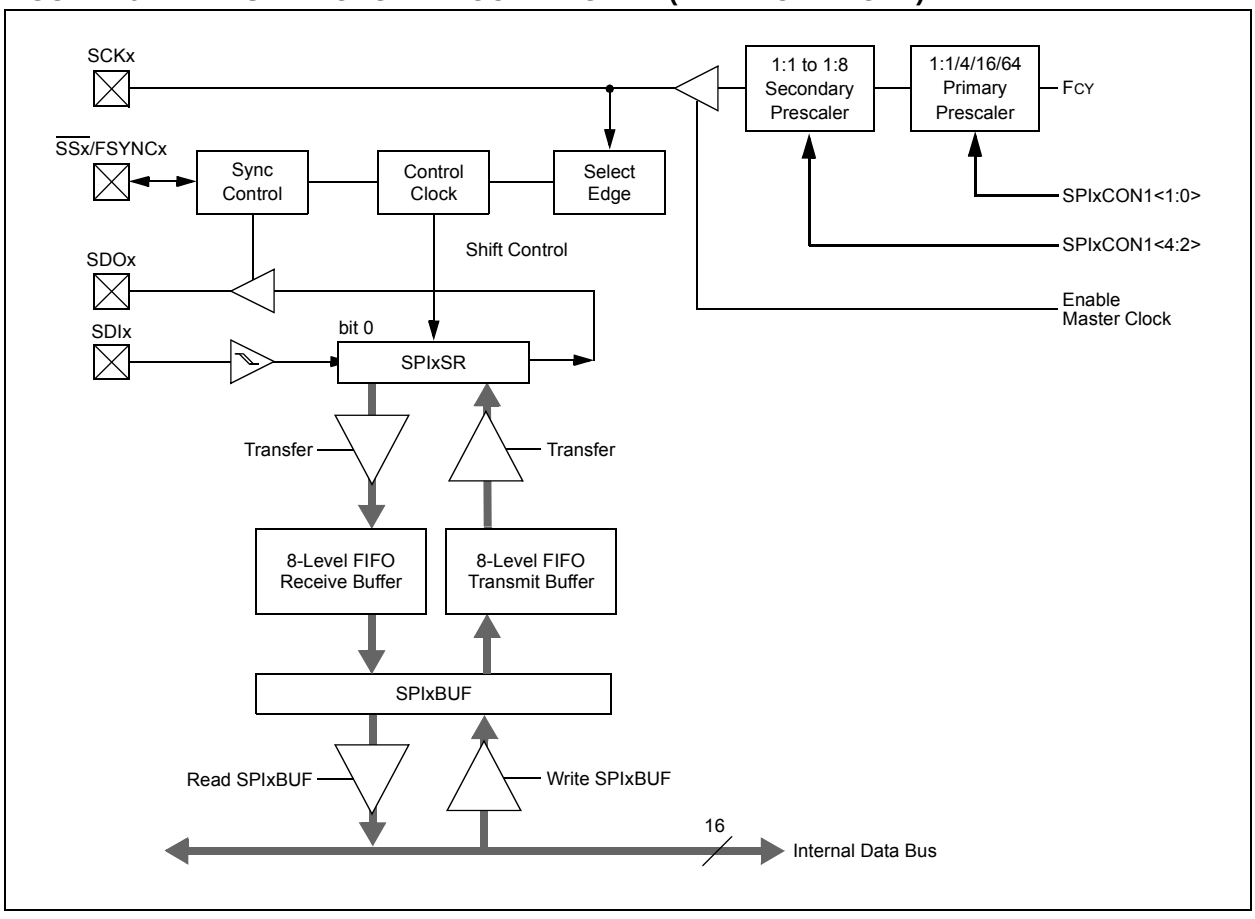
To set up the SPI module for the Enhanced Buffer Master mode of operation:

1. If using interrupts:
   a) Clear the SPIxIF bit in the respective IFSx register.
   b) Set the SPIxIE bit in the respective IECx register.
   c) Write the SPIxIP bits in the respective IPCx 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 IFSx register.
   b) Set the SPIxIE bit in the respective IECx register.
   c) Write the SPIxIP bits in the respective IPCx 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 16-2: SPIx MODULE BLOCK DIAGRAM (ENHANCED MODE)



REGISTER 16-1: SPIxSTAT: SPIx STATUS AND CONTROL REGISTER

R/W-0	U-0	R/W-0	U-0	U-0	R-0, HSC	R-0, HSC	R-0, HSC
SPIEN(1)	—	SPISIDL	—	—	SPIBEC2	SPIBEC1	SPIBEC0
bit 15							bit 8

R-0, HSC	R/C-0, HS	R-0, HSC	R/W-0	R/W-0	R/W-0	R-0, HSC	R-0, HSC
SRMPT	SPIROV	SRXMPT	SISEL2	SISEL1	SISEL0	SPITBF	SPIRBF
bit 7							bit 0

Legend:	C = Clearable bit	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
HSC = Hardware Settable/Clearable bit			

bit 15 **SPIEN:** SPIx Enable bit(1)
1 = Enables module and configures SCKx, SDOx, SDIx and $\overline{SSx}$ as serial port pins
0 = Disables module

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

bit 13 **SPISIDL:** SPIx Stop in Idle Mode bit
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode

bit 12-11 **Unimplemented:** Read as '0'

bit 10-8 **SPIBEC<2:0>:** SPIx Buffer Element Count bits (valid in Enhanced Buffer mode)
Master mode:
The number of SPI transfers pending.
Slave mode:
The number of SPI transfers unread.

bit 7 **SRMPT:** SPIx Shift Register (SPIxSR) Empty bit (valid in Enhanced Buffer mode)
1 = SPIx Shift register is empty and ready to send or receive
0 = SPIx Shift register is not empty

bit 6 **SPIROV:** SPIx Receive Overflow Flag bit
1 = A new byte/word is completely received and discarded; the user software has not read the previous data in the SPIxBUF register.
0 = No overflow has occurred

bit 5 **SRXMPT:** SPIx Receive FIFO Empty bit (valid in Enhanced Buffer mode)
1 = Receive FIFO is empty
0 = Receive FIFO is not empty

bit 4-2 **SISEL<2:0>:** SPIx Buffer Interrupt Mode bits (valid in Enhanced Buffer mode)
111 = Interrupt when the SPIx transmit buffer is full (SPITBF bit is set)
110 = Interrupt when the last bit is shifted into SPIxSR; as a result, the TX FIFO is empty
101 = Interrupt when the last bit is shifted out of SPIxSR; now the transmit is complete
100 = Interrupt when one data is shifted into the SPIxSR; as a result, the TX FIFO has one open spot
011 = Interrupt when the SPIx receive buffer is full (SPIRBF bit is set)
010 = Interrupt when the SPIx receive buffer is 3/4 or more full
001 = Interrupt when data is available in the receive buffer (SRMPT bit is set)
000 = Interrupt when the last data in the receive buffer is read; as a result, the buffer is empty (SRXMPT bit is set)

Note 1: If SPIEN = 1, these functions must be assigned to available RPn/RPIn pins before use. See **Section 11.4 "Peripheral Pin Select (PPS)"** for more information.

REGISTER 16-1: SPIxSTAT: SPIx STATUS AND CONTROL REGISTER (CONTINUED)

bit 1 **SPITBF:** SPIx Transmit Buffer Full Status bit

1 = Transmit has not yet started, SPIxTXB is full
0 = Transmit has started, SPIxTXB is empty

In Standard Buffer mode:
Automatically set in hardware when the CPU writes to the SPIxBUF location, loading the SPIxTXB. Automatically cleared in hardware when the SPIx module transfers data from SPIxTXB to SPIxSR.

In Enhanced Buffer mode:
Automatically set in hardware when the CPU writes to the SPIxBUF location, loading the last available buffer location. Automatically cleared in hardware when a buffer location is available for a CPU write.

bit 0 **SPIRBF:** SPIx Receive Buffer Full Status bit

1 = Receive is complete, SPIxRXB is full
0 = Receive is not complete, SPIxRXB is empty

In Standard Buffer mode:
Automatically set in hardware when SPIx transfers data from SPIxSR to SPIxRXB. Automatically cleared in hardware when the core reads the SPIxBUF location, reading SPIxRXB.

In Enhanced Buffer mode:
Automatically set in hardware when SPIx transfers data from the SPIxSR to the buffer, filling the last unread buffer location. Automatically cleared in hardware when a buffer location is available for a transfer from SPIxSR.

Note 1: If SPIEN = 1, these functions must be assigned to available RPn/RPIn pins before use. See **Section 11.4 "Peripheral Pin Select (PPS)"** for more information.

REGISTER 16-2: 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:** SPIx 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 11.4 "Peripheral Pin Select (PPS)"** for more information.

2: If DISSDO = 0, SDOx must be configured to an available RPn pin. See **Section 11.4 "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 11.4 "Peripheral Pin Select (PPS)"** for more information.

REGISTER 16-2: SPIxCON1: SPIx CONTROL REGISTER 1 (CONTINUED)

bit 4-2 **SPRE<2:0>:** Secondary Prescale bits (Master mode)
111 = Secondary prescale 1:1
110 = Secondary prescale 2:1
.
.
.
000 = Secondary prescale 8:1

bit 1-0 **PPRE<1:0>:** Primary Prescale bits (Master mode)
11 = Primary prescale 1:1
10 = Primary prescale 4:1
01 = Primary prescale 16:1
00 = Primary prescale 64:1

Note 1: If DISSCK = 0, SCKx must be configured to an available RPn pin. See **Section 11.4 "Peripheral Pin Select (PPS)"** for more information.

2: If DISSDO = 0, SDOx must be configured to an available RPn pin. See **Section 11.4 "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 11.4 "Peripheral Pin Select (PPS)"** for more information.

REGISTER 16-3: SPIxCON2: SPIx CONTROL REGISTER 2

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0
FRMEN	SPIFSD	SPIFPOL	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	SPIFE	SPIBEN
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 **FRMEN:** Framed SPIx Support bit
1 = Framed SPIx support is enabled
0 = Framed SPIx support is disabled

bit 14 **SPIFSD:** SPIx Frame Sync Pulse Direction Control on $\overline{SSx}$ Pin bit
1 = Frame Sync pulse input (slave)
0 = Frame Sync pulse output (master)

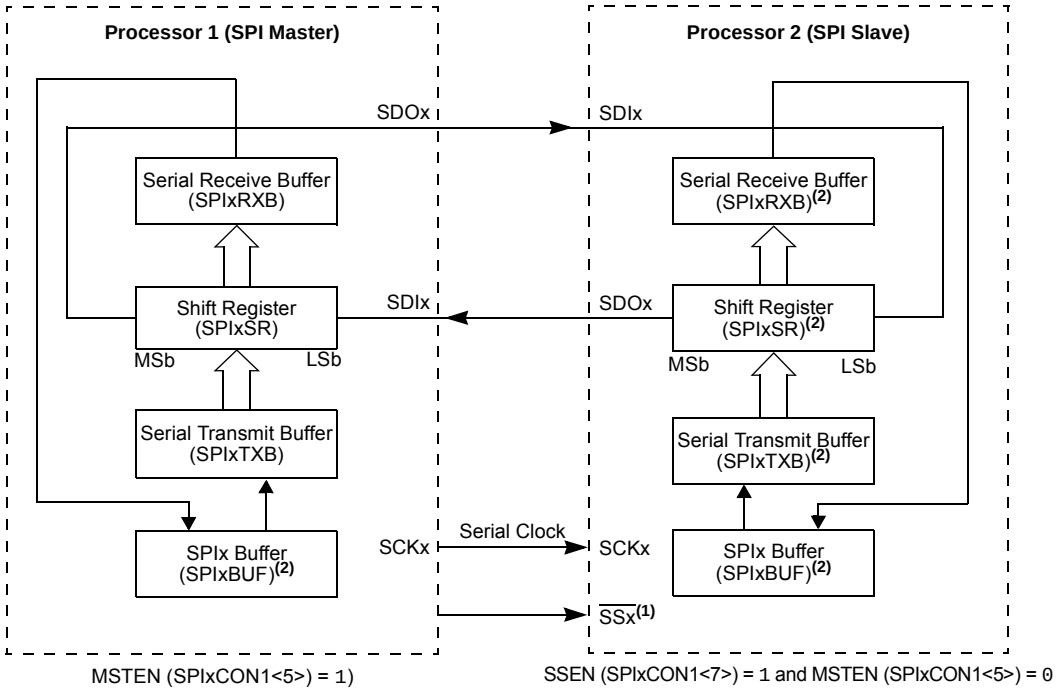
bit 13 **SPIFPOL:** SPIx Frame Sync Pulse Polarity bit (Frame mode only)
1 = Frame Sync pulse is active-high
0 = Frame Sync pulse is active-low

bit 12-2 **Unimplemented:** Read as '0'

bit 1 **SPIFE:** SPIx Frame Sync Pulse Edge Select bit
1 = Frame Sync pulse coincides with the first bit clock
0 = Frame Sync pulse precedes the first bit clock

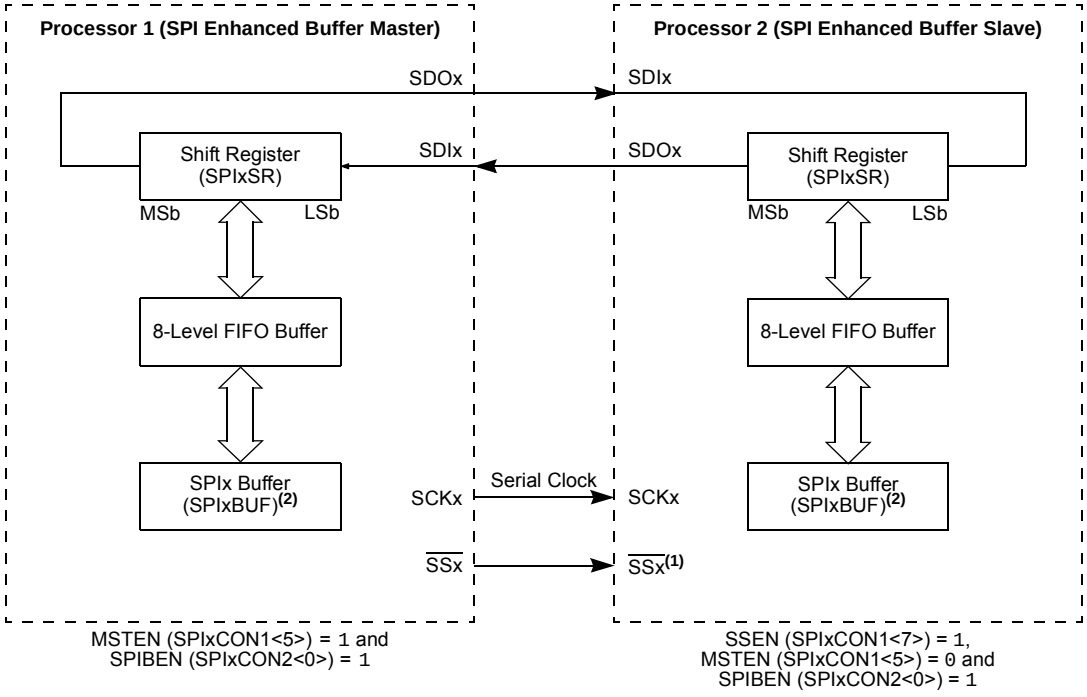
bit 0 **SPIBEN:** SPIx Enhanced Buffer Enable bit
1 = Enhanced buffer is enabled
0 = Enhanced buffer is disabled (Legacy mode)

FIGURE 16-3: SPIx MASTER/SLAVE CONNECTION (STANDARD MODE)



Note 1: Using the $\overline{SSx}$ pin in Slave mode of operation is optional.
2: User must write transmit data to read the received data from SPIxBUF. The SPIxTXB and SPIxRXB registers are memory mapped to SPIxBUF.

FIGURE 16-4: SPIx MASTER/SLAVE CONNECTION (ENHANCED BUFFER MODES)



Note 1: Using the $\overline{SSx}$ pin in Slave mode of operation is optional.
2: User must write transmit data to read the received data from SPIxBUF. The SPIxTXB and SPIxRXB registers are memory mapped to SPIxBUF.

FIGURE 16-5: SPIx MASTER, FRAME MASTER CONNECTION DIAGRAM

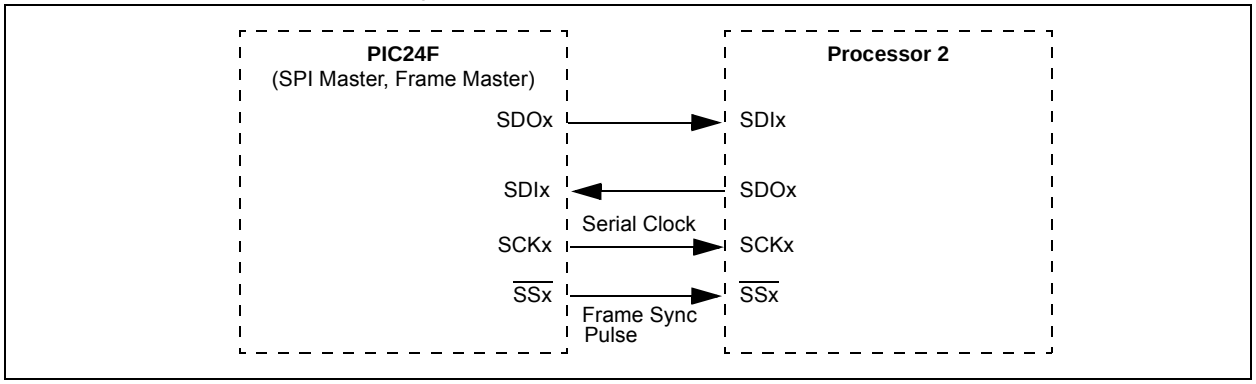


FIGURE 16-6: SPIx MASTER, FRAME SLAVE CONNECTION DIAGRAM

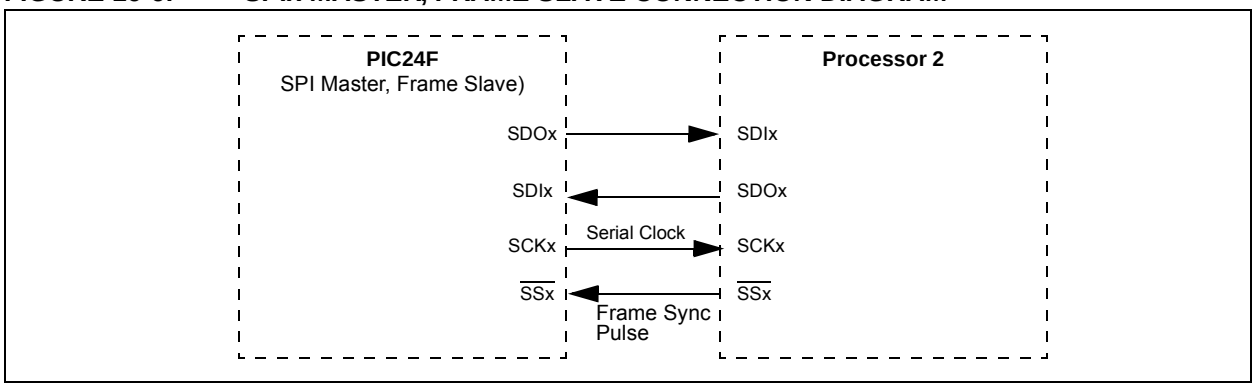


FIGURE 16-7: SPIx SLAVE, FRAME MASTER CONNECTION DIAGRAM

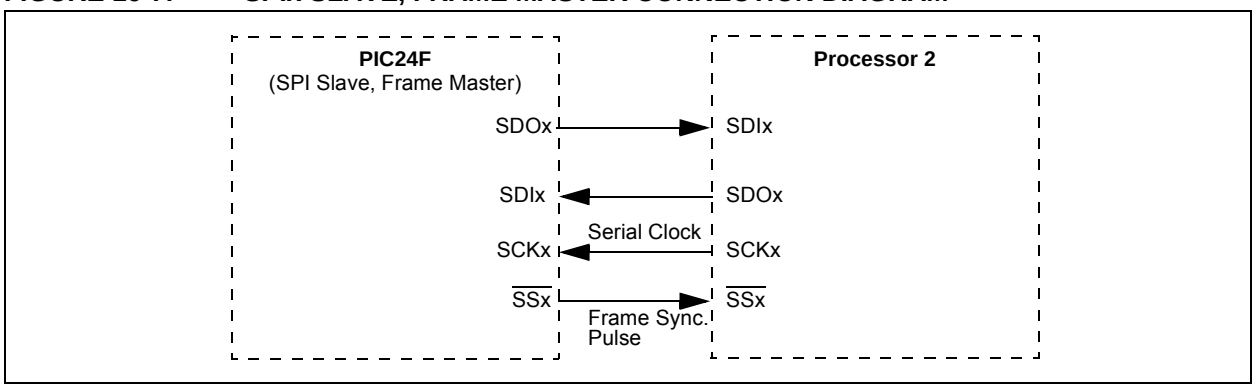
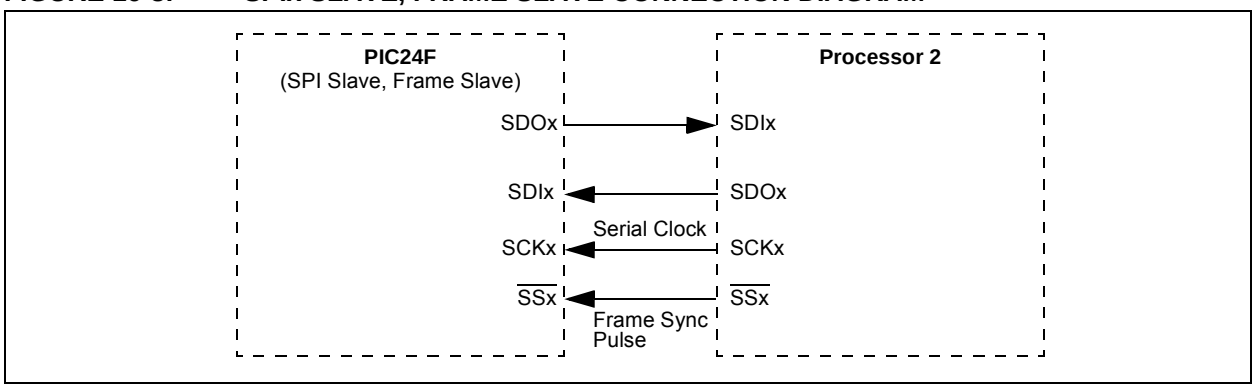


FIGURE 16-8: SPIx SLAVE, FRAME SLAVE CONNECTION DIAGRAM



EQUATION 16-1: RELATIONSHIP BETWEEN DEVICE AND SPIx CLOCK SPEED(1)

$$FSCK = \frac{FCY}{\text{Primary Prescaler x Secondary Prescaler}}$$

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

TABLE 16-1: SAMPLE SCKx FREQUENCIES(1,2)

FCY = 16 MHz		Secondary Prescaler Settings				
		1:1	**2:1**	**4:1**	**6:1**	**8:1**
Primary Prescaler Settings	1:1	Invalid	8000	4000	2667	2000
	4:1	4000	2000	1000	667	500
	16:1	1000	500	250	167	125
	64:1	250	125	63	42	31
FCY = 5 MHz						
Primary Prescaler Settings	1:1	5000	2500	1250	833	625
	4:1	1250	625	313	208	156
	16:1	313	156	78	52	39
	64:1	78	39	20	13	10

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

2: SCKx frequencies are shown in kHz.

NOTES:

17.0 INTER-INTEGRATED CIRCUIT™ (I^2C™)

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"*, **"Inter-Integrated Circuit™ (I^2C™)"** (DS39702). The information in this data sheet supersedes the information in the FRM.

The Inter-Integrated Circuit™ (I^2C™) module is a serial interface useful for communicating with other peripheral or microcontroller devices. These peripheral devices may be serial EEPROMs, display drivers, A/D Converters, etc.

The I^2C module supports these features:

- Independent master and slave logic
- 7-bit and 10-bit device addresses
- General call address as defined in the I^2C protocol
- Clock stretching to provide delays for the processor to respond to a slave data request
- Both 100 kHz and 400 kHz bus specifications
- Configurable address masking
- Multi-Master modes to prevent loss of messages in arbitration
- Bus Repeater mode, allowing the acceptance of all messages as a slave regardless of the address
- Automatic SCL

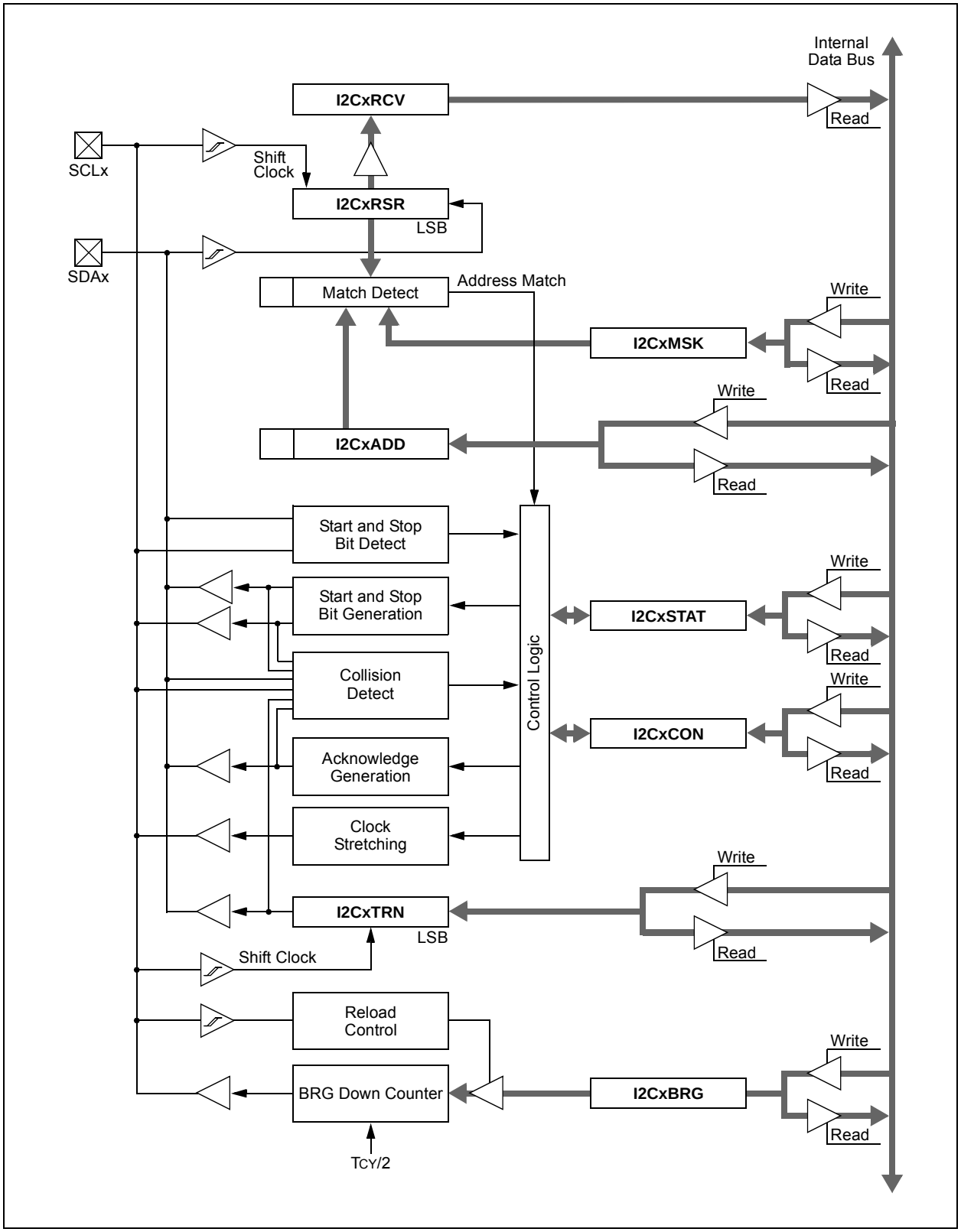
A block diagram of the module is shown in Figure 17-1.

17.1 Communicating as a Master in a Single Master Environment

The details of sending a message in Master mode depends on the communications protocol for the device being communicated with. Typically, the sequence of events is as follows:

1. Assert a Start condition on SDAx and SCLx.
2. Send the I^2C device address byte to the slave with a write indication.
3. Wait for and verify an Acknowledge from the slave.
4. Send the first data byte (sometimes known as the command) to the slave.
5. Wait for and verify an Acknowledge from the slave.
6. Send the serial memory address low byte to the slave.
7. Repeat Steps 4 and 5 until all data bytes are sent.
8. Assert a Repeated Start condition on SDAx and SCLx.
9. Send the device address byte to the slave with a read indication.
10. Wait for and verify an Acknowledge from the slave.
11. Enable master reception to receive serial memory data.
12. Generate an ACK or NACK condition at the end of a received byte of data.
13. Generate a Stop condition on SDAx and SCLx.

FIGURE 17-1: I²C™ BLOCK DIAGRAM



17.2 Setting Baud Rate When Operating as a Bus Master

To compute the Baud Rate Generator reload value, use Equation 17-1.

EQUATION 17-1: COMPUTING BAUD RATE RELOAD VALUE(1,2)

$$FSCL = \frac{FCY}{I2CxBRG + 1 + \frac{FCY}{10{,}000{,}000}}$$

or:

$$I2CxBRG = \left(\frac{FCY}{FSCL} - \frac{FCY}{10{,}000{,}000} - 1\right)$$

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

2: These clock rate values are for guidance only. The actual clock rate can be affected by various system-level parameters. The actual clock rate should be measured in its intended application.

17.3 Slave Address Masking

The I2CxMSK register (Register 17-3) designates address bit positions as “don’t care” for both 7-Bit and 10-Bit Addressing modes. Setting a particular bit location (= 1) in the I2CxMSK register causes the slave module to respond, whether the corresponding address bit value is a ‘0’ or a ‘1’. For example, when I2CxMSK is set to ‘00100000’, the slave module will detect both addresses, ‘0000000’ and ‘0100000’.

To enable address masking, the Intelligent Peripheral Management Interface (IPMI) must be disabled by clearing the IPMIEN bit (I2CxCON<11>).

Note: As a result of changes in the I²C™ protocol, the addresses in Table 17-2 are reserved and will not be Acknowledged in Slave mode. This includes any address mask settings that include any of these addresses.

TABLE 17-1: I²C™ CLOCK RATES(1,2)

Required System FSCL	FCY	I2CxBRG Value		Actual FSCL
		(Decimal)	(Hexadecimal)	
100 kHz	16 MHz	157	9D	100 kHz
100 kHz	8 MHz	78	4E	100 kHz
100 kHz	4 MHz	39	27	99 kHz
400 kHz	16 MHz	37	25	404 kHz
400 kHz	8 MHz	18	12	404 kHz
400 kHz	4 MHz	9	9	385 kHz
400 kHz	2 MHz	4	4	385 kHz
1 MHz	16 MHz	13	D	1.026 MHz
1 MHz	8 MHz	6	6	1.026 MHz
1 MHz	4 MHz	3	3	0.909 MHz

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

2: These clock rate values are for guidance only. The actual clock rate can be affected by various system-level parameters. The actual clock rate should be measured in its intended application.

TABLE 17-2: I²C™ RESERVED ADDRESSES(1)

Slave Address	R/$\overline{W}$ Bit	Description
0000 000	0	General Call Address(2)
0000 000	1	Start Byte
0000 001	x	CBus Address
0000 01x	x	Reserved
0000 1xx	x	HS Mode Master Code
1111 0xx	x	10-Bit Slave Upper Byte(3)
1111 1xx	x	Reserved

Note 1: The address bits listed here will never cause an address match, independent of address mask settings.

2: The address will be Acknowledged only if GCEN = 1.

3: A match on this address can only occur on the upper byte in 10-Bit Addressing mode.

REGISTER 17-1: I2CxCON: I2Cx CONTROL REGISTER

R/W-0	U-0	R/W-0	R/W-1, HC	R/W-0	R/W-0	R/W-0	R/W-0
I2CEN	—	I2CSIDL	SCLREL	IPMIEN	A10M	DISSLW	SMEN
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC
GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN
bit 7							bit 0

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

bit 15 **I2CEN:** I2Cx Enable bit
1 = Enables the I2Cx module and configures the SDAx and SCLx pins as serial port pins
0 = Disables the I2Cx module; all I^2C™ pins are controlled by port functions

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

bit 13 **I2CSIDL:** I2Cx Stop in Idle Mode bit
1 = Discontinues module operation when device enters an Idle mode
0 = Continues module operation in Idle mode

bit 12 **SCLREL:** SCLx Release Control bit (when operating as I^2C slave)
1 = Releases SCLx clock
0 = Holds SCLx clock low (clock stretch)
If STREN = 1:
Bit is R/$\overline{W}$ (i.e., software may write '0' to initiate stretch and write '1' to release clock). Hardware is clear at the beginning of slave transmission. Hardware is clear at the end of slave reception.
If STREN = 0:
Bit is R/S (i.e., software may only write '1' to release clock). Hardware is clear at the beginning of slave transmission.

bit 11 **IPMIEN:** Intelligent Platform Management Interface (IPMI) Enable bit
1 = IPMI Support mode is enabled; all addresses are Acknowledged
0 = IPMI Support mode is disabled

bit 10 **A10M:** 10-Bit Slave Addressing bit
1 = I2CxADD is a 10-bit slave address
0 = I2CxADD is a 7-bit slave address

bit 9 **DISSLW:** Disable Slew Rate Control bit
1 = Slew rate control is disabled
0 = Slew rate control is enabled

bit 8 **SMEN:** SMBus Input Levels bit
1 = Enables I/O pin thresholds compliant with SMBus specifications
0 = Disables the SMBus input thresholds

bit 7 **GCEN:** General Call Enable bit (when operating as I^2C slave)
1 = Enables interrupt when a general call address is received in the I2CxRSR (module is enabled for reception)
0 = General call address is disabled

bit 6 **STREN:** SCLx Clock Stretch Enable bit (when operating as I^2C slave)
Used in conjunction with the SCLREL bit.
1 = Enables software or receive clock stretching
0 = Disables software or receive clock stretching

REGISTER 17-1: I2CxCON: I2Cx CONTROL REGISTER (CONTINUED)

bit 5 **ACKDT:** Acknowledge Data bit (when operating as I^2C master; applicable during master receive)
Value that will be transmitted when the software initiates an Acknowledge sequence.
1 = Sends NACK during Acknowledge
0 = Sends ACK during Acknowledge

bit 4 **ACKEN:** Acknowledge Sequence Enable bit (when operating as I^2C master; applicable during master receive)
1 = Initiates Acknowledge sequence on SDAx and SCLx pins and transmits the ACKDT data bit. Hardware is clear at the end of the master Acknowledge sequence.
0 = Acknowledge sequence is not in progress

bit 3 **RCEN:** Receive Enable bit (when operating as I^2C master)
1 = Enables Receive mode for I^2C. Hardware is clear at the end of the eighth bit of the master receive data byte.
0 = Receive sequence is not in progress

bit 2 **PEN:** Stop Condition Enable bit (when operating as I^2C master)
1 = Initiates Stop condition on the SDAx and SCLx pins. Hardware is clear at the end of the master Stop sequence.
0 = Stop condition is not in progress

bit 1 **RSEN:** Repeated Start Condition Enable bit (when operating as I^2C master)
1 = Initiates Repeated Start condition on the SDAx and SCLx pins. Hardware is clear at the end of the master Repeated Start sequence.
0 = Repeated Start condition is not in progress

bit 0 **SEN:** Start Condition Enable bit (when operating as I^2C master)
1 = Initiates Start condition on SDAx and SCLx pins. Hardware is clear at the end of the master Start sequence.
0 = Start condition is not in progress

REGISTER 17-2: I2CxSTAT: I2Cx STATUS REGISTER

R-0, HSC	R-0, HSC	U-0	U-0	U-0	R/C-0, HS	R-0, HSC	R-0, HSC
ACKSTAT	TRSTAT	—	—	—	BCL	GCSTAT	ADD10
bit 15							bit 8

R/C-0, HS	R/C-0, HS	R-0, HSC	R/C-0, HSC	R/C-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC
IWCOL	I2COV	DAC	P	S	R/$\overline{W}$	RBF	TBF
bit 7							bit 0

Legend:	C = Clearable bit	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
HSC = Hardware Settable/Clearable bit			

bit 15 **ACKSTAT:** Acknowledge Status bit
1 = NACK was detected last
0 = ACK was detected last
Hardware is set or cleared at the end of Acknowledge.

bit 14 **TRSTAT:** Transmit Status bit (when operating as I^2C™ master; applicable to master transmit operation)
1 = Master transmit is in progress (8 bits + ACK)
0 = Master transmit is not in progress
Hardware is set at the beginning of master transmission; hardware is clear at the end of slave Acknowledge.

bit 13-11 **Unimplemented:** Read as '0'

bit 10 **BCL:** Master Bus Collision Detect bit
1 = A bus collision has been detected during a master operation
0 = No bus collision
Hardware is set at the detection of a bus collision.

bit 9 **GCSTAT:** General Call Status bit
1 = General call address was received
0 = General call address was not received
Hardware is set when the address matches the general call address; hardware is clear at Stop detection.

bit 8 **ADD10:** 10-Bit Address Status bit
1 = 10-bit address was matched
0 = 10-bit address was not matched
Hardware is set at the match of the 2nd byte of the matched 10-bit address; hardware is clear at Stop detection.

bit 7 **IWCOL:** Write Collision Detect bit
1 = An attempt to write to the I2CxTRN register failed because the I^2C module is busy
0 = No collision
Hardware is set at an occurrence of a write to I2CxTRN while busy (cleared by software).

bit 6 **I2COV:** I2Cx Receive Overflow Flag bit
1 = A byte was received while the I2CxRCV register is still holding the previous byte
0 = No overflow
Hardware is set at an attempt to transfer I2CxRSR to I2CxRCV (cleared by software).

bit 5 **DAC:** Data/$\overline{\text{Address}}$ bit (when operating as I^2C slave)
1 = Indicates that the last byte received was data
0 = Indicates that the last byte received was a device address
Hardware is clear at the device address match. Hardware is set after a transmission finishes or by reception of a slave byte.

REGISTER 17-2: I2CxSTAT: I2Cx STATUS REGISTER (CONTINUED)

bit 4 **P:** Stop bit
1 = Indicates that a Stop bit has been detected last
0 = Stop bit was not detected last
Hardware is set or clear when Start, Repeated Start or Stop is detected.

bit 3 **S:** Start bit
1 = Indicates that a Start (or Repeated Start) bit has been detected last
0 = Start bit was not detected last
Hardware is set or clear when Start, Repeated Start or Stop is detected.

bit 2 **R/$\overline{W}$:** Read/$\overline{Write}$ Information bit (when operating as I^2C slave)
1 = Read: Indicates the data transfer is output from the slave
0 = Write: Indicates the data transfer is input to the slave
Hardware is set or clear after the reception of an I^2C device address byte.

bit 1 **RBF:** Receive Buffer Full Status bit
1 = Receive is complete, I2CxRCV is full
0 = Receive is not complete, I2CxRCV is empty
Hardware is set when I2CxRCV is written with the received byte; hardware is clear when the software reads I2CxRCV.

bit 0 **TBF:** Transmit Buffer Full Status bit
1 = Transmit is in progress, I2CxTRN is full
0 = Transmit is complete, I2CxTRN is empty
Hardware is set when software writes to I2CxTRN; hardware is clear at the completion of data transmission.

REGISTER 17-3: I2CxMSK: I2Cx SLAVE MODE ADDRESS MASK REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	AMSK9	AMSK8
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
AMSK7	AMSK6	AMSK5	AMSK4	AMSK3	AMSK2	AMSK1	AMSK0
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 15-10 **Unimplemented:** Read as ‘0’

bit 9-0 **AMSK<9:0>:** Mask for Address Bit x Select bits
1 = Enables masking for bit x of the incoming message address; bit match is not required in this position
0 = Disables masking for bit x; bit match is required in this position

NOTES:

18.0 UNIVERSAL ASYNCHRONOUS RECEIVER TRANSMITTER (UART)

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"*, **"UART"** (DS39708). The information in this data sheet supersedes the information in the FRM.

The Universal Asynchronous Receiver Transmitter (UART) module is one of the serial I/O modules available in the PIC24F device family. The UART is a full-duplex, asynchronous system that can communicate with peripheral devices, such as personal computers, LIN/J2602, RS-232 and RS-485 interfaces. The module also supports a hardware flow control option with the $\overline{\text{UxCTS}}$ and $\overline{\text{UxRTS}}$ pins, and includes an IrDA® encoder and decoder.

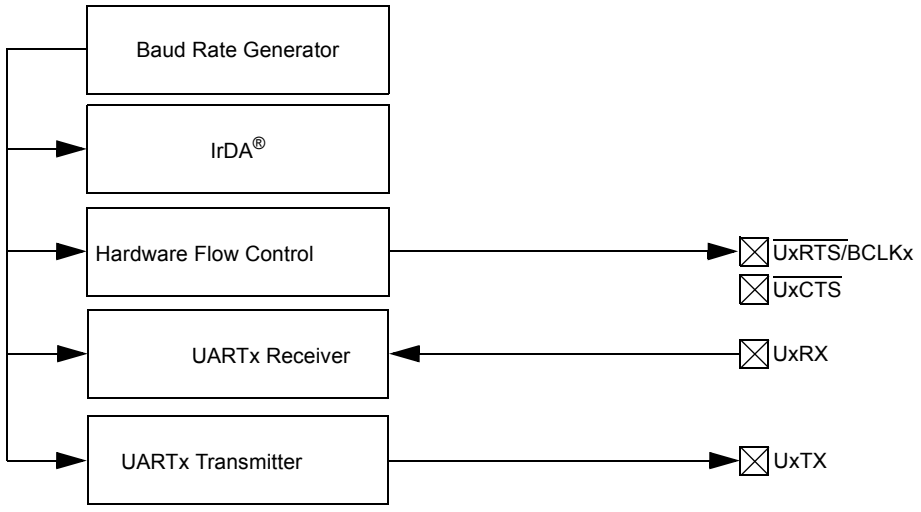
The primary features of the UARTx module are:

- Full-Duplex, 8 or 9-Bit Data Transmission Through the UxTX and UxRX Pins
- Even, Odd or No Parity Options (for 8-bit data)
- One or Two Stop bits
- Hardware Flow Control Option with the $\overline{\text{UxCTS}}$ and $\overline{\text{UxRTS}}$ Pins
- Fully Integrated Baud Rate Generator with 16-Bit Prescaler
- Baud Rates Ranging from 15 bps to 1 Mbps at 16 MIPS
- 4-Deep, First-In-First-Out (FIFO) Transmit Data Buffer
- 4-Deep FIFO Receive Data Buffer
- Parity, Framing and Buffer Overrun Error Detection
- Support for 9-Bit mode with Address Detect (9^{th} bit = 1)
- Transmit and Receive Interrupts
- Loopback mode for Diagnostic Support
- Support for Sync and Break Characters
- Supports Automatic Baud Rate Detection
- IrDA® Encoder and Decoder Logic
- 16x Baud Clock Output for IrDA Support

A simplified block diagram of the UARTx is shown in Figure 18-1. The UARTx module consists of these key important hardware elements:

- Baud Rate Generator
- Asynchronous Transmitter
- Asynchronous Receiver

FIGURE 18-1: UARTx SIMPLIFIED BLOCK DIAGRAM



Note: The UARTx inputs and outputs must all be assigned to available RPn/RPIn pins before use. See **Section 11.4 "Peripheral Pin Select (PPS)"** for more information.

18.1 UARTx Baud Rate Generator (BRG)

The UARTx module includes a dedicated, 16-bit Baud Rate Generator. The UxBRG register controls the period of a free-running, 16-bit timer. Equation 18-1 shows the formula for computation of the baud rate with BRGH = 0.

EQUATION 18-1: UARTx BAUD RATE WITH BRGH = 0(1,2)

$$\text{Baud Rate} = \frac{FCY}{16 \bullet (UxBRG + 1)}$$

$$UxBRG = \frac{FCY}{16 \bullet \text{Baud Rate}} - 1$$

Note 1: FCY denotes the instruction cycle clock frequency (FOSC/2).
2: Based on FCY = FOSC/2; Doze mode and PLL are disabled.

Example 18-1 shows the calculation of the baud rate error for the following conditions:

- FCY = 4 MHz
- Desired Baud Rate = 9600

The maximum baud rate (BRGH = 0) possible is FCY/16 (for UxBRG = 0) and the minimum baud rate possible is FCY/(16 * 65536).

Equation 18-2 shows the formula for computation of the baud rate with BRGH = 1.

EQUATION 18-2: UARTx BAUD RATE WITH BRGH = 1(1,2)

$$\text{Baud Rate} = \frac{FCY}{4 \bullet (UxBRG + 1)}$$

$$UxBRG = \frac{FCY}{4 \bullet \text{Baud Rate}} - 1$$

Note 1: FCY denotes the instruction cycle clock frequency.
2: Based on FCY = FOSC/2; Doze mode and PLL are disabled.

The maximum baud rate (BRGH = 1) possible is FCY/4 (for UxBRG = 0) and the minimum baud rate possible is FCY/(4 * 65536).

Writing a new value to the UxBRG register causes the BRG timer to be reset (cleared). This ensures the BRG does not wait for a timer overflow before generating the new baud rate.

EXAMPLE 18-1: BAUD RATE ERROR CALCULATION (BRGH = 0)(1)

Desired Baud Rate = FCY/(16 (BRGx + 1))

Solving for BRGx Value:

BRGx = ((FCY/Desired Baud Rate)/16) – 1
BRGx = ((4000000/9600)/16) – 1
BRGx = 25

Calculated Baud Rate = 4000000/(16 (25 + 1))
= 9615

Error = (Calculated Baud Rate – Desired Baud Rate)/Desired Baud Rate
= (9615 – 9600)/9600
= 0.16%

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

18.2 Transmitting in 8-Bit Data Mode

1. Set up the UARTx:
   a) Write appropriate values for data, parity and Stop bits.
   b) Write appropriate baud rate value to the UxBRG register.
   c) Set up transmit and receive interrupt enable and priority bits.
2. Enable the UARTx.
3. Set the UTXEN bit (causes a transmit interrupt, two cycles after being set).
4. Write a data byte to the lower byte of the UxTXREG word. The value will be immediately transferred to the Transmit Shift Register (TSR) and the serial bit stream will start shifting out with the next rising edge of the baud clock.
5. Alternatively, the data byte may be transferred while UTXEN = 0 and then the user may set UTXEN. This will cause the serial bit stream to begin immediately because the baud clock will start from a cleared state.
6. A transmit interrupt will be generated as per interrupt control bit, UTXISELx.

18.3 Transmitting in 9-Bit Data Mode

1. Set up the UARTx (as described in **Section 18.2 "Transmitting in 8-Bit Data Mode"**).
2. Enable the UARTx.
3. Set the UTXEN bit (causes a transmit interrupt).
4. Write UxTXREG as a 16-bit value only.
5. A word write to UxTXREG triggers the transfer of the 9-bit data to the TSR. The serial bit stream will start shifting out with the first rising edge of the baud clock.
6. A transmit interrupt will be generated as per the setting of control bit, UTXISELx.

18.4 Break and Sync Transmit Sequence

The following sequence will send a message frame header, made up of a Break, followed by an auto-baud Sync byte.

1. Configure the UARTx for the desired mode.
2. Set UTXEN and UTXBRK to set up the Break character.
3. Load the UxTXREG with a dummy character to initiate transmission (value is ignored).
4. Write '55h' to UxTXREG; this loads the Sync character into the transmit FIFO.
5. After the Break has been sent, the UTXBRK bit is reset by hardware. The Sync character now transmits.

18.5 Receiving in 8-Bit or 9-Bit Data Mode

1. Set up the UARTx (as described in **Section 18.2 "Transmitting in 8-Bit Data Mode"**).
2. Enable the UARTx.
3. A receive interrupt will be generated when one or more data characters have been received, as per interrupt control bit, URXISELx.
4. Read the OERR bit to determine if an overrun error has occurred. The OERR bit must be reset in software.
5. Read UxRXREG.

The act of reading the UxRXREG character will move the next character to the top of the receive FIFO, including a new set of PERR and FERR values.

18.6 Operation of $\overline{\text{UxCTS}}$ and $\overline{\text{UxRTS}}$ Control Pins

UARTx Clear-to-Send ($\overline{\text{UxCTS}}$) and Request-to-Send ($\overline{\text{UxRTS}}$) are the two hardware controlled pins that are associated with the UARTx module. These two pins allow the UARTx to operate in Simplex and Flow Control mode. They are implemented to control the transmission and reception between the Data Terminal Equipment (DTE). The UEN<1:0> bits in the UxMODE register configure these pins.

18.7 Infrared Support

The UARTx module provides two types of infrared UART support: one is the IrDA clock output to support an external IrDA encoder and decoder device (legacy module support), and the other is the full implementation of the IrDA encoder and decoder. Note that because the IrDA modes require a 16x baud clock, they will only work when the BRGH bit (UxMODE<3>) is '0'.

18.7.1 IrDA CLOCK OUTPUT FOR EXTERNAL IrDA SUPPORT

To support external IrDA encoder and decoder devices, the BCLKx pin (same as the $\overline{\text{UxRTS}}$ pin) can be configured to generate the 16x baud clock. With UEN<1:0> = 11, the BCLKx pin will output the 16x baud clock if the UARTx module is enabled. It can be used to support the IrDA codec chip.

18.7.2 BUILT-IN IrDA ENCODER AND DECODER

The UARTx has full implementation of the IrDA encoder and decoder as part of the UARTx module. The built-in IrDA encoder and decoder functionality is enabled using the IREN bit (UxMODE<12>). When enabled (IREN = 1), the receive pin (UxRX) acts as the input from the infrared receiver. The transmit pin (UxTX) acts as the output to the infrared transmitter.

REGISTER 18-1: UxMODE: UARTx MODE REGISTER

R/W-0	U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
UARTEN(1)	—	USIDL	IREN(2)	RTSMD	—	UEN1	UEN0
bit 15							bit 8

R/W-0, HC	R/W-0	R/W-0, HC	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
WAKE	LPBACK	ABAUD	RXINV	BRGH	PDSEL1	PDSEL0	STSEL
bit 7							bit 0

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

bit 15 **UARTEN:** UARTx Enable bit(1)
- 1 = UARTx is enabled, all UARTx pins are controlled by UARTx as defined by UEN<1:0>
- 0 = UARTx is disabled, all UARTx pins are controlled by port latches; UARTx power consumption is minimal

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

bit 13 **USIDL:** UARTx Stop in Idle Mode bit
- 1 = Discontinues module operation when device enters Idle mode
- 0 = Continues module operation in Idle mode

bit 12 **IREN:** IrDA® Encoder and Decoder Enable bit(2)
- 1 = IrDA encoder and decoder are enabled
- 0 = IrDA encoder and decoder are disabled

bit 11 **RTSMD:** Mode Selection for $\overline{\text{UxRTS}}$ Pin bit
- 1 = $\overline{\text{UxRTS}}$ pin is in Simplex mode
- 0 = $\overline{\text{UxRTS}}$ pin is in Flow Control mode

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

bit 9-8 **UEN<1:0>:** UARTx Enable bits
- 11 = UxTX, UxRX and BCLKx pins are enabled and used; $\overline{\text{UxCTS}}$ pin is controlled by port latches
- 10 = UxTX, UxRX, $\overline{\text{UxCTS}}$ and $\overline{\text{UxRTS}}$ pins are enabled and used
- 01 = UxTX, UxRX and $\overline{\text{UxRTS}}$ pins are enabled and used; $\overline{\text{UxCTS}}$ pin is controlled by port latches
- 00 = UxTX and UxRX pins are enabled and used; $\overline{\text{UxCTS}}$ and $\overline{\text{UxRTS}}$/BCLKx pins are controlled by port latches

bit 7 **WAKE:** Wake-up on Start Bit Detect During Sleep Mode Enable bit
- 1 = UARTx will continue to sample the UxRX pin; interrupt is generated on the falling edge, bit is cleared in hardware on the following rising edge
- 0 = No wake-up is enabled

bit 6 **LPBACK:** UARTx Loopback Mode Select bit
- 1 = Enables Loopback mode
- 0 = Loopback mode is disabled

bit 5 **ABAUD:** Auto-Baud Enable bit
- 1 = Enables baud rate measurement on the next character – requires reception of a Sync field (55h); cleared in hardware upon completion
- 0 = Baud rate measurement is disabled or completed

Note 1: If UARTEN = 1, the peripheral inputs and outputs must be configured to an available RPn/RPIn pin. See **Section 11.4 "Peripheral Pin Select (PPS)"** for more information.

2: This feature is only available for the 16x BRG mode (BRGH = 0).

REGISTER 18-1: UxMODE: UARTx MODE REGISTER (CONTINUED)

bit 4 **RXINV:** Receive Polarity Inversion bit
1 = UxRX Idle state is '0'
0 = UxRX Idle state is '1'

bit 3 **BRGH:** High Baud Rate Enable bit
1 = High-Speed mode (4 BRG clock cycles per bit)
0 = Standard Speed mode (16 BRG clock cycles per bit)

bit 2-1 **PDSEL<1:0>:** Parity and Data Selection bits
11 = 9-bit data, no parity
10 = 8-bit data, odd parity
01 = 8-bit data, even parity
00 = 8-bit data, no parity

bit 0 **STSEL:** Stop Bit Selection bit
1 = Two Stop bits
0 = One Stop bit

Note 1: If UARTEN = 1, the peripheral inputs and outputs must be configured to an available RPn/RPIn pin. See **Section 11.4 "Peripheral Pin Select (PPS)"** for more information.

2: This feature is only available for the 16x BRG mode (BRGH = 0).

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, HSC	R-1, HSC
UTXISEL1	UTXINV(1)	UTXISEL0	—	UTXBRK	UTXEN(2)	UTXBF	TRMT
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R-1, HSC	R-0, HSC	R-0, HSC	R/C-0, HS	R-0, HSC
URXISEL1	URXISEL0	ADDEN	RIDLE	PERR	FERR	OERR	URXDA
bit 7							bit 0

Legend:	C = Clearable bit	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
HS = Hardware Settable bit	HC = Hardware Clearable bit		

bit 15,13 **UTXISEL<1:0>:** UARTx Transmission Interrupt Mode Selection bits
- 11 = Reserved; do not use
- 10 = Interrupt when a character is transferred to the Transmit Shift Register (TSR), 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:** IrDA® Encoder Transmit Polarity Inversion bit(1)

IREN = 0:
- 1 = UxTX is Idle '0'
- 0 = UxTX is Idle '1'

IREN = 1:
- 1 = UxTX is Idle '1'
- 0 = UxTX is Idle '0'

bit 12 **Unimplemented:** Read as '0'

bit 11 **UTXBRK:** UARTx Transmit Break bit
- 1 = Sends 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 is disabled or completed

bit 10 **UTXEN:** UARTx Transmit Enable bit(2)
- 1 = Transmit is enabled, UxTX pin is controlled by UARTx
- 0 = Transmit is disabled, any pending transmission is aborted and the buffer is reset; UxTX pin is controlled by the port

bit 9 **UTXBF:** UARTx 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

Note 1: The value of the bit only affects the transmit properties of the module when the IrDA® encoder is enabled (IREN = 1).

2: If UARTEN = 1, the peripheral inputs and outputs must be configured to an available RPn/RPIn pin. See **Section 11.4 "Peripheral Pin Select (PPS)"** for more information.

REGISTER 18-2: UxSTA: UARTx STATUS AND CONTROL REGISTER (CONTINUED)

bit 7-6 **URXISEL<1:0>:** UARTx Receive Interrupt Mode Selection bits
11 = Interrupt is set on an RSR transfer, making the receive buffer full (i.e., has 4 data characters)
10 = Interrupt is set on an RSR 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 RSR to the receive buffer; receive buffer has one or more characters

bit 5 **ADDEN:** Address Character Detect bit (bit 8 of received data = 1)
1 = Address Detect mode is enabled (if 9-bit mode is not selected, this does not take effect)
0 = Address Detect mode is disabled

bit 4 **RIDLE:** Receiver Idle bit (read-only)
1 = Receiver is Idle
0 = Receiver is active

bit 3 **PERR:** Parity Error Status bit (read-only)
1 = Parity error has been detected for the current character (character at the top of the receive FIFO)
0 = Parity error has not been detected

bit 2 **FERR:** Framing Error Status bit (read-only)
1 = Framing error has been detected for the current character (character at the top of the receive FIFO)
0 = Framing error has not been detected

bit 1 **OERR:** Receive Buffer Overrun Error Status bit (clear/read-only)
1 = Receive buffer has overflowed
0 = Receive buffer has not overflowed (clearing a previously set OERR bit (1 → 0 transition); will reset the receiver buffer and the RSR to the empty state

bit 0 **URXDA:** UARTx Receive Buffer Data Available bit (read-only)
1 = Receive buffer has data, at least one more character can be read
0 = Receive buffer is empty

Note 1: The value of the bit only affects the transmit properties of the module when the IrDA® encoder is enabled (IREN = 1).

2: If UARTEN = 1, the peripheral inputs and outputs must be configured to an available RPn/RPIn pin. See **Section 11.4 "Peripheral Pin Select (PPS)"** for more information.

NOTES:

19.0 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"*, **"USB On-The-Go (OTG)"** (DS39721). The information in this data sheet supersedes the information in the FRM.

PIC24FJ128GC010 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 as either 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 19-1.

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 19-1 shows the relationship between data direction in this nomenclature and the USB tokens exchanged.

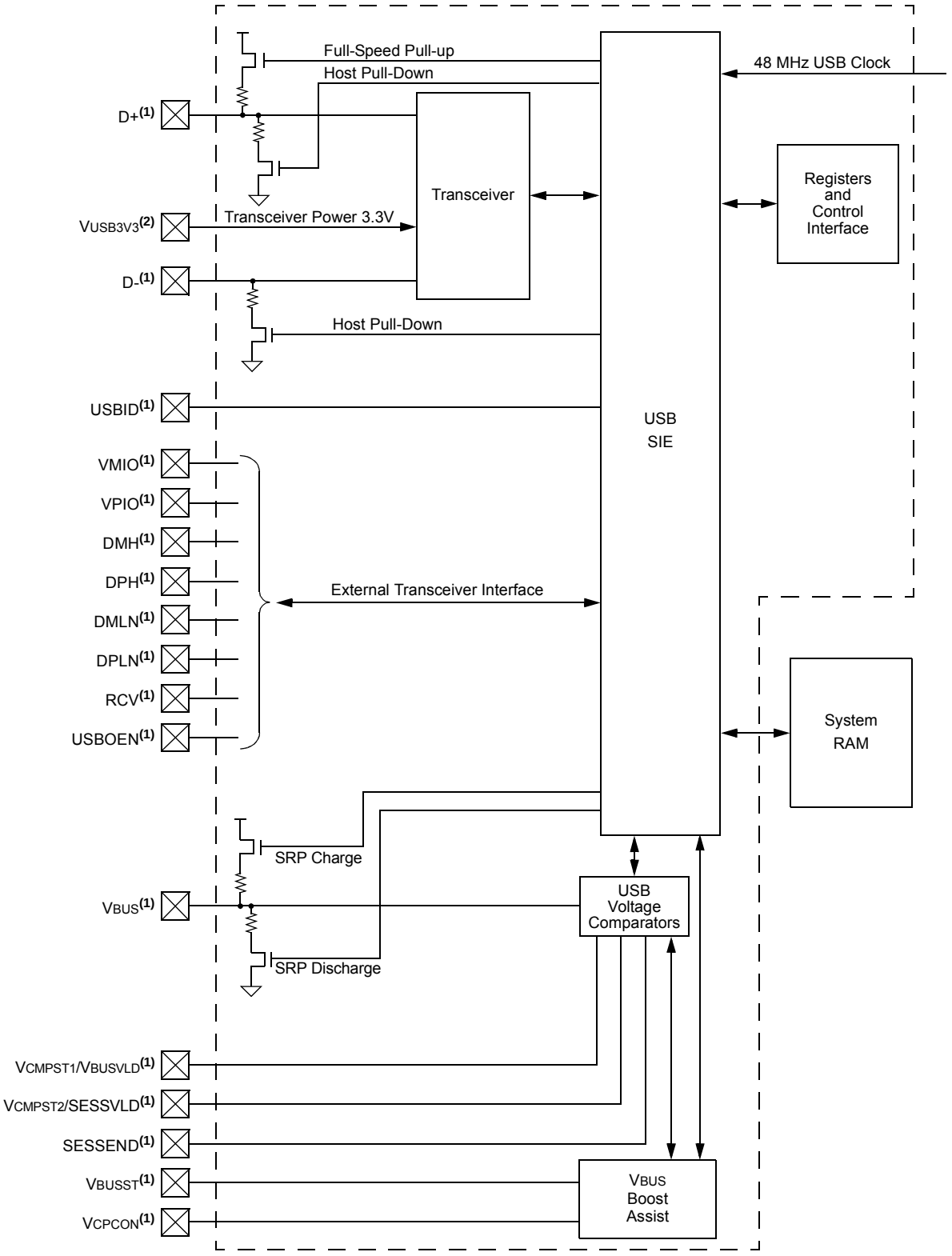
TABLE 19-1: CONTROLLER-CENTRIC DATA DIRECTION FOR USB HOST OR TARGET

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.

FIGURE 19-1: USB OTG MODULE BLOCK DIAGRAM



Note 1: Pins are multiplexed with digital I/O and other device features.

2: Connecting V_{BUS3V3} to V_{DD} is highly recommended, as floating this input can cause increased I_{PD} currents. The pin should be tied to V_{DD} when the USB functions are not used.

19.1 Hardware Configuration

19.1.1 DEVICE MODE

19.1.1.1 D+ Pull-up Resistor

PIC24FJ128GC010 family devices have a built-in 1.5 kΩ resistor on the D+ line that is available when the microcontroller is operating in Device mode. This is used to signal an external host that the device is operating in Full-Speed Device mode. It is engaged by setting the USBEN bit (U1CON<0>) and powering up the USB module (USBPWR = 1). If the OTGEN bit (U1OTGCON<2>) is set, then the D+ pull-up is enabled through the DPPULUP bit (U1OTGCON<7>).

19.1.1.2 The VBUS Pin

In order to meet the USB 2.0 specification requirement, relating to the back drive voltage on the D+/D- pins, the USB module incorporates VBUS level sensing comparators. When the comparators detect the VBUS level below the VA_SESS_VLD level, the hardware will automatically disable the D+ pull-up resistor described in **Section 19.1.1.1 "D+ Pull-up Resistor"**. This allows the device to automatically meet the back drive requirement for D+ and D-, even if the application firmware does not explicitly monitor the VBUS level. Therefore, the VBUS microcontroller pin should not be left floating in USB Device mode application designs, and should normally be connected to the VBUS pin on the USB connector/cable (either directly or through a small resistance ≤ 100 ohms).

19.1.1.3 Power Modes

Many USB applications will likely have several different sets of power requirements and configuration. The most common power modes encountered are:

- Bus Power Only mode
- Self-Power Only mode
- Dual Power with Self-Power Dominance

Bus Power Only mode (Figure 19-2) is effectively the simplest method. All power for the application is drawn from the USB.

To meet the inrush current requirements of the *"USB 2.0 OTG Specification"*, the total effective capacitance, appearing across VBUS and ground, must be no more than 10 µF.

In the USB Suspend mode, devices must consume no more than 2.5 mA from the 5V VBUS line of the USB cable. During the USB Suspend mode, the D+ or D- pull-up resistor must remain active, which will consume some of the allowed suspend current.

In Self-Power Only mode (Figure 19-3), the USB application provides its own power, with very little power being pulled from the USB. Note that an attach indication is added to indicate when the USB has been connected and the host is actively powering VBUS.

To meet compliance specifications, the USB module (and the D+ or D- pull-up resistor) should not be enabled until the host actively drives VBUS high. One of the 5.5V tolerant I/O pins may be used for this purpose.

The application should never source any current onto the 5V VBUS pin of the USB cable when the USB module is operated in USB Device mode.

The Dual Power mode with Self-Power Dominance (Figure 19-4) allows the application to use internal power primarily, but switch to power from the USB when no internal power is available. Dual power devices must also meet all of the special requirements for inrush current and Suspend mode current previously described, and must not enable the USB module until VBUS is driven high.

FIGURE 19-2: BUS-POWERED INTERFACE EXAMPLE

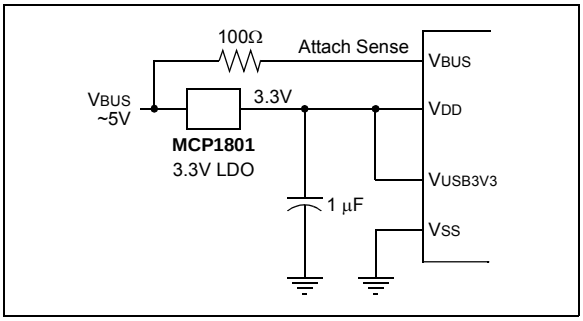


FIGURE 19-3: SELF-POWER ONLY

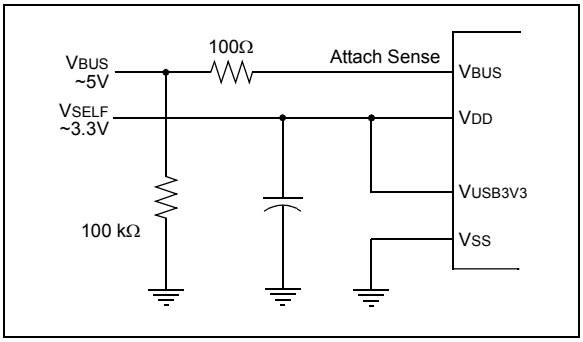
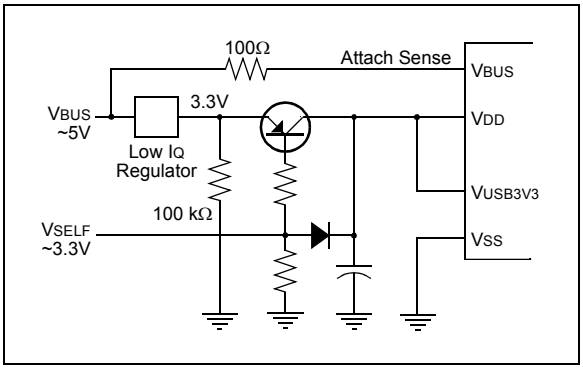


FIGURE 19-4: DUAL POWER EXAMPLE



19.1.2 HOST AND OTG MODES

19.1.2.1 D+ and D- Pull-Down Resistors

PIC24FJ128GC010 family devices have a built-in 15 kΩ pull-down resistor on the D+ and D- lines. These are used in tandem to signal to the bus that the microcontroller is operating in Host mode. They are engaged by setting the HOSTEN bit (U1CON<3>). If the OTGEN bit (U1OTGCON<2>) is set, then these pull-downs are enabled by setting the DPPULDWN and DMPULDWN bits (U1OTGCON<5:4>).

19.1.2.2 Power Configurations

In Host mode, as well as Host mode in On-The-Go operation, the *"USB 2.0 OTG Specification"* requires that the host application should supply power on VBUS. Since the microcontroller is running below VBUS, and is not able to source sufficient current, a separate power supply must be provided.

When the application is always operating in Host mode, a simple circuit can be used to supply VBUS and regulate current on the bus (Figure 19-5). For OTG operation, it is necessary to be able to turn VBUS on or off as needed, as the microcontroller switches between Device and Host modes. A typical example using an external charge pump is shown in Figure 19-6.

FIGURE 19-5: HOST INTERFACE EXAMPLE

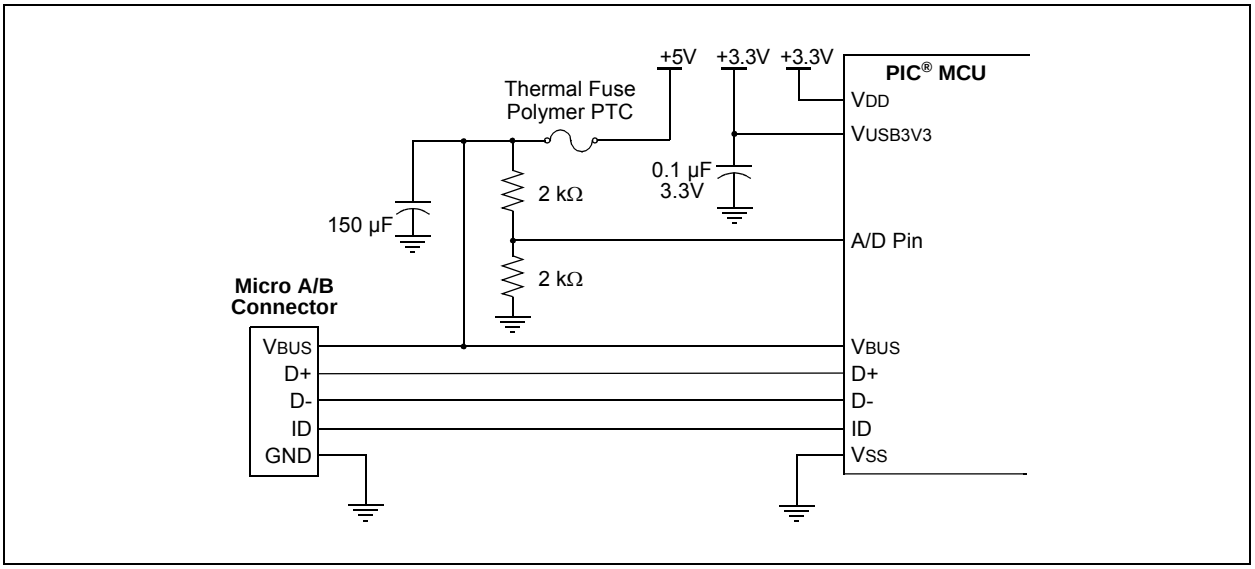
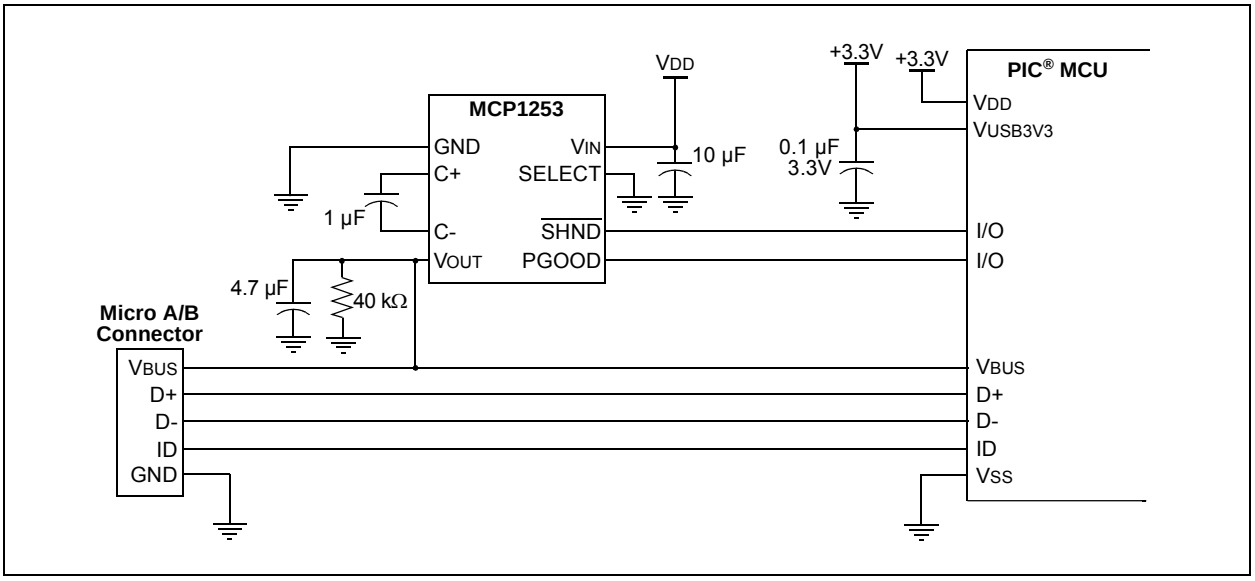


FIGURE 19-6: OTG INTERFACE EXAMPLE



19.1.3 USING AN EXTERNAL INTERFACE

Some applications may require the USB interface to be isolated from the rest of the system. PIC24FJ128GC010 family devices include a complete interface to communicate with and control an external USB transceiver, including the control of data line pull-ups and pull-downs. The VBUS voltage generation control circuit can also be configured for different VBUS generation topologies.

Refer to the "*PIC24F Family Reference Manual*", **Section 27. "USB On-The-Go (OTG)"** for information on using the external interface.

19.1.4 CALCULATING TRANSCEIVER POWER REQUIREMENTS

The USB transceiver consumes a variable amount of current depending on the characteristic impedance of the USB cable, the length of the cable, the VUSB supply voltage and the actual data patterns moving across the USB cable. Longer cables have larger capacitances and consume more total energy when switching output states. The total transceiver current consumption will be application-specific. Equation 19-1 can help estimate how much current actually may be required in full-speed applications.

Refer to the "*PIC24F Family Reference Manual*", **Section 27. "USB On-The-Go (OTG)"** for a complete discussion on transceiver power consumption.

EQUATION 19-1: ESTIMATING USB TRANSCEIVER CURRENT CONSUMPTION

$$IXCVR = \frac{40\ mA \bullet V_{USB} \bullet P_{ZERO} \bullet P_{IN} \bullet L_{CABLE}}{3.3V \bullet 5m} + I_{PULLUP}$$

Legend: VUSB – Voltage applied to the VUSB3V3 pin in volts (3.0V to 3.6V).

PZERO – Percentage (in decimal) of the IN traffic bits sent by the PIC® microcontroller that are a value of '0'.

PIN – Percentage (in decimal) of total bus bandwidth that is used for IN traffic.

LCABLE – Length (in meters) of the USB cable. The *"USB 2.0 OTG Specification"* requires that full-speed applications use cables no longer than 5m.

IPULLUP – Current which the nominal, 1.5 kΩ pull-up resistor (when enabled) must supply to the USB cable.

19.2 USB Buffer Descriptors and the BDT

Endpoint buffer control is handled through a structure called the Buffer Descriptor Table (BDT). This provides a flexible method for users to construct and control endpoint buffers of various lengths and configurations.

The BDT can be located in any available 512-byte, aligned block of data RAM. The BDT Pointer (U1BDTP1) contains the upper address byte of the BDT and sets the location of the BDT in RAM. The user must set this pointer to indicate the table's location.

The BDT is composed of Buffer Descriptors (BDs) which are used to define and control the actual buffers in the USB RAM space. Each BD consists of two 16-bit, "soft" (non-fixed-address) registers, BDnSTAT and BDnADR, where n represents one of the 64 possible BDs (range of 0 to 63). BDnSTAT is the status register for BDn, while BDnADR specifies the starting address for the buffer associated with BDn.

Note: Since BDnADR is a 16-bit register, only the first 64 Kbytes of RAM can be accessed by the USB module.

Depending on the endpoint buffering configuration used, there are up to 64 sets of Buffer Descriptors, for a total of 256 bytes. At a minimum, the BDT must be at least 8 bytes long. This is because the *"USB 2.0 OTG Specification"* mandates that every device must have Endpoint 0 with both input and output for initial setup.

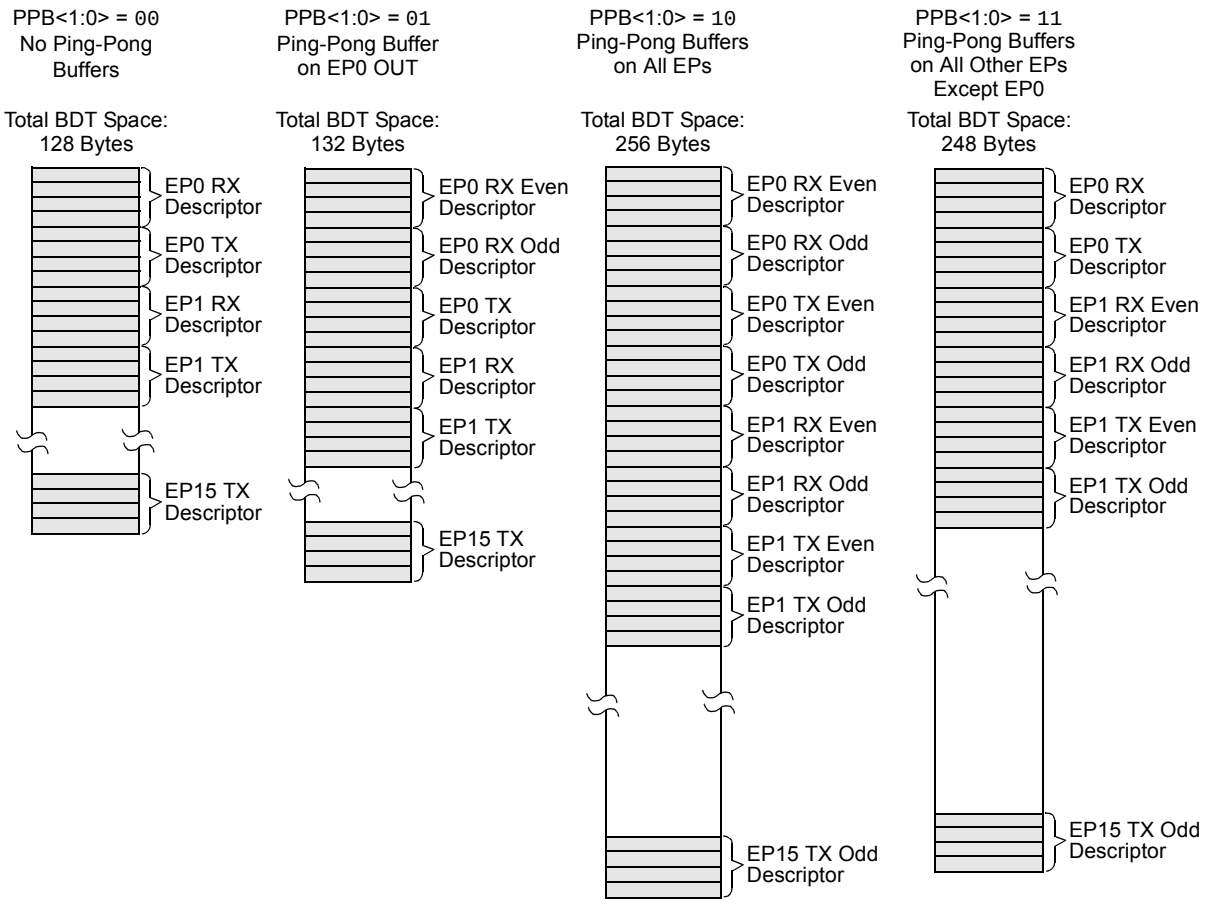
Endpoint mapping in the BDT is dependent on three variables:

- Endpoint number (0 to 15)
- Endpoint direction (RX or TX)
- Ping-pong settings (U1CNFG1<1:0>)

Figure 19-7 illustrates how these variables are used to map endpoints in the BDT.

In Host mode, only Endpoint 0 Buffer Descriptors are used. All transfers utilize the Endpoint 0 Buffer Descriptor and Endpoint Control register (U1EP0). For received packets, the attached device's source endpoint is indicated by the value of ENDPT<3:0> in the USB status register (U1STAT<7:4>). For transmitted packets, the attached device's destination endpoint is indicated by the value written to the Token register (U1TOK).

FIGURE 19-7: BDT MAPPING FOR ENDPOINT BUFFERING MODES



Note: Memory area is not shown to scale.

BDs have a fixed relationship to a particular endpoint, depending on the buffering configuration. Table 19-2 provides the mapping of BDs to endpoints. This relationship also means that gaps may occur in the BDT if endpoints are not enabled contiguously. This, theoretically, means that the BDs for disabled endpoints could be used as buffer space. In practice, users should avoid using such spaces in the BDT unless a method of validating BD addresses is implemented.

19.2.1 BUFFER OWNERSHIP

Because the buffers and their BDs are shared between the CPU and the USB module, a simple semaphore mechanism is used to distinguish which is allowed to update the BD and associated buffers in memory. This is done by using the UOWN bit as a semaphore to distinguish which is allowed to update the BD and associated buffers in memory. UOWN is the only bit that is shared between the two configurations of BDnSTAT.

When UOWN is clear, the BD entry is “owned” by the microcontroller core. When the UOWN bit is set, the BD entry and the buffer memory are “owned” by the USB peripheral. The core should not modify the BD or its corresponding data buffer during this time. Note that the microcontroller core can still read BDnSTAT while the SIE owns the buffer and vice versa.

The Buffer Descriptors have a different meaning based on the source of the register update. Register 19-1 and Register 19-2 show the differences in BDnSTAT depending on its current “ownership”.

When UOWN is set, the user can no longer depend on the values that were written to the BDs. From this point, the USB module updates the BDs as necessary, overwriting the original BD values. The BDnSTAT register is updated by the SIE with the token PID and the transfer count is updated.

19.2.2 DMA INTERFACE

The USB OTG module uses a dedicated DMA to access both the BDT and the endpoint data buffers. Since part of the address space of the DMA is dedicated to the Buffer Descriptors, a portion of the memory connected to the DMA must comprise a contiguous address space, properly mapped for the access by the module.

TABLE 19-2: ASSIGNMENT OF BUFFER DESCRIPTORS FOR THE DIFFERENT BUFFERING MODES

Endpoint	BDs Assigned to Endpoint							
	Mode 0 (No Ping-Pong)		Mode 1 (Ping-Pong on EP0 OUT)		Mode 2 (Ping-Pong on All EPs)		Mode 3 (Ping-Pong on All Other EPs, Except EP0)	
	Out	In	Out	In	Out	In	Out	In
0	0	1	0 (E), 1 (O)	2	0 (E), 1 (O)	2 (E), 3 (O)	0	1
1	2	3	3	4	4 (E), 5 (O)	6 (E), 7 (O)	2 (E), 3 (O)	4 (E), 5 (O)
2	4	5	5	6	8 (E), 9 (O)	10 (E), 11 (O)	6 (E), 7 (O)	8 (E), 9 (O)
3	6	7	7	8	12 (E), 13 (O)	14 (E), 15 (O)	10 (E), 11 (O)	12 (E), 13 (O)
4	8	9	9	10	16 (E), 17 (O)	18 (E), 19 (O)	14 (E), 15 (O)	16 (E), 17 (O)
5	10	11	11	12	20 (E), 21 (O)	22 (E), 23 (O)	18 (E), 19 (O)	20 (E), 21 (O)
6	12	13	13	14	24 (E), 25 (O)	26 (E), 27 (O)	22 (E), 23 (O)	24 (E), 25 (O)
7	14	15	15	16	28 (E), 29 (O)	30 (E), 31 (O)	26 (E), 27 (O)	28 (E), 29 (O)
8	16	17	17	18	32 (E), 33 (O)	34 (E), 35 (O)	30 (E), 31 (O)	32 (E), 33 (O)
9	18	19	19	20	36 (E), 37 (O)	38 (E), 39 (O)	34 (E), 35 (O)	36 (E), 37 (O)
10	20	21	21	22	40 (E), 41 (O)	42 (E), 43 (O)	38 (E), 39 (O)	40 (E), 41 (O)
11	22	23	23	24	44 (E), 45 (O)	46 (E), 47 (O)	42 (E), 43 (O)	44 (E), 45 (O)
12	24	25	25	26	48 (E), 49 (O)	50 (E), 51 (O)	46 (E), 47 (O)	48 (E), 49 (O)
13	26	27	27	28	52 (E), 53 (O)	54 (E), 55 (O)	50 (E), 51 (O)	52 (E), 53 (O)
14	28	29	29	30	56 (E), 57 (O)	58 (E), 59 (O)	54 (E), 55 (O)	56 (E), 57 (O)
15	30	31	31	32	60 (E), 61 (O)	62 (E), 63 (O)	58 (E), 59 (O)	60 (E), 61 (O)

Legend: (E) = Even transaction buffer, (O) = Odd transaction buffer

REGISTER 19-1: BDnSTAT: BUFFER DESCRIPTOR n STATUS REGISTER PROTOTYPE, USB MODE (BD0STAT THROUGH BD63STAT)

R/W-x	R/W-x	R/W-x, HSC	R/W-x, HSC	R/W-x, HSC	R/W-x, HSC	R/W-x, HSC	R/W-x, HSC
UOWN	DTS	PID3	PID2	PID1	PID0	BC9	BC8
bit 15							bit 8

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	R/W-x, HSC
BC7	BC6	BC5	BC4	BC3	BC2	BC1	BC0
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 **UOWN:** USB Own bit

1 = The USB module owns the BD and its corresponding buffer; the CPU must not modify the BD or the buffer

bit 14 **DTS:** Data Toggle Packet bit

1 = Data 1 packet
0 = Data 0 packet

bit 13-10 **PID<3:0>:** Packet Identifier bits (written by the USB module)

In Device mode:
Represents the PID of the received token during the last transfer.

In Host mode:
Represents the last returned PID or the transfer status indicator.

bit 9-0 **BC<9:0>:** Byte Count bits

This represents the number of bytes to be transmitted or the maximum number of bytes to be received during a transfer. Upon completion, the byte count is updated by the USB module with the actual number of bytes transmitted or received.

REGISTER 19-2: BDnSTAT: BUFFER DESCRIPTOR n STATUS REGISTER PROTOTYPE, CPU MODE (BD0STAT THROUGH BD63STAT)

R/W-x	R/W-x	r-0	r-0	R/W-x	R/W-x	R/W-x, HSC	R/W-x, HSC
UOWN	DTS(1)	r	r	DTSEN	BSTALL	BC9	BC8
bit 15							bit 8

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	R/W-x, HSC
BC7	BC6	BC5	BC4	BC3	BC2	BC1	BC0
bit 7							bit 0

Legend:	r = Reserved bit	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	'r' = Reserved bit	x = Bit is unknown

bit 15 **UOWN:** USB Own bit
- 0 = The microcontroller core owns the BD and its corresponding buffer; the USB module ignores all other fields in the BD

bit 14 **DTS:** Data Toggle Packet bit(1)
- 1 = Data 1 packet
- 0 = Data 0 packet

bit 13-12 **Reserved:** Maintain as '0'

bit 11 **DTSEN:** Data Toggle Synchronization Enable bit
- 1 = Data toggle synchronization is enabled; data packets with incorrect Sync value will be ignored
- 0 = No data toggle synchronization is performed

bit 10 **BSTALL:** Buffer STALL Enable bit
- 1 = Buffer STALL is enabled; STALL handshake issued if a token is received that would use the BD in the given location (UOWN bit remains set, BD value is unchanged); corresponding EPSTALL bit will get set on any STALL handshake
- 0 = Buffer STALL is disabled

bit 9-0 **BC<9:0>:** Byte Count bits

This represents the number of bytes to be transmitted or the maximum number of bytes to be received during a transfer. Upon completion, the byte count is updated by the USB module with the actual number of bytes transmitted or received.

Note 1: This bit is ignored unless DTSEN = 1.

19.3 USB Interrupts

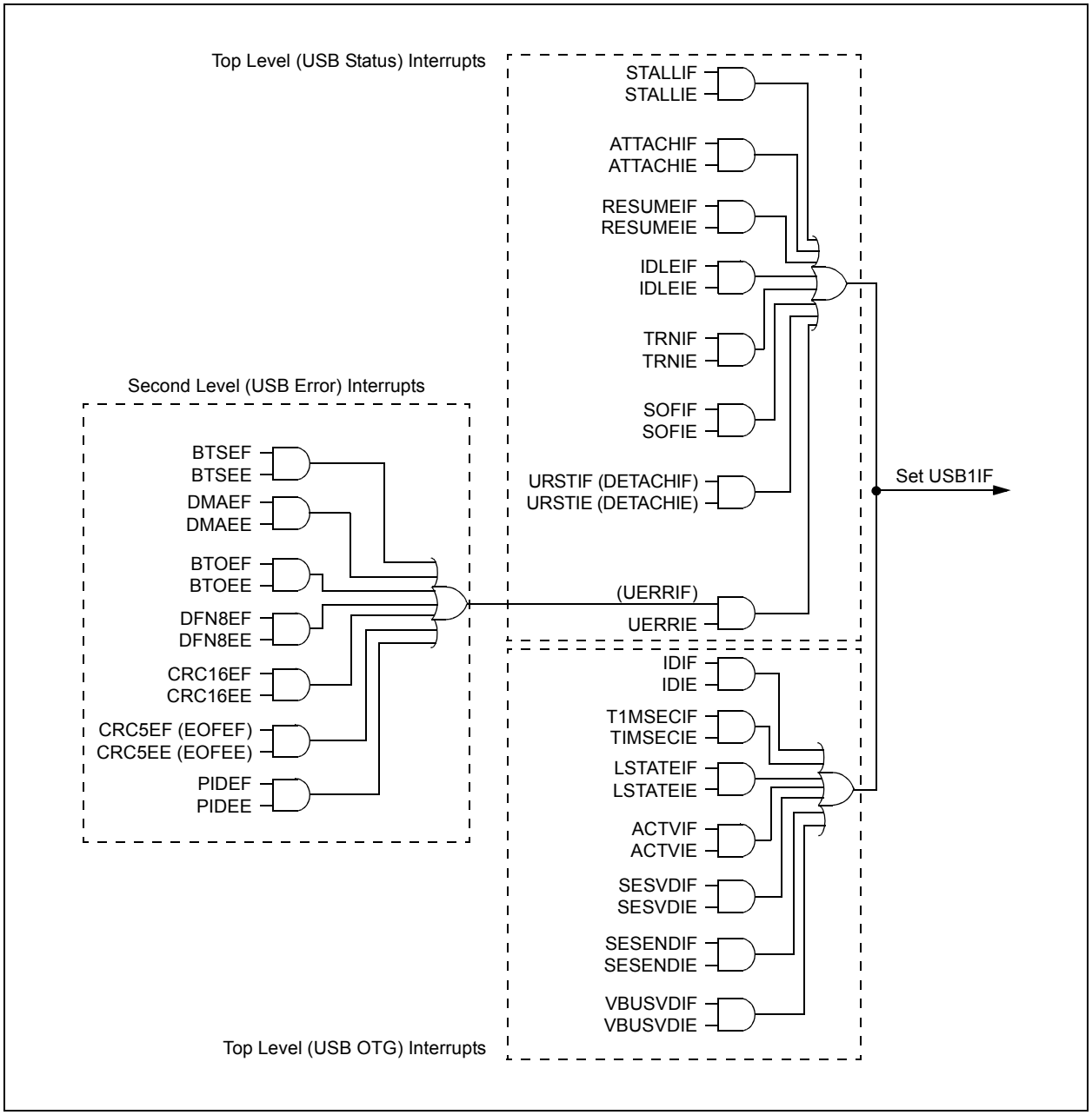
The USB OTG module has many conditions that can be configured to cause an interrupt. All interrupt sources use the same interrupt vector.

Figure 19-8 shows the interrupt logic for the USB module. There are two layers of interrupt registers in the USB module. The top level consists of overall USB status interrupts; these are enabled and flagged in the U1IE and U1IR registers, respectively. The second level consists of USB error conditions, which are enabled and flagged in the U1EIR and U1EIE registers.

An interrupt condition in any of these triggers a USB Error Interrupt Flag (UERRIF) in the top level. Unlike the device-level interrupt flags in the IFSx registers, USB interrupt flags in the U1IR registers can only be cleared by writing a '1' to the bit position.

Interrupts may be used to trap routine events in a USB transaction. Figure 19-9 provides some common events within a USB frame and their corresponding interrupts.

FIGURE 19-8: USB OTG INTERRUPT FUNNEL

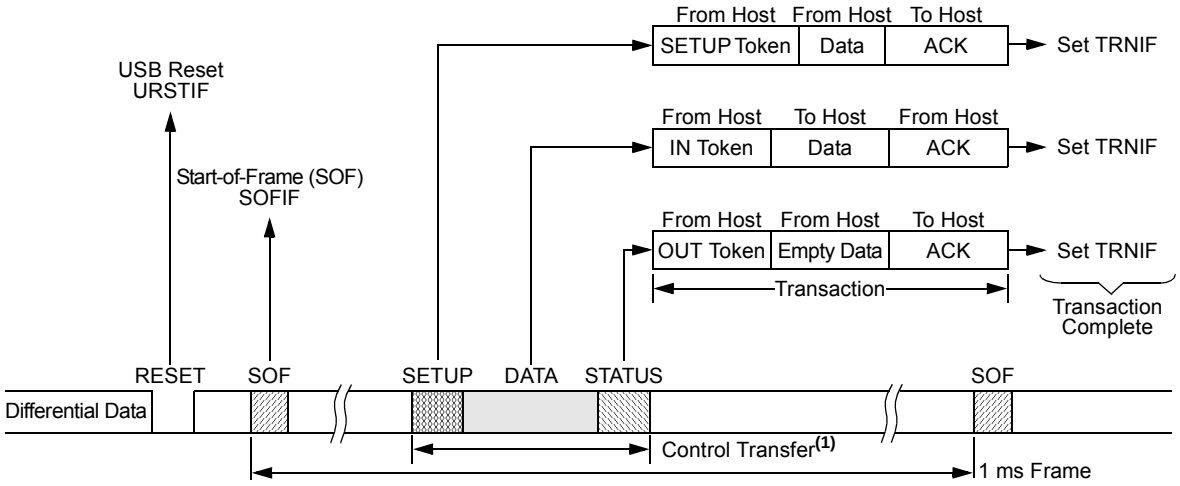


19.3.1 CLEARING USB OTG INTERRUPTS

Unlike device level interrupts, the USB OTG interrupt status flags are not freely writable in software. All USB OTG flag bits are implemented as hardware set only bits. Additionally, these bits can only be cleared in software by writing a '1' to their locations (i.e., performing a MOV type instruction). Writing a '0' to a flag bit (i.e., a BCLR instruction) has no effect.

Note:	Throughout this data sheet, a bit that can only be cleared by writing a '1' to its location is referred to as "Write 1 to clear". In register descriptions, this function is indicated by the descriptor, "K".

FIGURE 19-9: EXAMPLE OF A USB TRANSACTION AND INTERRUPT EVENTS



Note 1: The control transfer shown here is only an example showing events that can occur for every transaction. Typical control transfers will spread across multiple frames.

19.4 Device Mode Operation

The following section describes how to perform a common Device mode task. In Device mode, USB transfers are performed at the transfer level. The USB module automatically performs the status phase of the transfer.

19.4.1 ENABLING DEVICE MODE

1. Reset the Ping-Pong Buffer Pointers by setting, then clearing, the Ping-Pong Buffer Reset bit, PPBRST (U1CON<1>).
2. Disable all interrupts (U1IE and U1EIE = 00h).
3. Clear any existing interrupt flags by writing FFh to U1IR and U1EIR.
4. Verify that VBUS is present (non-OTG devices only).
5. Enable the USB module by setting the USBEN bit (U1CON<0>).
6. Set the OTGEN bit (U1OTGCON<2>) to enable OTG operation.
7. Enable the endpoint zero buffer to receive the first setup packet by setting the EPRXEN and EPHSHK bits for Endpoint 0 (U1EP0<3,0> = 1).
8. Power up the USB module by setting the USBPWR bit (U1PWRC<0>).
9. Enable the D+ pull-up resistor to signal an attach by setting the DPPULUP bit (U1OTGCON<7>).

19.4.2 RECEIVING AN IN TOKEN IN DEVICE MODE

1. Attach to a USB host and enumerate as described in Chapter 9 of the *"USB 2.0 Specification"*.
2. Create a data buffer and populate it with the data to send to the host.
3. In the appropriate (even or odd) TX BD for the desired endpoint:
   a) Set up the status register (BDnSTAT) with the correct data toggle (DATA0/1) value and the byte count of the data buffer.
   b) Set up the address register (BDnADR) with the starting address of the data buffer.
   c) Set the UOWN bit of the status register to '1'.
4. When the USB module receives an IN token, it automatically transmits the data in the buffer. Upon completion, the module updates the status register (BDnSTAT) and sets the Token Complete Interrupt Flag, TRNIF (U1IR<3>).

19.4.3 RECEIVING AN OUT TOKEN IN DEVICE MODE

1. Attach to a USB host and enumerate as described in Chapter 9 of the *"USB 2.0 Specification"*.
2. Create a data buffer with the amount of data you are expecting from the host.
3. In the appropriate (even or odd) TX BD for the desired endpoint:
   a) Set up the status register (BDnSTAT) with the correct data toggle (DATA0/1) value and the byte count of the data buffer.
   b) Set up the address register (BDnADR) with the starting address of the data buffer.
   c) Set the UOWN bit of the status register to '1'.
4. When the USB module receives an OUT token, it automatically receives the data sent by the host to the buffer. Upon completion, the module updates the status register (BDnSTAT) and sets the Token Complete Interrupt Flag, TRNIF (U1IR<3>).

19.5 Host Mode Operation

The following sections describe how to perform common Host mode tasks. In Host mode, USB transfers are invoked explicitly by the host software. The host software is responsible for the Acknowledge portion of the transfer. Also, all transfers are performed using the Endpoint 0 Control register (U1EP0) and Buffer Descriptors.

19.5.1 ENABLE HOST MODE AND DISCOVER A CONNECTED DEVICE

1. Enable Host mode by setting the HOSTEN bit (U1CON<3>). This causes the Host mode control bits in other USB OTG registers to become available.
2. Enable the D+ and D- pull-down resistors by setting the DPPULDWN and DMPULDWN bits (U1OTGCON<5:4>). Disable the D+ and D- pull-up resistors by clearing the DPPULUP and DMPULUP bits (U1OTGCON<7:6>).
3. At this point, SOF generation begins with the SOF counter loaded with 12,000. Eliminate noise on the USB by clearing the SOFEN bit (U1CON<0>) to disable Start-of-Frame (SOF) packet generation.
4. Enable the device attached interrupt by setting the ATTACHIE bit (U1IE<6>).
5. Wait for the device attached interrupt (U1IR<6> = 1). This is signaled by the USB device changing the state of D+ or D- from '0' to '1' (SE0 to J-state). After it occurs, wait 100 ms for the device power to stabilize.
6. Check the state of the JSTATE and SE0 bits in U1CON. If the JSTATE bit (U1CON<7>) is '0', the connecting device is low speed. If the connecting device is low speed, set the low LSPDEN and LSPD bits (U1ADDR<7> and U1EP0<7>) to enable low-speed operation.
7. Reset the USB device by setting the USBRST bit (U1CON<4>) for at least 50 ms, sending Reset signaling on the bus. After 50 ms, terminate the Reset by clearing USBRST.
8. In order to keep the connected device from going into suspend, enable the SOF packet generation by setting the SOFEN bit.
9. Wait 10 ms for the device to recover from Reset.
10. Perform enumeration as described by Chapter 9 of the *"USB 2.0 Specification"*.

19.5.2 COMPLETE A CONTROL TRANSACTION TO A CONNECTED DEVICE

1. Follow the procedure described in **Section 19.5.1 "Enable Host Mode and Discover a Connected Device"** to discover a device.
2. Set up the Endpoint Control register for bidirectional control transfers by writing 0Dh to U1EP0 (this sets the EPCONDIS, EPTXEN and EPHSHK bits).
3. Place a copy of the device framework setup command in a memory buffer. See Chapter 9 of the *"USB 2.0 Specification"* for information on the device framework command set.
4. Initialize the Buffer Descriptor (BD) for the current (even or odd) TX EP0 to transfer the eight bytes of command data for a device framework command (i.e., `GET DEVICE DESCRIPTOR`):
   a) Set the BD Data Buffer Address (BD0ADR) to the starting address of the 8-byte memory buffer containing the command.
   b) Write 8008h to BD0STAT (this sets the UOWN bit and sets a byte count of 8).
5. Set the USB device address of the target device in the address register (U1ADDR<6:0>). After a USB bus Reset, the device USB address will be zero. After enumeration, it will be set to another value between 1 and 127.
6. Write D0h to U1TOK; this is a SETUP token to Endpoint 0, the target device's default control pipe. This initiates a SETUP token on the bus, followed by a data packet. The device handshake is returned in the PID field of BD0STAT after the packets are complete. When the USB module updates BD0STAT, a Token Complete Interrupt Flag is asserted (the TRNIF flag is set). This completes the setup phase of the setup transaction, as referenced in Chapter 9 of the *"USB 2.0 Specification"*.
7. To initiate the data phase of the setup transaction (i.e., get the data for the `GET DEVICE DESCRIPTOR` command), set up a buffer in memory to store the received data.
8. Initialize the current (even or odd) RX or TX (RX for IN, TX for OUT) EP0 BD to transfer the data.
   a) Write C040h to BD0STAT. This sets the UOWN, configures Data Toggle (DTS) to DATA1 and sets the byte count to the length of the data buffer (64 or 40h in this case).
   b) Set BD0ADR to the starting address of the data buffer.
9. Write the Token register with the appropriate IN or OUT token to Endpoint 0, the target device's default control pipe (e.g., write 90h to U1TOK for an IN token for a `GET DEVICE DESCRIPTOR` command). This initiates an IN token on the bus, followed by a data packet from the device to the host. When the data packet completes, the BD0STAT is written and a Token Complete Interrupt Flag is asserted (the TRNIF flag is set). For control transfers with a single packet data phase, this completes the data phase of the setup transaction, as referenced in Chapter 9 of the *"USB 2.0 Specification"*. If more data needs to be transferred, return to Step 8.
10. To initiate the status phase of the setup transaction, set up a buffer in memory to receive or send the zero length status phase data packet.
11. Initialize the current (even or odd) TX EP0 BD to transfer the status data:
    a) Set the BDT buffer address field to the start address of the data buffer.
    b) Write 8000h to BD0STAT (set UOWN bit, configure DTS to DATA0 and set byte count to 0).
12. Write the Token register with the appropriate IN or OUT token to Endpoint 0, the target device's default control pipe (e.g., write 01h to U1TOK for an OUT token for a `GET DEVICE DESCRIPTOR` command). This initiates an OUT token on the bus, followed by a zero length data packet from the host to the device. When the data packet completes, the BD is updated with the handshake from the device and a Token Complete Interrupt Flag is asserted (the TRNIF flag is set). This completes the status phase of the setup transaction, as described in Chapter 9 of the *"USB 2.0 Specification"*.

Note: Only one control transaction can be performed per frame.

19.5.3 SEND A FULL-SPEED BULK DATA TRANSFER TO A TARGET DEVICE

1. Follow the procedure described in **Section 19.5.1 "Enable Host Mode and Discover a Connected Device"** and **Section 19.5.2 "Complete a Control Transaction to a Connected Device"** to discover and configure a device.
2. To enable transmit and receive transfers with handshaking enabled, write 1Dh to U1EP0. If the target device is a low-speed device, also set the LSPD (U1EP0<7>) bit. If you want the hardware to automatically retry indefinitely if the target device asserts a NAK on the transfer, clear the Retry Disable bit, RETRYDIS (U1EP0<6>).
3. Set up the BD for the current (even or odd) TX EP0 to transfer up to 64 bytes.
4. Set the USB device address of the target device in the address register (U1ADDR<6:0>).
5. Write an OUT token to the desired endpoint to U1TOK. This triggers the module's transmit state machines to begin transmitting the token and the data.
6. Wait for the Token Complete Interrupt Flag, TRNIF. This indicates that the BD has been released back to the microprocessor and the transfer has completed. If the Retry Disable bit (RETRYDIS) is set, the handshake (ACK, NAK, STALL or ERROR (0Fh)) is returned in the BD PID field. If a STALL interrupt occurs, the pending packet must be dequeued and the error condition in the target device cleared. If a detach interrupt occurs (SE0 for more than 2.5 µs), then the target has detached (U1IR<0> is set).
7. Once the Token Complete Interrupt Flag occurs (TRNIF is set), the BD can be examined and the next data packet queued by returning to Step 2.

Note:	USB speed, transceiver and pull-ups should only be configured during the module setup phase. It is not recommended to change these settings while the module is enabled.

19.6 OTG Operation

19.6.1 SESSION REQUEST PROTOCOL (SRP)

An OTG A-device may decide to power down the VBUS supply when it is not using the USB link through the Session Request Protocol (SRP). Software may do this by configuring a GPIO pin to disable an external power transistor, or voltage regulator enable signal, which controls the VBUS supply. When the VBUS supply is powered down, the A-device is said to have ended a USB session.

An OTG A-device or embedded host may repower the VBUS supply at any time (initiate a new session). An OTG B-device may also request that the OTG A-device repower the VBUS supply (initiate a new session). This is accomplished via Session Request Protocol (SRP).

Prior to requesting a new session, the B-device must first check that the previous session has definitely ended. To do this, the B-device must check for two conditions:

1. VBUS supply is below the session valid voltage.
2. Both D+ and D- have been low for at least 2 ms.

The B-device will be notified of Condition 1 by the SESENDIF (U1OTGIR<2>) interrupt. Software will have to manually check for Condition 2.

Note:	When the A-device powers down the VBUS supply, the B-device must disconnect its pull-up resistor from power. If the device is self-powered, it can do this by clearing DPPULUP (U1OTGCON<7>) and DMPULUP (U1OTGCON<6>).

The B-device may aid in achieving Condition 1 by discharging the VBUS supply through a resistor. Software may do this by setting VBUSDIS (U1OTGCON<0>).

After these initial conditions are met, the B-device may begin requesting the new session. The B-device begins by pulsing the D+ data line. Software should do this by setting DPPULUP (U1OTGCON<7>). The data line should be held high for 5 to 10 ms.

The B-device then proceeds by pulsing the VBUS supply. Software should do this by setting PUVBUS (U1CNFG2<4>). When an A-device detects SRP signaling (either via the ATTACHIF (U1IR<6>) interrupt or via the SESVDIF (U1OTGIR<3>) interrupt), the A-device must restore the VBUS supply by properly configuring the general purpose I/O port pin controlling the external power source.

The B-device should not monitor the state of the VBUS supply while performing VBUS supply pulsing. When the B-device does detect that the VBUS supply has been restored (via the SESVDIF (U1OTGIR<3>) interrupt), the B-device must reconnect to the USB link by pulling up D+ or D- (via the DPPULUP or DMPULUP bit).

The A-device must complete the SRP by driving USB Reset signaling.

19.6.2 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.

19.6.3 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 19-3), based on the current level of the VBUS voltage.

TABLE 19-3: 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

19.7 USB OTG Module Registers

There are a total of 37 memory mapped registers associated with the USB OTG module. They can be divided into four general categories:

- USB OTG Module Control (12)
- USB Interrupt (7)
- USB Endpoint Management (16)
- USB VBUS Power Control (2)

This total does not include the (up to) 128 BD registers in the BDT. Their prototypes, described in Register 19-1 and Register 19-2, are shown separately in **Section 19.2 "USB Buffer Descriptors and the BDT"**.

All USB OTG registers are implemented in the Least Significant Byte (LSB) of the register. Bits in the upper byte are unimplemented and have no function. Note that some registers are instantiated only in Host mode, while other registers have different bit instantiations and functions in Device and Host modes.

The registers described in the following sections are those that have bits with specific control and configuration features. The following registers are used for data or address values only:

- U1BDTP1: Specifies the 256-word page in data RAM used for the BDT; 8-bit value with bit 0 fixed as '0' for boundary alignment.
- U1FRML and U1FRMH: Contain the 11-bit byte counter for the current data frame.

19.7.1 USB OTG MODULE CONTROL REGISTERS

REGISTER 19-3: U1OTGSTAT: USB OTG STATUS REGISTER (HOST MODE ONLY)

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

R-0, HSC	U-0	R-0, HSC	U-0	R-0, HSC	R-0, HSC	U-0	R-0, HSC
ID	—	LSTATE	—	SESVD	SESEND	—	VBUSVD
bit 7							bit 0

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

bit 15-8 **Unimplemented:** Read as '0'

bit 7 **ID:** ID Pin State Indicator bit
1 = No plug is attached or a Type B cable has been plugged into the USB receptacle
0 = A Type A plug has been plugged into the USB receptacle

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

bit 5 **LSTATE:** Line State Stable Indicator bit
1 = The USB line state (as defined by SE0 and JSTATE) has been stable for the previous 1 ms
0 = The USB line state has not been stable for the previous 1 ms

bit 4 **Unimplemented:** Read as '0'

bit 3 **SESVD:** Session Valid Indicator bit
1 = The VBUS voltage is above VA_SESS_VLD (as defined in the *"USB 2.0 OTG Specification"*) on the A or B-device
0 = The VBUS voltage is below VA_SESS_VLD on the A or B-device

bit 2 **SESEND:** B Session End Indicator bit
1 = The VBUS voltage is below VB_SESS_END (as defined in the *"USB 2.0 OTG Specification"*) on the B-device
0 = The VBUS voltage is above VB_SESS_END on the B-device

bit 1 **Unimplemented:** Read as '0'

bit 0 **VBUSVD:** A VBUS Valid Indicator bit
1 = The VBUS voltage is above VA_VBUS_VLD (as defined in the *"USB 2.0 OTG Specification"*) on the A-device
0 = The VBUS voltage is below VA_VBUS_VLD on the A-device

REGISTER 19-4: U1OTGCON: USB ON-THE-GO CONTROL REGISTER

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

R/W-0	R/W-0	R/W-0	R/W-0	r-0	R/W-0	r-0	R/W-0
DPPULUP	DMPULUP	DPPULDWN(1)	DMPULDWN(1)	r	OTGEN(1)	r	VBUSDIS(1)
bit 7							bit 0

Legend:	r = Reserved 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 15-8 **Unimplemented:** Read as '0'

bit 7 **DPPULUP:** D+ Pull-up Enable bit
1 = D+ data line pull-up resistor is enabled
0 = D+ data line pull-up resistor is disabled

bit 6 **DMPULUP:** D- Pull-up Enable bit
1 = D- data line pull-up resistor is enabled
0 = D- data line pull-up resistor is disabled

bit 5 **DPPULDWN:** D+ Pull-Down Enable bit(1)
1 = D+ data line pull-down resistor is enabled
0 = D+ data line pull-down resistor is disabled

bit 4 **DMPULDWN:** D- Pull-Down Enable bit(1)
1 = D- data line pull-down resistor is enabled
0 = D- data line pull-down resistor is disabled

bit 3 **Reserved:** Maintain as '0'

bit 2 **OTGEN:** OTG Features Enable bit(1)
1 = USB OTG is enabled; all D+/D- pull-up and pull-down bits are enabled
0 = USB OTG is disabled; D+/D- pull-up and pull-down bits are controlled in hardware by the settings of the HOSTEN and USBEN (U1CON<3,0>) bits

bit 1 **Reserved:** Maintain as '0'

bit 0 **VBUSDIS:** VBUS Discharge Enable bit(1)
1 = VBUS line is discharged through a resistor
0 = VBUS line is not discharged

Note 1: These bits are only used in Host mode; do not use in Device mode.

REGISTER 19-5: U1PWRC: USB POWER CONTROL REGISTER

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

R-x, HSC	U-0	U-0	R/W-0	U-0	U-0	R/W-0, HC	R/W-0
UACTPND	—	—	USLPGRD	—	—	USUSPND	USBPWR
bit 7							bit 0

Legend:	HC = Hardware Clearable bit	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 15-8 **Unimplemented:** Read as '0'

bit 7 **UACTPND:** USB Activity Pending bit
1 = Module should not be suspended at the moment (requires the USLPGRD bit to be set)
0 = Module may be suspended or powered down

bit 6-5 **Unimplemented:** Read as '0'

bit 4 **USLPGRD:** USB Sleep/Suspend Guard bit
1 = Indicates to the USB module that it is about to be suspended or powered down
0 = No suspend

bit 3-2 **Unimplemented:** Read as '0'

bit 1 **USUSPND:** USB Suspend Mode Enable bit
1 = USB OTG module is in Suspend mode; USB clock is gated and the transceiver is placed in a low-power state
0 = Normal USB OTG operation

bit 0 **USBPWR:** USB Operation Enable bit
1 = USB OTG module is enabled
0 = USB OTG module is disabled(1)

Note 1: Do not clear this bit unless the HOSTEN, USBEN and OTGEN bits (U1CON<3,0> and U1OTGCON<2>) are all cleared.

REGISTER 19-6: U1STAT: USB STATUS REGISTER

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

R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	U-0	U-0
ENDPT3	ENDPT2	ENDPT1	ENDPT0	DIR	PPBI(1)	—	—
bit 7							bit 0

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

bit 15-8 **Unimplemented:** Read as '0'

bit 7-4 **ENDPT<3:0>:** Number of the Last Endpoint Activity bits
(Represents the number of the BDT updated by the last USB transfer.)
1111 = Endpoint 15
1110 = Endpoint 14
.
.
.
0001 = Endpoint 1
0000 = Endpoint 0

bit 3 **DIR:** Last BD Direction Indicator bit
1 = The last transaction was a transmit transfer (TX)
0 = The last transaction was a receive transfer (RX)

bit 2 **PPBI:** Ping-Pong BD Pointer Indicator bit(1)
1 = The last transaction was to the odd BD bank
0 = The last transaction was to the even BD bank

bit 1-0 **Unimplemented:** Read as '0'

Note 1: This bit is only valid for endpoints with available even and odd BD registers.

REGISTER 19-7: U1CON: USB CONTROL REGISTER (DEVICE MODE)

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

U-0	R-x, HSC	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	SE0	PKTDIS	—	HOSTEN	RESUME	PPBRST	USBEN
bit 7							bit 0

Legend:	U = Unimplemented bit, read as '0'		
R = Readable bit	W = Writable bit	HSC = Hardware Settable/Clearable bit	
-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 **SE0:** Live Single-Ended Zero Flag bit
1 = Single-ended zero is active on the USB bus
0 = No single-ended zero is detected

bit 5 **PKTDIS:** Packet Transfer Disable bit
1 = SIE token and packet processing are disabled; automatically set when a SETUP token is received
0 = SIE token and packet processing are enabled

bit 4 **Unimplemented:** Read as '0'

bit 3 **HOSTEN:** Host Mode Enable bit
1 = USB host capability is enabled; pull-downs on D+ and D- are activated in hardware
0 = USB host capability is disabled

bit 2 **RESUME:** Resume Signaling Enable bit
1 = Resume signaling is activated
0 = Resume signaling is disabled

bit 1 **PPBRST:** Ping-Pong Buffers Reset bit
1 = Resets all Ping-Pong Buffer Pointers to the even BD banks
0 = Ping-Pong Buffer Pointers are not reset

bit 0 **USBEN:** USB Module Enable bit
1 = USB module and supporting circuitry are enabled (device attached); D+ pull-up is activated in hardware
0 = USB module and supporting circuitry are disabled (device detached)

REGISTER 19-8: U1CON: USB CONTROL REGISTER (HOST MODE ONLY)

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

R-x, HSC	R-x, HSC	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
JSTATE	SE0	TOKBUSY	USBRST	HOSTEN	RESUME	PPBRST	SOFEN
bit 7							bit 0

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

bit 15-8 **Unimplemented:** Read as '0'

bit 7 **JSTATE:** Live Differential Receiver J-State Flag bit
- 1 = J-state (differential '0' in low speed, differential '1' in full speed) is detected on the USB
- 0 = No J-state is detected

bit 6 **SE0:** Live Single-Ended Zero Flag bit
- 1 = Single-ended zero is active on the USB bus
- 0 = No single-ended zero is detected

bit 5 **TOKBUSY:** Token Busy Status bit
- 1 = Token is being executed by the USB module in On-The-Go state
- 0 = No token is being executed

bit 4 **USBRST:** USB Module Reset bit
- 1 = USB Reset has been generated for a software Reset; application must set this bit for 50 ms, then clear it
- 0 = USB Reset is terminated

bit 3 **HOSTEN:** Host Mode Enable bit
- 1 = USB host capability is enabled; pull-downs on D+ and D- are activated in hardware
- 0 = USB host capability is disabled

bit 2 **RESUME:** Resume Signaling Enable bit
- 1 = Resume signaling is activated; software must set bit for 10 ms and then clear to enable remote wake-up
- 0 = Resume signaling is disabled

bit 1 **PPBRST:** Ping-Pong Buffers Reset bit
- 1 = Resets all Ping-Pong Buffer Pointers to the even BD banks
- 0 = Ping-Pong Buffer Pointers are not reset

bit 0 **SOFEN:** Start-of-Frame Enable bit
- 1 = Start-of-Frame token is sent every one 1 ms
- 0 = Start-of-Frame token is disabled

REGISTER 19-9: U1ADDR: USB ADDRESS REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-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
LSPDEN(1)	ADDR6	ADDR5	ADDR4	ADDR3	ADDR2	ADDR1	ADDR0
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 **Unimplemented:** Read as '0'

bit 7 **LSPDEN:** Low-Speed Enable Indicator bit(1)
1 = USB module operates at low speed
0 = USB module operates at full speed

bit 6-0 **ADDR<6:0>:** USB Device Address bits

Note 1: Host mode only. In Device mode, this bit is unimplemented and read as '0'.

REGISTER 19-10: U1TOK: USB TOKEN REGISTER (HOST MODE ONLY)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-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
PID3	PID2	PID1	PID0	EP3	EP2	EP1	EP0
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 **Unimplemented:** Read as '0'

bit 7-4 **PID<3:0>:** Token Type Identifier bits
1101 = SETUP (TX) token type transaction(1)
1001 = IN (RX) token type transaction(1)
0001 = OUT (TX) token type transaction(1)

bit 3-0 **EP<3:0>:** Token Command Endpoint Address bits
This value must specify a valid endpoint on the attached device.

Note 1: All other combinations are reserved and are not to be used.

REGISTER 19-11: U1SOF: USB OTG START-OF-TOKEN THRESHOLD REGISTER (HOST MODE ONLY)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-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
CNT7	CNT6	CNT5	CNT4	CNT3	CNT2	CNT1	CNT0
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 **Unimplemented:** Read as ‘0’

bit 7-0 **CNT<7:0>:** Start-of-Frame Size bits
Value represents 10 + (packet size of n bytes). For example:
0100 1010 = 64-byte packet
0010 1010 = 32-byte packet
0001 0010 = 8-byte packet

REGISTER 19-12: U1CNFG1: USB CONFIGURATION REGISTER 1

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

R/W-0	R/W-0	U-0	R/W-0	U-0	U-0	R/W-0	R/W-0
UTEYE	UOEMON(1)	—	USBSIDL	—	—	PPB1	PPB0
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 **Unimplemented:** Read as '0'

bit 7 **UTEYE:** USB Eye Pattern Test Enable bit
- 1 = Eye pattern test is enabled
- 0 = Eye pattern test is disabled

bit 6 **UOEMON:** USB $\overline{OE}$ Monitor Enable bit(1)
- 1 = $\overline{OE}$ signal is active; it indicates intervals during which the D+/D- lines are driving
- 0 = $\overline{OE}$ signal is inactive

bit 5 **Unimplemented:** Read as '0'

bit 4 **USBSIDL:** USB OTG Stop in Idle Mode bit
- 1 = Discontinues module operation when the device enters Idle mode
- 0 = Continues module operation in Idle mode

bit 3-2 **Unimplemented:** Read as '0'

bit 1-0 **PPB<1:0>:** Ping-Pong Buffers Configuration bits
- 11 = Even/Odd Ping-Pong Buffers are enabled for Endpoints 1 to 15
- 10 = Even/Odd Ping-Pong Buffers are enabled for all endpoints
- 01 = Even/Odd Ping-Pong Buffers are enabled for OUT Endpoint 0
- 00 = Even/Odd Ping-Pong Buffers are disabled

Note 1: This bit is only active when the UTRDIS bit (U1CNFG2<0>) is set.

REGISTER 19-13: U1CNFG2: USB CONFIGURATION REGISTER 2

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

U-0	U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
—	—	UVCMPSEL	PUVBUS	EXTI2CEN	—	UVCMPDIS(1)	UTRDIS(1)
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 15-6 **Unimplemented:** Read as '0'

bit 5 **UVCMPSEL:** VBUS Comparator External Interface Selection bit
1 = Uses VBUSVLD, **SESSVLD** and **SESSEND** as comparator interface pins
0 = Uses VCMPST1 and VCMPST2 as comparator interface pins

bit 4 **PUVBUS:** VBUS Pull-Up Enable bit
1 = Pull-up on VBUS pin is enabled
0 = Pull-up on VBUS pin is disabled

bit 3 **EXTI2CEN:** I^2C™ Interface for External Module Control Enable bit
1 = External module(s) is controlled via the I^2C™ interface
0 = External module(s) is controlled via the dedicated pins

bit 2 **Unimplemented:** Read as '0'

bit 1 **UVCMPDIS:** USB On-Chip VBUS Comparator Disable bit(1)
1 = On-chip charge VBUS comparator is disabled; digital input status interface is enabled
0 = On-chip charge VBUS comparator is active

bit 0 **UTRDIS:** USB On-Chip Transceiver Disable bit(1)
1 = On-chip transceiver is disabled; digital transceiver interface is enabled
0 = On-chip transceiver is active

Note 1: Never change these bits while the USBPWR bit is set (U1PWRC<0> = 1).

19.7.2 USB INTERRUPT REGISTERS

REGISTER 19-14: U1OTGIR: USB OTG INTERRUPT STATUS REGISTER (HOST MODE ONLY)

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

R/K-0, HS	R/K-0, HS	R/K-0, HS	R/K-0, HS	R/K-0, HS	R/K-0, HS	U-0	R/K-0, HS
IDIF	T1MSECIF	LSTATEIF	ACTVIF	SESVDIF	SESENDIF	—	VBUSVDIF
bit 7							bit 0

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

bit 15-8 **Unimplemented:** Read as '0'

bit 7 **IDIF:** ID State Change Indicator bit
1 = Change in ID state is detected
0 = No ID state change is detected

bit 6 **T1MSECIF:** 1 Millisecond Timer bit
1 = The 1 millisecond timer has expired
0 = The 1 millisecond timer has not expired

bit 5 **LSTATEIF:** Line State Stable Indicator bit
1 = USB line state (as defined by the SE0 and JSTATE bits) has been stable for 1 ms, but different from the last time
0 = USB line state has not been stable for 1 ms

bit 4 **ACTVIF:** Bus Activity Indicator bit
1 = Activity on the D+/D- lines or VBUS is detected
0 = No activity on the D+/D- lines or VBUS is detected

bit 3 **SESVDIF:** Session Valid Change Indicator bit
1 = VBUS has crossed VA_SESS_END (as defined in the *"USB 2.0 OTG Specification"*)(1)
0 = VBUS has not crossed VA_SESS_END

bit 2 **SESENDIF:** B-Device VBUS Change Indicator bit
1 = VBUS change on B-device is detected; VBUS has crossed VB_SESS_END (as defined in the *"USB 2.0 OTG Specification"*)(1)
0 = VBUS has not crossed VA_SESS_END

bit 1 **Unimplemented:** Read as '0'

bit 0 **VBUSVDIF:** A-Device VBUS Change Indicator bit
1 = VBUS change on A-device is detected; VBUS has crossed VA_VBUS_VLD (as defined in the *"USB 2.0 OTG Specification"*)(1)
0 = No VBUS change on A-device is detected

Note 1: VBUS threshold crossings may either be rising or falling.

Note: Individual bits can only be cleared by writing a '1' to the bit position as part of a word write operation on the entire register. Using Boolean instructions or bitwise operations to write to a single bit position will cause all set bits, at the moment of the write, to become cleared.

REGISTER 19-15: U1OTGIE: USB OTG INTERRUPT ENABLE REGISTER (HOST MODE ONLY)

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

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0
IDIE	T1MSECIE	LSTATEIE	ACTVIE	SESVDIE	SESENDIE	—	VBUSVDIE
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 **Unimplemented:** Read as ‘0’

bit 7 **IDIE:** ID Interrupt Enable bit
1 = Interrupt is enabled
0 = Interrupt is disabled

bit 6 **T1MSECIE:** 1 Millisecond Timer Interrupt Enable bit
1 = Interrupt is enabled
0 = Interrupt is disabled

bit 5 **LSTATEIE:** Line State Stable Interrupt Enable bit
1 = Interrupt is enabled
0 = Interrupt is disabled

bit 4 **ACTVIE:** Bus Activity Interrupt Enable bit
1 = Interrupt is enabled
0 = Interrupt is disabled

bit 3 **SESVDIE:** Session Valid Interrupt Enable bit
1 = Interrupt is enabled
0 = Interrupt is disabled

bit 2 **SESENDIE:** B-Device Session End Interrupt Enable bit
1 = Interrupt is enabled
0 = Interrupt is disabled

bit 1 **Unimplemented:** Read as ‘0’

bit 0 **VBUSVDIE:** A-Device VBUS Valid Interrupt Enable bit
1 = Interrupt is enabled
0 = Interrupt is disabled

REGISTER 19-16: U1IR: USB INTERRUPT STATUS REGISTER (DEVICE MODE ONLY)

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

R/K-0, HS	U-0	R/K-0, HS	R/K-0, HS	R/K-0, HS	R/K-0, HS	R/K-0, HS	R/K-0, HS
STALLIF	—	RESUMEIF	IDLEIF	TRNIF	SOFIF	UERRIF	URSTIF
bit 7							bit 0

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

bit 15-8 **Unimplemented:** Read as '0'

bit 7 **STALLIF:** STALL Handshake Interrupt bit
- 1 = A STALL handshake was sent by the peripheral during the handshake phase of the transaction in Device mode
- 0 = A STALL handshake has not been sent

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

bit 5 **RESUMEIF:** Resume Interrupt bit
- 1 = A K-state is observed on the D+ or D- pin for 2.5 μs (differential '1' for low speed, differential '0' for full speed)
- 0 = No K-state is observed

bit 4 **IDLEIF:** Idle Detect Interrupt bit
- 1 = Idle condition is detected (constant Idle state of 3 ms or more)
- 0 = No Idle condition is detected

bit 3 **TRNIF:** Token Processing Complete Interrupt bit
- 1 = Processing of the current token is complete; read the U1STAT register for endpoint information
- 0 = Processing of the current token is not complete; clear the U1STAT register or load the next token from STAT (clearing this bit causes the STAT FIFO to advance)

bit 2 **SOFIF:** Start-of-Frame Token Interrupt bit
- 1 = A Start-of-Frame token is received by the peripheral or the Start-of-Frame threshold is reached by the host
- 0 = No Start-of-Frame token is received or threshold reached

bit 1 **UERRIF**: USB Error Condition Interrupt bit
- 1 = An unmasked error condition has occurred; only error states enabled in the U1EIE register can set this bit
- 0 = No unmasked error condition has occurred

bit 0 **URSTIF:** USB Reset Interrupt bit
- 1 = Valid USB Reset has occurred for at least 2.5 μs; Reset state must be cleared before this bit can be re-asserted
- 0 = No USB Reset has occurred; individual bits can only be cleared by writing a '1' to the bit position as part of a word write operation on the entire register. Using Boolean instructions or bitwise operations to write to a single bit position will cause all set bits, at the moment of the write, to become cleared

Note: Individual bits can only be cleared by writing a '1' to the bit position as part of a word write operation on the entire register. Using Boolean instructions or bitwise operations to write to a single bit position will cause all set bits, at the moment of the write, to become cleared.

REGISTER 19-17: U1IR: USB INTERRUPT STATUS REGISTER (HOST MODE ONLY)

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

R/K-0, HS	R/K-0, HS	R/K-0, HS	R/K-0, HS	R/K-0, HS	R/K-0, HS	R/K-0, HS	R/K-0, HS
STALLIF	ATTACHIF	RESUMEIF	IDLEIF	TRNIF	SOFIF	UERRIF	DETACHIF
bit 7							bit 0

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

bit 15-8 **Unimplemented:** Read as '0'

bit 7 **STALLIF:** STALL Handshake Interrupt bit
- 1 = A STALL handshake was sent by the peripheral device during the handshake phase of the transaction in Device mode
- 0 = A STALL handshake has not been sent

bit 6 **ATTACHIF:** Peripheral Attach Interrupt bit
- 1 = A peripheral attachment has been detected by the module; it is set if the bus state is not SE0 and there has been no bus activity for 2.5 μs
- 0 = No peripheral attachment has been detected

bit 5 **RESUMEIF:** Resume Interrupt bit
- 1 = A K-state is observed on the D+ or D- pin for 2.5 μs (differential '1' for low speed, differential '0' for full speed)
- 0 = No K-state is observed

bit 4 **IDLEIF:** Idle Detect Interrupt bit
- 1 = Idle condition is detected (constant Idle state of 3 ms or more)
- 0 = No Idle condition is detected

bit 3 **TRNIF:** Token Processing Complete Interrupt bit
- 1 = Processing of the current token is complete; read the U1STAT register for endpoint information
- 0 = Processing of the current token is not complete; clear the U1STAT register or load the next token from U1STAT

bit 2 **SOFIF:** Start-of-Frame Token Interrupt bit
- 1 = A Start-of-Frame token is received by the peripheral or the Start-of-Frame threshold is reached by the host
- 0 = No Start-of-Frame token is received or threshold reached

bit 1 **UERRIF:** USB Error Condition Interrupt bit
- 1 = An unmasked error condition has occurred; only error states enabled in the U1EIE register can set this bit
- 0 = No unmasked error condition has occurred

bit 0 **DETACHIF:** Detach Interrupt bit
- 1 = A peripheral detachment has been detected by the module; Reset state must be cleared before this bit can be re-asserted
- 0 = No peripheral detachment is detected. Individual bits can only be cleared by writing a '1' to the bit position as part of a word write operation on the entire register. Using Boolean instructions or bitwise operations to write to a single bit position will cause all set bits, at the moment of the write, to become cleared.

Note: Individual bits can only be cleared by writing a '1' to the bit position as part of a word write operation on the entire register. Using Boolean instructions or bitwise operations to write to a single bit position will cause all set bits, at the moment of the write, to become cleared.

REGISTER 19-18: U1IE: USB INTERRUPT ENABLE REGISTER (ALL USB MODES)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-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
STALLIE	ATTACHIE(1)	RESUMEIE	IDLEIE	TRNIE	SOFIE	UERRIE	URSTIE / DETACHIE
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 **Unimplemented:** Read as '0'

bit 7 **STALLIE:** STALL Handshake Interrupt Enable bit
1 = Interrupt is enabled
0 = Interrupt is disabled

bit 6 **ATTACHIE:** Peripheral Attach Interrupt bit (Host mode only)(1)
1 = Interrupt is enabled
0 = Interrupt is disabled

bit 5 **RESUMEIE:** Resume Interrupt bit
1 = Interrupt is enabled
0 = Interrupt is disabled

bit 4 **IDLEIE:** Idle Detect Interrupt bit
1 = Interrupt is enabled
0 = Interrupt is disabled

bit 3 **TRNIE:** Token Processing Complete Interrupt bit
1 = Interrupt is enabled
0 = Interrupt is disabled

bit 2 **SOFIE:** Start-of-Frame Token Interrupt bit
1 = Interrupt is enabled
0 = Interrupt is disabled

bit 1 **UERRIE:** USB Error Condition Interrupt bit
1 = Interrupt is enabled
0 = Interrupt is disabled

bit 0 **URSTIE or DETACHIE:** USB Reset Interrupt (Device mode) or USB Detach Interrupt (Host mode) Enable bit
1 = Interrupt is enabled
0 = Interrupt is disabled

Note 1: This bit is unimplemented in Device mode, read as '0'.

REGISTER 19-19: U1EIR: USB ERROR INTERRUPT STATUS REGISTER

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

R/K-0, HS	U-0	R/K-0, HS	R/K-0, HS	R/K-0, HS	R/K-0, HS	R/K-0, HS	R/K-0, HS
BTSEF	—	DMAEF	BTOEF	DFN8EF	CRC16EF	CRC5EF / EOFEF	PIDEF
bit 7							bit 0

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

bit 15-8 **Unimplemented:** Read as '0'

bit 7 **BTSEF:** Bit Stuff Error Flag bit
1 = Bit stuff error has been detected
0 = No bit stuff error has been detected

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

bit 5 **DMAEF:** DMA Error Flag bit
1 = A USB DMA error condition is detected; the data size indicated by the BD byte count field is less than the number of received bytes, the received data is truncated
0 = No DMA error

bit 4 **BTOEF:** Bus Turnaround Time-out Error Flag bit
1 = Bus turnaround time-out has occurred
0 = No bus turnaround time-out has occurred

bit 3 **DFN8EF:** Data Field Size Error Flag bit
1 = Data field was not an integral number of bytes
0 = Data field was an integral number of bytes

bit 2 **CRC16EF:** CRC16 Failure Flag bit
1 = CRC16 failed
0 = CRC16 passed

bit 1 For Device mode:
CRC5EF: CRC5 Host Error Flag bit
1 = Token packet is rejected due to CRC5 error
0 = Token packet is accepted (no CRC5 error)
For Host mode:
EOFEF: End-of-Frame (EOF) Error Flag bit
1 = End-of-Frame error has occurred
0 = End-of-Frame interrupt is disabled

bit 0 **PIDEF:** PID Check Failure Flag bit
1 = PID check failed
0 = PID check passed

Note: Individual bits can only be cleared by writing a '1' to the bit position as part of a word write operation on the entire register. Using Boolean instructions or bitwise operations to write to a single bit position will cause all set bits, at the moment of the write, to become cleared.

REGISTER 19-20: U1EIE: USB ERROR INTERRUPT ENABLE REGISTER

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

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BTSEE	—	DMAEE	BTOEE	DFN8EE	CRC16EE	CRC5EE EOFEE	PIDEE
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 **Unimplemented:** Read as '0'

bit 7 **BTSEE:** Bit Stuff Error Interrupt Enable bit
1 = Interrupt is enabled
0 = Interrupt is disabled

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

bit 5 **DMAEE:** DMA Error Interrupt Enable bit
1 = Interrupt is enabled
0 = Interrupt is disabled

bit 4 **BTOEE:** Bus Turnaround Time-out Error Interrupt Enable bit
1 = Interrupt is enabled
0 = Interrupt is disabled

bit 3 **DFN8EE:** Data Field Size Error Interrupt Enable bit
1 = Interrupt is enabled
0 = Interrupt is disabled

bit 2 **CRC16EE:** CRC16 Failure Interrupt Enable bit
1 = Interrupt is enabled
0 = Interrupt is disabled

bit 1 For Device mode:
CRC5EE: CRC5 Host Error Interrupt Enable bit
1 = Interrupt is enabled
0 = Interrupt is disabled
For Host mode:
EOFEE: End-of-Frame (EOF) Error interrupt Enable bit
1 = Interrupt is enabled
0 = Interrupt is disabled

bit 0 **PIDEE:** PID Check Failure Interrupt Enable bit
1 = Interrupt is enabled
0 = Interrupt is disabled

19.7.3 USB ENDPOINT MANAGEMENT REGISTERS

REGISTER 19-21: U1EPn: USB ENDPOINT n CONTROL REGISTERS (n = 0 TO 15)

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

R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
LSPD(1)	RETRYDIS(1)	—	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK
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 **Unimplemented:** Read as '0'

bit 7 **LSPD:** Low-Speed Direct Connection Enable bit (U1EP0 only)(1)
1 = Direct connection to a low-speed device is enabled
0 = Direct connection to a low-speed device is disabled

bit 6 **RETRYDIS:** Retry Disable bit (U1EP0 only)(1)
1 = Retry NAK transactions are disabled
0 = Retry NAK transactions are enabled; retry is done in hardware

bit 5 **Unimplemented:** Read as '0'

bit 4 **EPCONDIS:** Bidirectional Endpoint Control bit
If EPTXEN and EPRXEN = 1:
1 = Disables Endpoint n from control transfers; only TX and RX transfers are allowed
0 = Enables Endpoint n for control (SETUP) transfers; TX and RX transfers are also allowed
For All Other Combinations of EPTXEN and EPRXEN:
This bit is ignored.

bit 3 **EPRXEN:** Endpoint Receive Enable bit
1 = Endpoint n receive is enabled
0 = Endpoint n receive is disabled

bit 2 **EPTXEN:** Endpoint Transmit Enable bit
1 = Endpoint n transmit is enabled
0 = Endpoint n transmit is disabled

bit 1 **EPSTALL:** Endpoint STALL Status bit
1 = Endpoint n was stalled
0 = Endpoint n was not stalled

bit 0 **EPHSHK:** Endpoint Handshake Enable bit
1 = Endpoint handshake is enabled
0 = Endpoint handshake is disabled (typically used for isochronous endpoints)

Note 1: These bits are available only for U1EP0 and only in Host mode. For all other U1EPn registers, these bits are always unimplemented and read as '0'.

20.0 DATA SIGNAL MODULATOR

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"*, **"Data Signal Modulator (DSM)"** (DS39744). The information in this data sheet supersedes the information in the FRM.

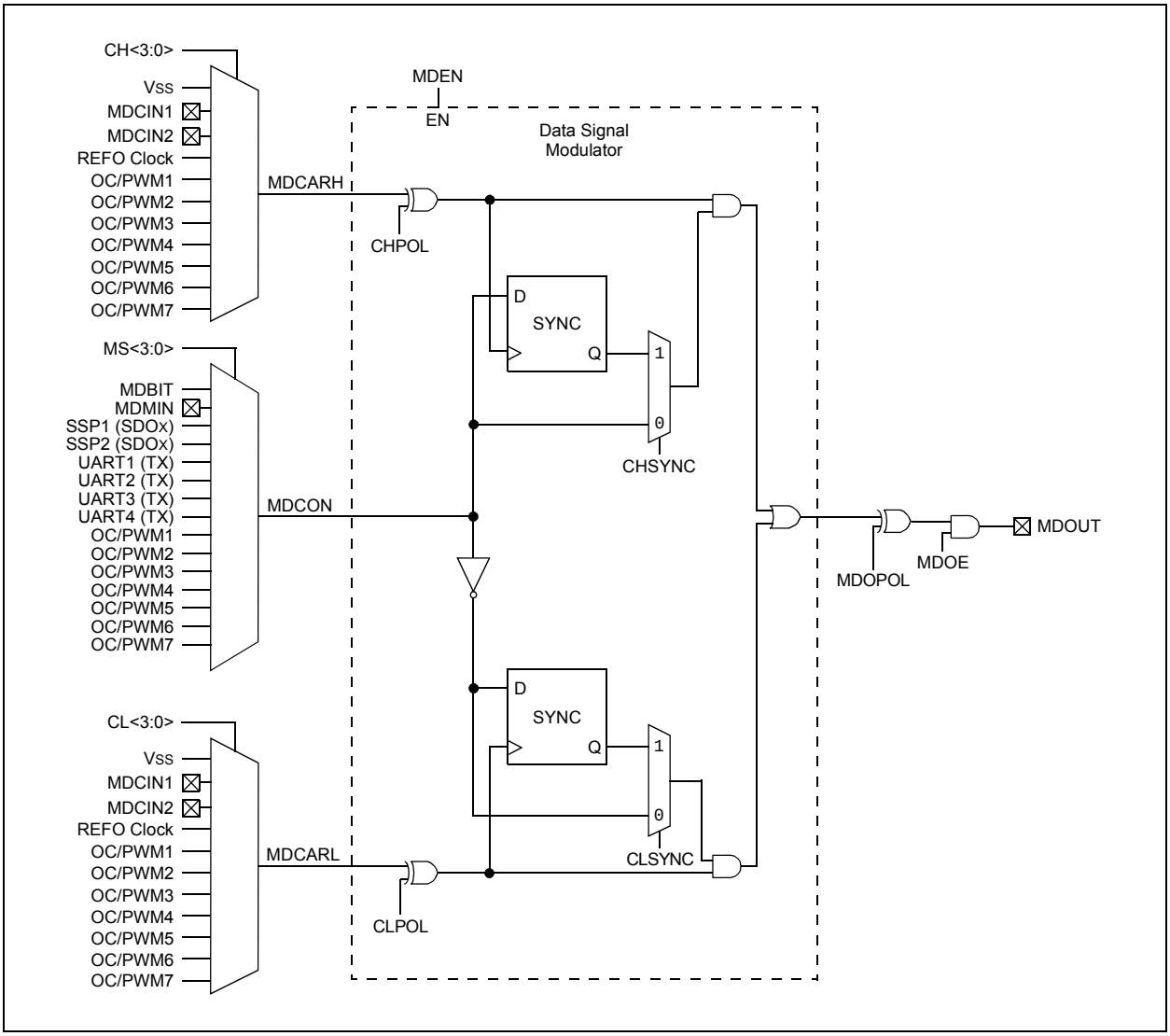
The Data Signal Modulator (DSM) allows the user to mix a digital data stream (the "modulator signal") with a carrier signal to produce a modulated output. Both the carrier and the modulator signals are supplied to the DSM module, either internally from the output of a peripheral, or externally through an input pin.

The modulated output signal is generated by performing a logical AND operation of both the carrier and modulator signals and then it is provided to the MDOUT pin. Using this method, the DSM can generate the following types of key modulation schemes:

- Frequency Shift Keying (FSK)
- Phase-Shift Keying (PSK)
- On-Off Keying (OOK)

Figure 20-1 shows a simplified block diagram of the Data Signal Modulator peripheral.

FIGURE 20-1: SIMPLIFIED BLOCK DIAGRAM OF THE DATA SIGNAL MODULATOR



REGISTER 20-1: MDCON: DATA SIGNAL MODULATOR CONTROL REGISTER

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
MDEN	—	MSIDL	—	—	—	—	—
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	R/W-0
—	MDOE	MDSLR	MDOPOL	—	—	—	MDBIT(1)
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 **MDEN:** DSM Module Enable bit
1 = DSM module is enabled and mixing input signals
0 = DSM module is disabled and has no output

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

bit 13 **MSIDL:** DSM Stop in Idle Mode bit
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode

bit 12-7 **Unimplemented:** Read as '0'

bit 6 **MDOE:** DSM Module Pin Output Enable bit
1 = DSM pin output is enabled
0 = DSM pin output is disabled

bit 5 **MDSLR:** MDOUT Pin Slew Rate Limiting bit
1 = MDOUT pin slew rate limiting is enabled
0 = MDOUT pin slew rate limiting is disabled

bit 4 **MDOPOL:** DSM Output Polarity Select bit
1 = DSM output signal is inverted
0 = DSM output signal is not inverted

bit 3-1 **Unimplemented:** Read as '0'

bit 0 **MDBIT:** DSM Manual Modulation Input bit(1)
1 = Carrier is modulated
0 = Carrier is not modulated

Note 1: The MDBIT must be selected as the modulation source (MDSRC<3:0> = 0000).

REGISTER 20-2: MDSRC: DATA SIGNAL MODULATOR SOURCE CONTROL REGISTER

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

R/W-x	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x
SODIS(1)	—	—	—	MS3(2)	MS2(2)	MS1(2)	MS0(2)
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 **Unimplemented:** Read as '0'

bit 7 **SODIS:** DSM Modulation Source Output Disable bit(1)
1 = Output signal driving the peripheral output pin (selected by MS<3:0>) is disabled
0 = Output signal driving the peripheral output pin (selected by MS<3:0>) is enabled

bit 6-4 **Unimplemented:** Read as '0'

bit 3-0 **MS<3:0>:** DSM Modulation Source Selection bits(2)
1111 = Unimplemented
1110 = Output Compare/PWM Module 7 output
1101 = Output Compare/PWM Module 6 output
1100 = Output Compare/PWM Module 5 output
1011 = Output Compare/PWM Module 4 output
1010 = Output Compare/PWM Module 3 output
1001 = Output Compare/PWM Module 2 output
1000 = Output Compare/PWM Module 1 output
0111 = UART4 TX output
0110 = UART3 TX output
0101 = UART2 TX output
0100 = UART1 TX output
0011 = SPI2 module output (SDO2)
0010 = SPI1 module output (SDO1)
0001 = Input on MDMIN pin
0000 = Manual modulation using MDBIT (MDCON<0>)

Note 1: This bit is only affected by a POR.
2: These bits are not affected by a POR.

REGISTER 20-3: MDCAR: DATA SIGNAL MODULATOR CARRIER CONTROL REGISTER

R/W-x	R/W-x	R/W-x	U-0	R/W-x	R/W-x	R/W-x	R/W-x
CHODIS	CHPOL	CHSYNC	—	CH3(1)	CH2(1)	CH1(1)	CH0(1)
bit 15							bit 8

R/W-0	R/W-x	R/W-x	U-0	R/W-x	R/W-x	R/W-x	R/W-x
CLODIS	CLPOL	CLSYNC	—	CL3(1)	CL2(1)	CL1(1)	CL0(1)
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 **CHODIS:** DSM High Carrier Output Disable bit
1 = Output signal driving the peripheral output pin (selected by CH<3:0>) is disabled
0 = Output signal driving the peripheral output pin is enabled

bit 14 **CHPOL:** DSM High Carrier Polarity Select bit
1 = Selected high carrier signal is inverted
0 = Selected high carrier signal is not inverted

bit 13 **CHSYNC:** DSM High Carrier Synchronization Enable bit
1 = Modulator waits for a falling edge on the high carrier before allowing a switch to the low carrier
0 = Modulator output is not synchronized to the high time carrier signal(1)

bit 12 **Unimplemented:** Read as '0'

bit 11-8 **CH<3:0>:** DSM Data High Carrier Selection bits(1)
1111
... = Reserved
1011
1010 = Output Compare/PWM Module 7 output
1001 = Output Compare/PWM Module 6 output
1000 = Output Compare/PWM Module 5 output
0111 = Output Compare/PWM Module 4 output
0110 = Output Compare/PWM Module 3 output
0101 = Output Compare/PWM Module 2 output
0100 = Output Compare/PWM Module 1 output
0011 = Reference clock (REFO) output
0010 = Input on MDCIN2 pin
0001 = Input on MDCIN1 pin
0000 = Vss

bit 7 **CLODIS:** DSM Low Carrier Output Disable bit
1 = Output signal driving the peripheral output pin (selected by CL<3:0>) is disabled
0 = Output signal driving the peripheral output pin is enabled

bit 6 **CLPOL:** DSM Low Carrier Polarity Select bit
1 = Selected low carrier signal is inverted
0 = Selected low carrier signal is not inverted

bit 5 **CLSYNC:** DSM Low Carrier Synchronization Enable bit
1 = Modulator waits for a falling edge on the low carrier before allowing a switch to the high carrier
0 = Modulator output is not synchronized to the low time carrier signal(1)

bit 4 **Unimplemented:** Read as '0'

bit 3-0 **CL<3:0>:** DSM Data Low Carrier Selection bits(1)
Bit settings are identical to those for CH<3:0>.

Note 1: Narrowed carrier pulse widths or spurs may occur in the signal stream if the carrier is not synchronized.

21.0 ENHANCED PARALLEL MASTER PORT (EPMP)

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*", **"Enhanced Parallel Master Port (EPMP)"** (DS39730). The information in this data sheet supersedes the information in the FRM.

The Enhanced Parallel Master Port (EPMP) module provides a parallel, 4-bit (Master mode only), 8-bit (Master and Slave modes) or 16-bit (Master mode only) data bus interface to communicate with off-chip modules, such as memories, FIFOs, LCD controllers and other microcontrollers. This module can serve as either the master or the slave on the communication bus.

For EPMP Master modes, all external addresses are mapped into the internal Extended Data Space (EDS). This is done by allocating a region of the EDS for each Chip Select (CS) and then assigning each Chip Select to a particular external resource, such as a memory or external controller. This region should not be assigned to another device resource, such as RAM or SFRs. To perform a write or read on an external resource, the CPU simply performs a write or read within the address range assigned for the EPMP.

Key features of the EPMP module are:

- Extended Data Space (EDS) Interface Allows Direct Access from the CPU
- Up to 23 Programmable Address Lines
- Up to 2 Chip Select Lines
- Up to 2 Acknowledgment Lines (one per Chip Select)
- 4-Bit, 8-Bit or 16-Bit Wide Data Bus
- Programmable Strobe Options (per Chip Select):
  - Individual Read and Write Strobes; or
  - Read/$\overline{\text{Write}}$ Strobe with Enable Strobe
- Programmable Address/Data Multiplexing
- Programmable Address Wait States
- Programmable Data Wait States (per Chip Select)
- Programmable Polarity on Control Signals (per Chip Select)
- Legacy Parallel Slave Port (PSP) Support
- Enhanced Parallel Slave Support:
  - Address Support
  - 4-Byte Deep Auto-Incrementing Buffer

21.1 Specific Package Variations

While all PIC24FJ128GC010 family devices implement the EPMP, I/O pin constraints place some limits on 16-Bit Master mode operations in some package types. This is reflected in the number of dedicated Chip Select pins implemented and the number of dedicated address lines that are available. The differences are summarized in Table 21-1. All available EPMP pin functions are summarized in Table 21-2.

For 64-pin devices, the dedicated Chip Select pins (PMCS1 and PMCS2) are not implemented. In addition, only 16 address lines (PMA<15:0>) are available. If required, PMA14 and PMA15 can be remapped to function as PMCS1 and PMCS2, respectively.

The memory space addressable by the device depends on the number of address lines available, as well as the number of Chip Select signals required for the application. Devices with lower pin counts are more affected by Chip Select requirements, as these take away address lines. Table 21-1 shows the maximum addressable range for each pin count.

TABLE 21-1: EPMP FEATURE DIFFERENCES BY DEVICE PIN COUNT

Device	Dedicated Chip Select		Address Lines	Address Range (bytes)		
	CS1	CS2		No CS	1 CS	2 CS
PIC24FJXXXGC006 (64-pin)	—	—	16	64K	32K	16K
PIC24FJXXXGC010 (100/121-pin)	X	X	23	16M		

TABLE 21-2: ENHANCED PARALLEL MASTER PORT PIN DESCRIPTIONS

Pin Name (Alternate Function)	Type	Description
PMA<22:16>	O	Address Bus bits<22:16>
PMA<15> (PMCS2)	O	Address Bus bit 15
	I/O	Data Bus bit 15 (16-bit port with multiplexed addressing)
	O	Chip Select 2 (alternate location)
PMA<14> (PMCS1)	O	Address Bus bit 14
	I/O	Data Bus bit 14 (16-bit port with multiplexed addressing)
	O	Chip Select 1 (alternate location)
PMA<13:8>	O	Address Bus bits<13:8>
	I/O	Data Bus bits<13:8> (16-bit port with multiplexed addressing)
PMA<7:3>	O	Address Bus bits<7:3>
PMA<2> (PMALU)	O	Address Bus bit 2
	O	Address Latch Upper Strobe for Multiplexed Addressing
PMA<1> (PMALH)	I/O	Address Bus bit 1
	O	Address Latch High Strobe for Multiplexed Addressing
PMA<0> (PMALL)	I/O	Address Bus bit 0
	O	Address Latch Low Strobe for Multiplexed Addressing
PMD<15:8>	I/O	Data Bus bits<15:8> (demultiplexed addressing)
PMD<7:4>	I/O	Data Bus bits<7:4>
	O	Address Bus bits<7:4> (4-bit port with 1-phase multiplexed addressing)
PMD<3:0>	I/O	Data Bus bits<3:0>
PMCS1(1)	I/O	Chip Select 1
PMCS2(1)	O	Chip Select 2
PMWR	I/O	Write Strobe(2)
(PMENB)	I/O	Enable Signal(2)
PMRD	I/O	Read Strobe(2)
(PMRD/$\overline{\text{PMWR}}$)	I/O	Read/$\overline{\text{Write}}$ Signal(2)
PMBE1	O	Byte Indicator
PMBE0	O	Nibble or Byte Indicator
PMACK1	I	Acknowledgment Signal 1
PMACK2	I	Acknowledgment Signal 2

Note 1: These pins are implemented in 100/121-pin devices only.

2: Signal function depends on the setting of the MODE<1:0> and SM bits (PMCON1<9:8> and PMCSxCF<8>).

REGISTER 21-1: PMCON1: EPMP CONTROL REGISTER 1

R/W-0	U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
PMPEN	—	PSIDL	ADRMUX1	ADRMUX0	—	MODE1	MODE0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
CSF1	CSF0	ALP	ALMODE	—	BUSKEEP	IRQM1	IRQM0
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 **PMPEN:** Parallel Master Port Enable bit
1 = EPMP is enabled
0 = EPMP is disabled

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

bit 13 **PSIDL:** EPMP Stop in Idle Mode bit
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode

bit 12-11 **ADRMUX<1:0>:** Address/Data Multiplexing Selection bits
11 = Lower address bits are multiplexed with data bits using 3 address phases
10 = Lower address bits are multiplexed with data bits using 2 address phases
01 = Lower address bits are multiplexed with data bits using 1 address phase
00 = Address and data appear on separate pins

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

bit 9-8 **MODE<1:0>:** Parallel Port Mode Select bits
11 = Master mode
10 = Enhanced PSP; pins used are PMRD, PMWR, PMCS<2:1>, PMD<7:0> and PMA<1:0>
01 = Buffered PSP; pins used are PMRD, PMWR, PMCS<2:1> and PMD<7:0>
00 = Legacy PSP; pins used are PMRD, PMWR, PMCS<2:1> and PMD<7:0>

bit 7-6 **CSF<1:0>:** Chip Select Function bits
11 = Reserved
10 = PMA<15> is used for Chip Select 2, PMA<14> is used for Chip Select 1
01 = PMA<15> is used for Chip Select 2, PMCS1 is used for Chip Select 1
00 = PMCS2 is used for Chip Select 2, PMCS1 is used for Chip Select 1

bit 5 **ALP:** Address Latch Polarity bit
1 = Active-high (PMALL, PMALH and PMALU)
0 = Active-low ($\overline{\text{PMALL}}$, $\overline{\text{PMALH}}$ and $\overline{\text{PMALU}}$)

bit 4 **ALMODE:** Address Latch Strobe Mode bit
1 = Enables "smart" address strobes (each address phase is only present if the current access would cause a different address in the latch than the previous address)
0 = Disables "smart" address strobes

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

bit 2 **BUSKEEP:** Bus Keeper bit
1 = Data bus keeps its last value when not actively being driven
0 = Data bus is in a high-impedance state when not actively being driven

REGISTER 21-1: PMCON1: EPMP CONTROL REGISTER 1 (CONTINUED)

bit 1-0 **IRQM<1:0>:** Interrupt Request Mode bits
11 = Interrupt is generated when Read Buffer 3 is read or Write Buffer 3 is written (Buffered PSP mode), or on a read or write operation when PMA<1:0> = 11 (Addressable PSP mode only)
10 = Reserved
01 = Interrupt is generated at the end of a read/write cycle
00 = No interrupt is generated

REGISTER 21-2: PMCON2: EPMP CONTROL REGISTER 2

R-0, HSC	U-0	R/C-0, HS	R/C-0, HS	U-0	U-0	U-0	U-0
BUSY	—	ERROR	TIMEOUT	—	—	—	—
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
RADDR23(1)	RADDR22(1)	RADDR21(1)	RADDR20(1)	RADDR19(1)	RADDR18(1)	RADDR17(1)	RADDR16(1)
bit 7							bit 0

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

bit 15 **BUSY:** Busy bit (Master mode only)
1 = Port is busy
0 = Port is not busy

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

bit 13 **ERROR:** Error bit
1 = Transaction error (illegal transaction was requested)
0 = Transaction completed successfully

bit 12 **TIMEOUT:** Time-out bit
1 = Transaction timed out
0 = Transaction completed successfully

bit 11-8 **Unimplemented:** Read as '0'

bit 7-0 **RADDR<23:16>:** Parallel Master Port Reserved Address Space bits(1)

Note 1: If RADDR<23:16> = 00000000, then the last EDS address for Chip Select 2 will be FFFFFFh.

REGISTER 21-3: PMCON3: EPMP CONTROL REGISTER 3

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
PTWREN	PTRDEN	PTBE1EN	PTBE0EN	—	AWAITM1	AWAITM0	AWAITE
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	PTEN22(1)	PTEN21(1)	PTEN20(1)	PTEN19(1)	PTEN18(1)	PTEN17(1)	PTEN16(1)
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 **PTWREN:** EPMP Write/Enable Strobe Port Enable bit
1 = PMWR/PMENB port is enabled
0 = PMWR/PMENB port is disabled

bit 14 **PTRDEN:** EPMP Read/Write Strobe Port Enable bit
1 = PMRD/$\overline{\text{PMWR}}$ port is enabled
0 = PMRD/$\overline{\text{PMWR}}$ port is disabled

bit 13 **PTBE1EN:** EPMP High Nibble/Byte Enable Port Enable bit
1 = PMBE1 port is enabled
0 = PMBE1 port is disabled

bit 12 **PTBE0EN:** EPMP Low Nibble/Byte Enable Port Enable bit
1 = PMBE0 port is enabled
0 = PMBE0 port is disabled

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

bit 10-9 **AWAITM<1:0>:** Address Latch Strobe Wait States bits
11 = Wait of 3½ TCY
10 = Wait of 2½ TCY
01 = Wait of 1½ TCY
00 = Wait of ½ TCY

bit bit 8 **AWAITE:** Address Hold After Address Latch Strobe Wait States bit
1 = Wait of 1¼ TCY
0 = Wait of ¼ TCY

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

bit 6-0 **PTEN<22:16>:** EPMP Address Port Enable bits(1)
1 = PMA<22:16> function as EPMP address lines
0 = PMA<22:16> function as port I/Os

Note 1: These bits are not available in 64-pin devices (PIC24FJXXXGC006).

REGISTER 21-4: PMCON4: EPMP CONTROL REGISTER 4

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTEN15	PTEN14	PTEN13	PTEN12	PTEN11	PTEN10	PTEN9	PTEN8
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
PTEN7	PTEN6	PTEN5	PTEN4	PTEN3	PTEN2	PTEN1	PTEN0
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 **PTEN15:** PMA15 Port Enable bit
1 = PMA15 functions as either Address Line 15 or Chip Select 2
0 = PMA15 functions as port I/O

bit 14 **PTEN14:** PMA14 Port Enable bit
1 = PMA14 functions as either Address Line 14 or Chip Select 1
0 = PMA14 functions as port I/O

bit 13-3 **PTEN<13:3>:** EPMP Address Port Enable bits
1 = PMA<13:3> function as EPMP address lines
0 = PMA<13:3> function as port I/Os

bit 2-0 **PTEN<2:0>:** PMALU/PMALH/PMALL Strobe Enable bits
1 = PMA<2:0> function as either address lines or address latch strobes
0 = PMA<2:0> function as port I/Os

REGISTER 21-5: PMCSxCF: EPMP CHIP SELECT x CONFIGURATION REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
CSDIS	CSP	CSPTEN	BEP	—	WRSP	RDSP	SM
bit 15							bit 8

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0
ACKP	PTSZ1	PTSZ0	—	—	—	—	—
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 **CSDIS:** Chip Select x Disable bit
1 = Disables the Chip Select x functionality
0 = Enables the Chip Select x functionality

bit 14 **CSP:** Chip Select x Polarity bit
1 = Active-high (PMCSx)
0 = Active-low ($\overline{\text{PMCSx}}$)

bit 13 **CSPTEN:** PMCSx Port Enable bit
1 = PMCSx port is enabled
0 = PMCSx port is disabled

bit 12 **BEP:** Chip Select x Nibble/Byte Enable Polarity bit
1 = Nibble/byte enable is active-high (PMBE0, PMBE1)
0 = Nibble/byte enable is active-low ($\overline{\text{PMBE0}}$, $\overline{\text{PMBE1}}$)

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

bit 10 **WRSP:** Chip Select x Write Strobe Polarity bit
For Slave modes and Master mode when SM = 0:
1 = Write strobe is active-high (PMWR)
0 = Write strobe is active-low ($\overline{\text{PMWR}}$)
For Master mode when SM = 1:
1 = Enable strobe is active-high (PMENB)
0 = Enable strobe is active-low ($\overline{\text{PMENB}}$)

bit 9 **RDSP:** Chip Select x Read Strobe Polarity bit
For Slave modes and Master mode when SM = 0:
1 = Read strobe is active-high (PMRD)
0 = Read strobe is active-low ($\overline{\text{PMRD}}$)
For Master mode when SM = 1:
1 = Read/write strobe is active-high (PMRD/$\overline{\text{PMWR}}$)
0 = Read/Write strobe is active-low ($\overline{\text{PMRD}}$/PMWR)

bit 8 **SM:** Chip Select x Strobe Mode bit
1 = Read/write and enable strobes (PMRD/$\overline{\text{PMWR}}$ and PMENB)
0 = Read and write strobes (PMRD and PMWR)

bit 7 **ACKP:** Chip Select x Acknowledge Polarity bit
1 = ACK is active-high (PMACK1)
0 = ACK is active-low ($\overline{\text{PMACK1}}$)

bit 6-5 **PTSZ<1:0>:** Chip Select x Port Size bits
11 = Reserved
10 = 16-bit port size (PMD<15:0>)
01 = 4-bit port size (PMD<3:0>)
00 = 8-bit port size (PMD<7:0>)

bit 4-0 **Unimplemented:** Read as '0'

REGISTER 21-6: PMCSxBS: EPMP CHIP SELECT x BASE ADDRESS REGISTER(2)

R/W(1)	R/W(1)	R/W(1)	R/W(1)	R/W(1)	R/W(1)	R/W(1)	R/W(1)
BASE23	BASE22	BASE21	BASE20	BASE19	BASE18	BASE17	BASE16
bit 15							bit 8

R/W(1)	U-0	U-0	U-0	U-0	U-0	U-0	U-0
BASE15	—	—	—	—	—	—	—
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 **BASE<23:15>:** Chip Select x Base Address bits(1)

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

Note 1: The value at POR is 0080h for PMCS1BS and 0880h for PMCS2BS.

2: If the whole PMCS2BS register is written together as 0x0000, then the last EDS address for Chip Select 1 will be FFFFFFh. In this case, Chip Select 2 should not be used. PMCS1BS has no such feature.

REGISTER 21-7: PMCSxMD: EPMP CHIP SELECT x MODE REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
ACKM1	ACKM0	AMWAIT2	AMWAIT1	AMWAIT0	—	—	—
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
DWAITB1	DWAITB0	DWAITM3	DWAITM2	DWAITM1	DWAITM0	DWAITE1	DWAITE0
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 15-14 **ACKM<1:0>:** Chip Select x Acknowledge Mode bits
11 = Reserved
10 = PMACKx is used to determine when a read/write operation is complete
01 = PMACKx is used to determine when a read/write operation is complete with time-out (if DWAITM<3:0> = 0000, the maximum time-out is 255 TCY or else it is DWAITM<3:0> cycles)
00 = PMACKx is not used

bit 13-11 **AMWAIT<2:0>:** Chip Select x Alternate Master Wait States bits
111 = Wait of 10 alternate master cycles
...
001 = Wait of 4 alternate master cycles
000 = Wait of 3 alternate master cycles

bit 10-8 **Unimplemented:** Read as '0'

bit 7-6 **DWAITB<1:0>:** Chip Select x Data Setup Before Read/Write Strobe Wait States bits
11 = Wait of 3¼ TCY
10 = Wait of 2¼ TCY
01 = Wait of 1¼ TCY
00 = Wait of ¼ TCY

bit 5-2 **DWAITM<3:0>:** Chip Select x Data Read/Write Strobe Wait States bits
For Write Operations:
1111 = Wait of 15½ TCY
...
0001 = Wait of 1½ TCY
0000 = Wait of ½ TCY
For Read Operations:
1111 = Wait of 15¾ TCY
...
0001 = Wait of 1¾ TCY
0000 = Wait of ¾ TCY

bit 1-0 **DWAITE<1:0>:** Chip Select x Data Hold After Read/Write Strobe Wait States bits
For Write Operations:
11 = Wait of 3¼ TCY
10 = Wait of 2¼ TCY
01 = Wait of 1¼ TCY
00 = Wait of ¼ TCY
For Read Operations:
11 = Wait of 3 TCY
10 = Wait of 2 TCY
01 = Wait of 1 TCY
00 = Wait of 0 TCY

REGISTER 21-8: PMSTAT: EPMP STATUS REGISTER (SLAVE MODE ONLY)

R-0, HSC	R/W-0, HS	U-0	U-0	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC
IBF	IBOV	—	—	IB3F(1)	IB2F(1)	IB1F(1)	IB0F(1)
bit 15							bit 8

R-1, HSC	R/W-0, HS	U-0	U-0	R-1, HSC	R-1, HSC	R-1, HSC	R-1, HSC
OBE	OBUF	—	—	OB3E	OB2E	OB1E	OB0E
bit 7							bit 0

Legend:	HS = Hardware Settable bit	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 **IBF:** Input Buffer Full Status bit
1 = All writable Input Buffer registers are full
0 = Some or all of the writable Input Buffer registers are empty

bit 14 **IBOV:** Input Buffer Overflow Status bit
1 = A write attempt to a full Input register occurred (must be cleared in software)
0 = No overflow occurred

bit 13-12 **Unimplemented:** Read as '0'

bit 11-8 **IB3F:IB0F:** Input Buffer x Status Full bits(1)
1 = Input buffer contains unread data (reading the buffer will clear this bit)
0 = Input buffer does not contain unread data

bit 7 **OBE:** Output Buffer Empty Status bit
1 = All readable Output Buffer registers are empty
0 = Some or all of the readable Output Buffer registers are full

bit 6 **OBUF:** Output Buffer Underflow Status bit
1 = A read occurred from an empty Output register (must be cleared in software)
0 = No underflow occurred

bit 5-4 **Unimplemented:** Read as '0'

bit 3-0 **OB3E:OB0E:** Output Buffer x Status Empty bit
1 = Output buffer is empty (writing data to the buffer will clear this bit)
0 = Output buffer contains untransmitted data

Note 1: Even though an individual bit represents the byte in the buffer, the bits corresponding to the word (Byte 0 and 1 or Byte 2 and 3) get cleared, even on byte reading.

REGISTER 21-9: PADCFG1: PAD CONFIGURATION CONTROL REGISTER

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

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	PMPTTL
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 15-1 **Unimplemented:** Read as ‘0’

bit 0 **PMPTTL:** EPMP Module TTL Input Buffer Select bit
1 = EPMP module inputs (PMDx, PMCS1) use TTL input buffers
0 = EPMP module inputs use Schmitt Trigger input buffers

NOTES:

22.0 LIQUID CRYSTAL DISPLAY (LCD) CONTROLLER

Note: This data sheet summarizes the features of the PIC24FJ128GC010 family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to the *"PIC24F Family Reference Manual"*, **"Liquid Crystal Display (LCD)"** (DS39740) which is available from the Microchip web site (www.microchip.com).

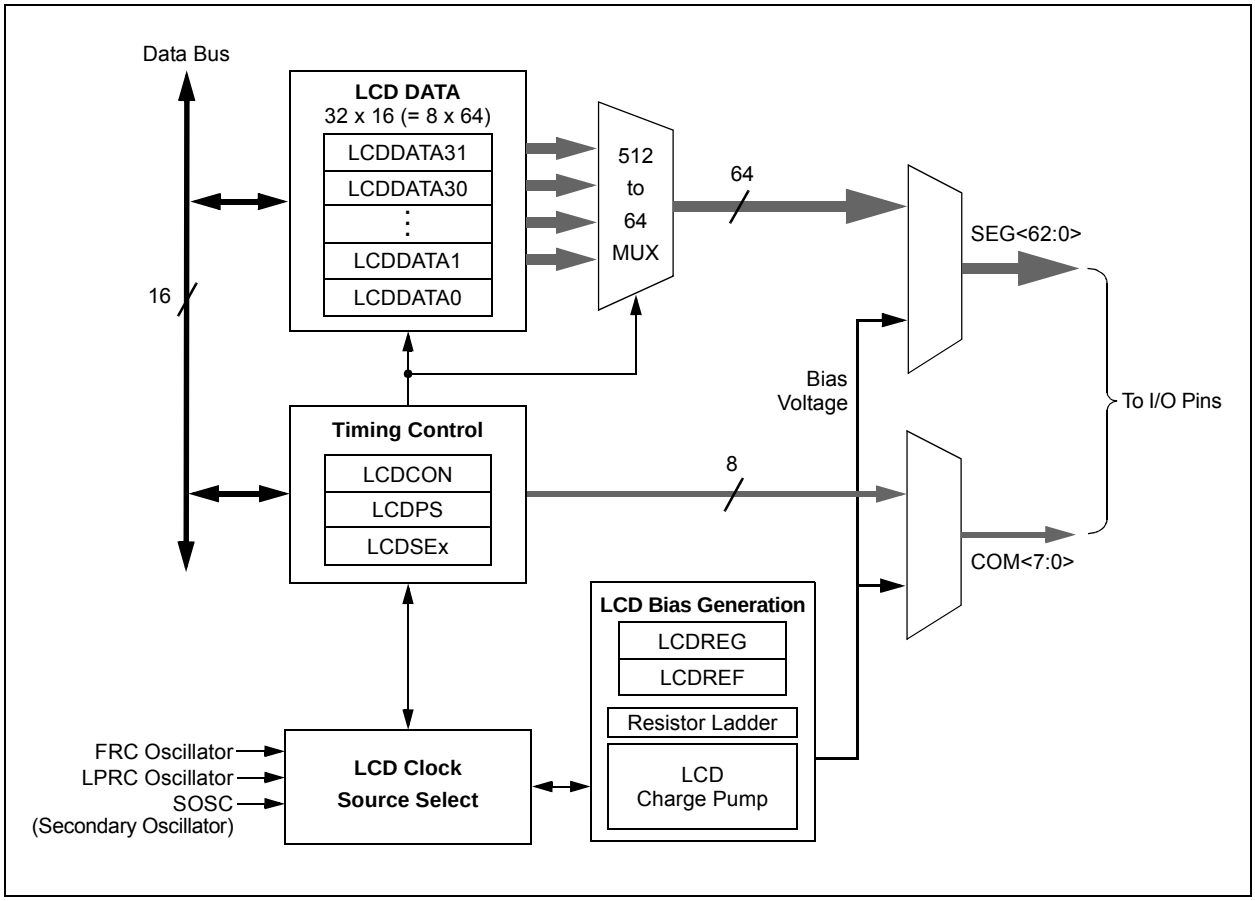
The Liquid Crystal Display (LCD) controller generates the data and timing control required to directly drive a static or multiplexed LCD panel. In 100-pin devices (PIC24FJXXXGC010), the module can drive panels of up to eight commons and up to 59 segments when 5 to 8 commons are used, or up to 63 segments when 1 to 4 commons are used.

The module has these features:

- Direct driving of LCD panel
- Three LCD clock sources with selectable prescaler
- Up to eight commons:
  - Static (one common)
  - 1/2 multiplex (two commons)
  - 1/3 multiplex (three commons)
  - 1/8 multiplex (eight commons)
- Ability to drive from 29 (in 64-pin devices) to 63 (100/121-pin) segments, depending on the Multiplexing mode selected
- Static, 1/2 or 1/3 LCD bias
- On-chip bias generator with dedicated charge pump to support a range of fixed and variable bias options
- Internal resistors for bias voltage generation
- Software contrast control for LCD using internal biasing

A simplified block diagram of the module is shown in Figure 22-1.

FIGURE 22-1: LCD CONTROLLER MODULE BLOCK DIAGRAM



22.1 Registers

The LCD controller has up to 40 registers:

- LCD Control Register (LCDCON)
- LCD Charge Pump Control Register (LCDREG)
- LCD Phase Register (LCDPS)
- LCD Voltage Ladder Control Register (LCDREF)
- Four LCD Segment Enable Registers (LCDSE3:LCDSE0)
- Up to 32 LCD Data Registers (LCDDATA31:LCD-DATA0)

REGISTER 22-1: LCDCON: LCD CONTROL REGISTER

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
LCDEN	—	LCDSIDL	—	—	—	—	—
bit 15							bit 8

U-0	R/W-0	R/C-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	SLPEN	WERR	CS1	CS0	LMUX2	LMUX1	LMUX0
bit 7							bit 0

Legend:	C = 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 **LCDEN:** LCD Driver Enable bit
1 = LCD driver module is enabled
0 = LCD driver module is disabled

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

bit 13 **LCDSIDL:** Stop LCD Drive in CPU Idle Mode Control bit
1 = LCD driver Halts in CPU Idle mode
0 = LCD driver continues to operate in CPU Idle mode

bit 12-7 **Unimplemented:** Read as '0'

bit 6 **SLPEN:** LCD Driver Enable in Sleep Mode bit
1 = LCD driver module is disabled in Sleep mode
0 = LCD driver module is enabled in Sleep mode

bit 5 **WERR:** LCD Write Failed Error bit
1 = LCDDATAx register is written while WA (LCDPS<4>) = 0 (must be cleared in software)
0 = No LCD write error

bit 4-3 **CS<1:0>:** Clock Source Select bits
00 = FRC
01 = LPRC
1x = SOSC

REGISTER 22-1: LCDCON: LCD CONTROL REGISTER (CONTINUED)

bit 2-0 **LMUX<2:0>:** LCD Commons Select bits

LMUX<2:0>	Multiplex	Bias
111	1/8 MUX (COM<7:0>)	1/3
110	1/7 MUX (COM<6:0>)	1/3
101	1/6 MUX (COM<5:0>)	1/3
100	1/5 MUX (COM<4:0>)	1/3
011	1/4 MUX (COM<3:0>)	1/3
010	1/3 MUX (COM<2:0>)	1/2 or 1/3
001	1/2 MUX (COM<1:0>)	1/2 or 1/3
000	Static (COM0)	Static

Note: For multiplex above 4 commons, COM4, COM5, COM6 and COM7 also have segment functionality. Therefore, if the COM is enabled in multiplexing, the segment will not be available on that pin.

REGISTER 22-2: LCDREG: LCD CHARGE PUMP CONTROL REGISTER

RW-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
CPEN	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	RW-1	RW-1	RW-1	RW-1	RW-0	RW-0
—	—	BIAS2	BIAS1	BIAS0	MODE13	CKSEL1	CKSEL0
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 **CPEN:** 3.6V Charge Pump Enable bit
1 = The regulator generates the highest (3.6V) voltage
0 = Highest voltage in the system is supplied externally (AVDD)

bit 14-6 **Unimplemented:** Read as ‘0’

bit 5-3 **BIAS<2:0>:** Regulator Voltage Output Control bits
111 = 3.60V peak (offset on LCDBIAS0 of 0V)
110 = 3.47V peak (offset on LCDBIAS0 of 0.13V)
101 = 3.34V peak (offset on LCDBIAS0 of 0.26V)
100 = 3.21V peak (offset on LCDBIAS0 of 0.39V)
011 = 3.08V peak (offset on LCDBIAS0 of 0.52V)
010 = 2.95V peak (offset on LCDBIAS0 of 0.65V)
001 = 2.82V peak (offset on LCDBIAS0 of 0.78V)
000 = 2.69V peak (offset on LCDBIAS0 of 0.91V)

bit 2 **MODE13:** 1/3 LCD Bias Enable bit
1 = Regulator output supports 1/3 LCD Bias mode
0 = Regulator output supports Static LCD Bias mode

bit 1-0 **CLKSEL<1:0>:** Regulator Clock Select Control bits
11 = SOSC
10 = 8 MHz FRC
01 = LPRC 31 kHz
00 = Disables regulator and floats regulator voltage output

REGISTER 22-3: LCDPS: LCD PHASE REGISTER

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

R/W-0	R/W-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
WFT	BIASMD	LCDA	WA	LP3	LP2	LP1	LP0
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 **Unimplemented:** Read as '0'

bit 7 **WFT:** Waveform Type Select bit
1 = Type-B waveform (phase changes on each frame boundary)
0 = Type-A waveform (phase changes within each common type)

bit 6 **BIASMD:** Bias Mode Select bit
When LMUX<2:0> = 000 or 011 through 111:
0 = Static Bias mode (do not set this bit to '1')
When LMUX<2:0> = 001 or 010:
1 = 1/2 Bias mode
0 = 1/3 Bias mode

bit 5 **LCDA:** LCD Active Status bit
1 = LCD driver module is active
0 = LCD driver module is inactive

bit 4 **WA:** LCD Write Allow Status bit
1 = Write into the LCDDATAx registers is allowed
0 = Write into the LCDDATAx registers is not allowed

bit 3-0 **LP<3:0>:** LCD Prescaler Select bits
1111 = 1:16
1110 = 1:15
1101 = 1:14
1100 = 1:13
1011 = 1:12
1010 = 1:11
1001 = 1:10
1000 = 1:9
0111 = 1:8
0110 = 1:7
0101 = 1:6
0100 = 1:5
0011 = 1:4
0010 = 1:3
0001 = 1:2
0000 = 1:1

REGISTER 22-4: LCDSEx: LCD SEGMENT x ENABLE 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
SE(n+15)(1,2)	SE(n+14)	SE(n+13)	SE(n+12)	SE(n+11)	SE(n+10)	SE(n+9)	SE(n+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
SE(n+7)	SE(n+6)	SE(n+5)	SE(n+4)	SE(n+3)	SE(n+2)	SE(n+1)	SE(n)
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 7-0 **SE(n+15):SE(n)**: Segment Enable bits
For LCDSE0: n = 0
For LCDSE1: n = 16
For LCDSE2: n = 32
For LCDSE3: n = 48(1,2)
1 = Segment function of the pin is enabled, digital I/O is disabled
0 = Segment function of the pin is disabled, digital I/O is enabled

Note 1: SE63 (LCDSE3<15>) is not implemented.
2: For the SEG49 to work correctly, the JTAG needs to be disabled.

REGISTER 22-5: LCDDATAx: LCD DATA x 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
S(n+15)Cy	S(n+14)Cy	S(n+13)Cy	S(n+12)Cy	S(n+11)Cy	S(n+10)Cy	S(n+9)Cy	S(n+8)Cy
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
S(n+7)Cy	S(n+6)Cy	S(n+5)Cy	S(n+4)Cy	S(n+3)Cy	S(n+2)Cy	S(n+1)Cy	S(n)Cy
bit 7							bit 0

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

bit 15-0 **S(n+15)Cy:S(n)Cy**: Pixel On bits
For registers, LCDDATA0 through LCDDATA3: n = (16x), y = 0
For registers, LCDDATA4 through LCDDATA7: n = (16(x – 4)), y = 1
For registers, LCDDATA8 through LCDDATA11: n = (16(x – 8)), y = 2
For registers, LCDDATA12 through LCDDATA15: n = (16(x – 12)), y = 3
For registers, LCDDATA16 through LCDDATA19: n = (16(x – 16)), y = 4
For registers, LCDDATA20 through LCDDATA23: n = (16(x – 20)), y = 5
For registers, LCDDATA24 through LCDDATA27: n = (16(x – 24)), y = 6
For registers, LCDDATA28 through LCDDATA31: n = (16(x – 28)), y = 7
1 = Pixel is on
0 = Pixel is off

TABLE 22-1: LCDDATA REGISTERS AND BITS FOR SEGMENT AND COM COMBINATIONS

COM Lines	Segments			
	0 to 15	16 to 31	32 to 47	48 to 64
0	LCDDATA0 S00C0:S15C0	LCDDATA1 S16C0:S31C0	LCDDATA2 S32C0:S47C0	LCDDATA3 S48C0:S63C0
1	LCDDATA4 S00C1:S15C1	LCDDATA5 S16C1:S31C1	LCDDATA6 S32C1:S47C1	LCDDATA7 S48C1:S63C1
2	LCDDATA8 S00C2:S15C2	LCDDATA9 S16C2:S31C2	LCDDATA10 S32C2:S47C2	LCDDATA11 S48C2:S63C2
3	LCDDATA12 S00C3:S15C3	LCDDATA13 S16C3:S31C3	LCDDATA14 S32C3:S47C3	LCDDATA15 S48C3:S63C3
4	LCDDATA16 S00C4:S15C4	LCDDATA17 S16C4:S31C4	LCDDATA18 S32C4:S47C4	LCDDATA19 S48C4:S59C4
5	LCDDATA20 S00C5:S15C5	LCDDATA21 S16C5:S31C5	LCDDATA22 S32C5:S47C5	LCDDATA23 S48C5:S69C5
6	LCDDATA24 S00C6:S15C6	LCDDATA25 S16C6:S31C6	LCDDATA26 S32C6:S47C6	LCDDATA27 S48C6:S59C6
7	LCDDATA28 S00C7:S15C7	LCDDATA29 S16C7:S31C7	LCDDATA30 S32C7:S47C7	LCDDATA31 S48C7:S59C7

REGISTER 22-6: LCDREF: LCD REFERENCE LADDER CONTROL REGISTER

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
LCDIRE	—	LCDCST2	LCDCST1	LCDCST0	VLCD3PE	VLCD2PE	VLCD1PE
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
LRLAP1	LRLAP0	LRLBP1	LRLBP0	—	LRLAT2	LRLAT1	LRLAT0
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 **LCDIRE**: LCD Internal Reference Enable bit
1 = Internal LCD reference is enabled and connected to the internal contrast control circuit
0 = Internal LCD reference is disabled

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

bit 13-11 **LCDCST<2:0>**: LCD Contrast Control bits
Selects the resistance of the LCD contrast control resistor ladder:
111 = Resistor ladder is at maximum resistance (minimum contrast)
110 = Resistor ladder is at 6/7th of maximum resistance
101 = Resistor ladder is at 5/7th of maximum resistance
100 = Resistor ladder is at 4/7th of maximum resistance
011 = Resistor ladder is at 3/7th of maximum resistance
010 = Resistor ladder is at 2/7th of maximum resistance
001 = Resistor ladder is at 1/7th of maximum resistance
000 = Minimum resistance (maximum contrast); resistor ladder is shorted

bit 10 **VLCD3PE**: LCD Bias 3 Pin Enable bit
1 = Bias 3 level is connected to the external pin, LCDBIAS3
0 = Bias 3 level is internal (internal resistor ladder)

bit 9 **VLCD2PE**: LCD Bias 2 Pin Enable bit
1 = Bias 2 level is connected to the external pin, LCDBIAS2
0 = Bias 2 level is internal (internal resistor ladder)

bit 8 **VLCD1PE**: LCD Bias 1 Pin Enable bit
1 = Bias 1 level is connected to the external pin, LCDBIAS1
0 = Bias 1 level is internal (internal resistor ladder)

bit 7-6 **LRLAP<1:0>**: LCD Reference Ladder A Time Power Control bits
During Time Interval A:
11 = Internal LCD reference ladder is powered in High-Power mode
10 = Internal LCD reference ladder is powered in Medium Power mode
01 = Internal LCD reference ladder is powered in Low-Power mode
00 = Internal LCD reference ladder is powered down and unconnected

bit 5-4 **LRLBP<1:0>**: LCD Reference Ladder B Time Power Control bits
During Time Interval B:
11 = Internal LCD reference ladder is powered in High-Power mode
10 = Internal LCD reference ladder is powered in Medium Power mode
01 = Internal LCD reference ladder is powered in Low-Power mode
00 = Internal LCD reference ladder is powered down and unconnected

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

REGISTER 22-6: LCDREF: LCD REFERENCE LADDER CONTROL REGISTER (CONTINUED)

bit 2-0 **LRLAT<2:0>**: LCD Reference Ladder A Time Interval Control bits

Sets the number of 32 clock counts when the A Time Interval Power mode is active.

For Type-A Waveforms (WFT = 0):

111 = Internal LCD reference ladder is in A Power mode for 7 clocks and B Power mode for 9 clocks
110 = Internal LCD reference ladder is in A Power mode for 6 clocks and B Power mode for 10 clocks
101 = Internal LCD reference ladder is in A Power mode for 5 clocks and B Power mode for 11 clocks
100 = Internal LCD reference ladder is in A Power mode for 4 clocks and B Power mode for 12 clocks
011 = Internal LCD reference ladder is in A Power mode for 3 clocks and B Power mode for 13 clocks
010 = Internal LCD reference ladder is in A Power mode for 2 clocks and B Power mode for 14 clocks
001 = Internal LCD reference ladder is in A Power mode for 1 clock and B Power mode for 15 clocks
000 = Internal LCD reference ladder is always in B Power mode

For Type-B Waveforms (WFT = 1):

111 = Internal LCD reference ladder is in A Power mode for 7 clocks and B Power mode for 25 clocks
110 = Internal LCD reference ladder is in A Power mode for 6 clocks and B Power mode for 26 clocks
101 = Internal LCD reference ladder is in A Power mode for 5 clocks and B Power mode for 27 clocks
100 = Internal LCD reference ladder is in A Power mode for 4 clocks and B Power mode for 28 clocks
011 = Internal LCD reference ladder is in A Power mode for 3 clocks and B Power mode for 29 clocks
010 = Internal LCD reference ladder is in A Power mode for 2 clocks and B Power mode for 30 clocks
001 = Internal LCD reference ladder is in A Power mode for 1 clock and B Power mode for 31 clocks
000 = Internal LCD reference ladder is always in B Power mode

23.0 REAL-TIME CLOCK AND CALENDAR (RTCC)

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 *"PIC24F Family Reference Manual"*, **"Real-Time Clock and Calendar (RTCC)"** (DS39696).

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

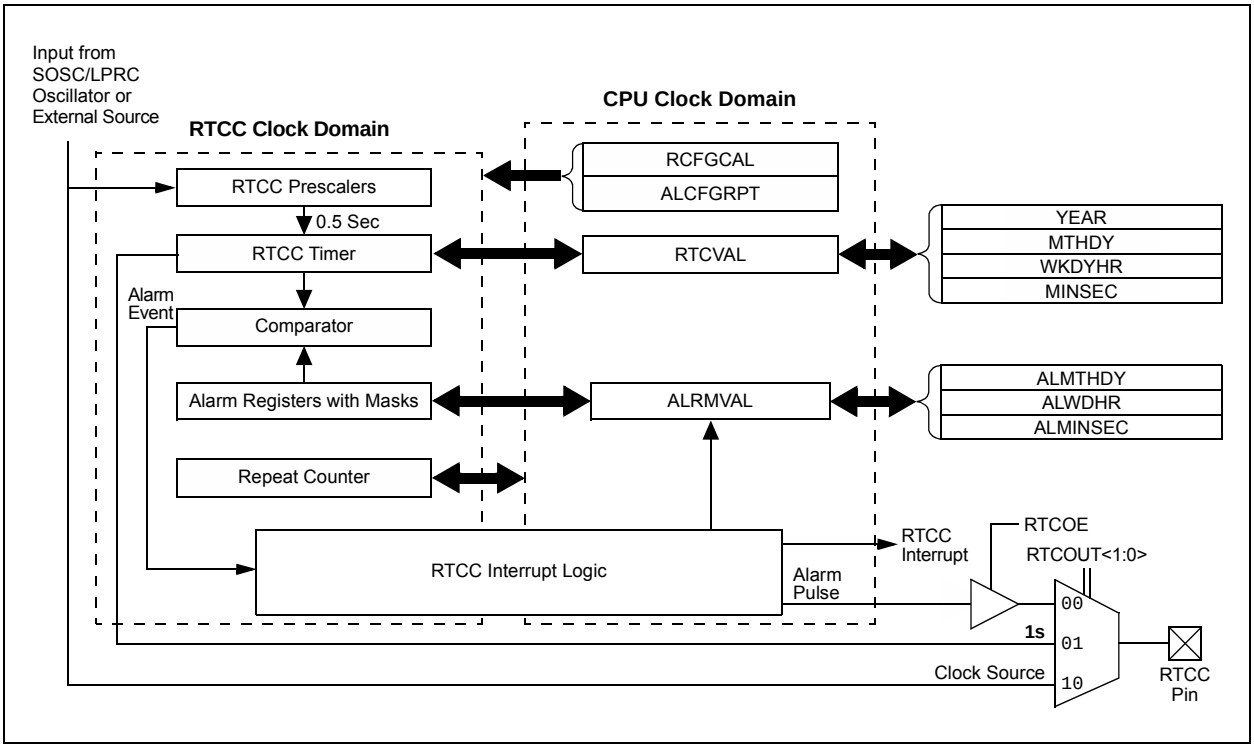
Key features of the RTCC module are:

- Operates in Deep Sleep mode
- 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, one second, 10 seconds, one minute, 10 minutes, one hour, one day, one week, one month or one 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
- Optimized for long-term battery operation
- 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
- Runs from any one of the following:
  - External Real-Time Clock (RTC) of 32.768 kHz
  - Internal 31.25 kHz LPRC clock
  - 50 Hz or 60 Hz external input

23.1 RTCC Source Clock

The user can select between the SOSC crystal oscillator, LPRC internal oscillator or an external 50 Hz/60 Hz power line input as the clock reference for the RTCC module. This gives the user an option to trade off system cost, accuracy and power consumption, based on the overall system needs.

FIGURE 23-1: RTCC BLOCK DIAGRAM



23.2 RTCC Module Registers

The RTCC module registers are organized into three categories:

- RTCC Control Registers
- RTCC Value Registers
- Alarm Value Registers

23.2.1 REGISTER MAPPING

To limit the register interface, the RTCC Timer and Alarm Time registers are accessed through corresponding register pointers. The RTCC Value register window (RTCVALH and RTCVALL) uses the RTCPTR<1:0> bits (RCFGCAL<9:8>) to select the desired Timer register pair (see Table 23-1).

By writing the RTCVALH byte, the RTCC Pointer value, the RTCPTR<1:0> bits decrement by one until they reach '00'. Once they reach '00', the MINUTES and SECONDS value will be accessible through RTCVALH and RTCVALL until the pointer value is manually changed.

TABLE 23-1: RTCVAL REGISTER MAPPING

RTCPTR<1:0>	RTCC Value Register Window	
	RTCVAL<15:8>	RTCVAL<7:0>
00	MINUTES	SECONDS
01	WEEKDAY	HOURS
10	MONTH	DAY
11	—	YEAR

The Alarm Value register window (ALRMVALH and ALRMVALL) uses the ALRMPTR<1:0> bits (ALCFGRPT<9:8>) to select the desired Alarm register pair (see Table 23-2).

By writing the ALRMVALH byte, the Alarm Pointer value, ALRMPTR<1:0> bits, decrement by one until they reach '00'. Once they reach '00', the ALRMMIN and ALRMSEC value will be accessible through ALRMVALH and ALRMVALL until the pointer value is manually changed.

TABLE 23-2: ALRMVAL REGISTER MAPPING

ALRMPTR <1:0>	Alarm Value Register Window	
	ALRMVAL<15:8>	ALRMVAL<7:0>
00	ALRMMIN	ALRMSEC
01	ALRMWD	ALRMHR
10	ALRMMNTH	ALRMDAY
11	—	—

Considering that the 16-bit core does not distinguish between 8-bit and 16-bit read operations, the user must be aware that when reading either the ALRMVALH or ALRMVALL bytes, the ALRMPTR<1:0> value will be decremented. The same applies to the RTCVALH or RTCVALL bytes with the RTCPTR<1:0> bits being decremented.

Note: This only applies to read operations and not write operations.

23.2.2 WRITE LOCK

In order to perform a write to any of the RTCC Timer registers, the RTCWREN bit (RCFGCAL<13>) must be set (see Example 23-1).

Note: To avoid accidental writes to the timer, it is recommended that the RTCWREN bit (RCFGCAL<13>) is kept clear at any other time. For the RTCWREN bit to be set, there is only one instruction cycle time window allowed between the 55h/AA sequence and the setting of RTCWREN; therefore, it is recommended that code follow the procedure in Example 23-1.

23.2.3 SELECTING RTCC CLOCK SOURCE

The clock source for the RTCC module can be selected using the RTCLK<1:0> bits in the RTCPWC register. When the bits are set to '00', the Secondary Oscillator (SOSC) is used as the reference clock and when the bits are '01', LPRC is used as the reference clock. When RTCLK<1:0> = 10 and 11, the external power line (50 Hz and 60 Hz) is used as the clock source.

EXAMPLE 23-1: SETTING THE RTCWREN BIT

```
    asm volatile   (“push w7”);
    asm volatile   (“push w8”);
    asm volatile   (“disi #5”);
    asm volatile   (“mov #0x55, w7”);
    asm volatile   (“mov w7, _NVMKEY”);
    asm volatile   (“mov #0xAA, w8”);
    asm volatile   (“mov w8, _NVMKEY”);
    asm volatile   (“bset _RCFGCAL, #13”);       //set the RTCWREN bit
    asm volatile   (“pop w8”);
    asm volatile   (“pop w7”);
```

23.3 Registers

23.3.1 RTCC CONTROL REGISTERS

REGISTER 23-1: RCFGCAL: RTCC CALIBRATION/CONFIGURATION REGISTER(1)

R/W-0	U-0	R/W-0	R-0, HSC	R-0, HSC	R/W-0	R/W-0	R/W-0
RTCEN(2)	—	RTCWREN	RTCSYNC	HALFSEC(3)	RTCOE	RTCPTR1	RTCPTR0
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
CAL7	CAL6	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0
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 **RTCEN:** RTCC Enable bit(2)
1 = RTCC module is enabled
0 = RTCC module is disabled

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

bit 13 **RTCWREN:** RTCC Value Registers Write Enable bit
1 = RTCVALH and RTCVALL registers can be written to by the user
0 = RTCVALH and RTCVALL registers are locked out from being written to by the user

bit 12 **RTCSYNC:** RTCC Value Registers Read Synchronization bit
1 = RTCVALH, RTCVALL and ALCFGRPT registers can change while reading due to a rollover ripple resulting in an invalid data read. If the register is read twice and results in the same data, the data can be assumed to be valid.
0 = RTCVALH, RTCVALL or ALCFGRPT registers can be read without concern over a rollover ripple

bit 11 **HALFSEC:** Half Second Status bit(3)
1 = Second half period of a second
0 = First half period of a second

bit 10 **RTCOE:** RTCC Output Enable bit
1 = RTCC output is enabled
0 = RTCC output is disabled

bit 9-8 **RTCPTR<1:0>:** RTCC Value Register Window Pointer bits
Points to the corresponding RTCC Value registers when reading the RTCVALH and RTCVALL registers. The RTCPTR<1:0> value decrements on every read or write of RTCVALH until it reaches '00'.
RTCVAL<15:8>:
11 = Reserved
10 = MONTH
01 = WEEKDAY
00 = MINUTES
RTCVAL<7:0>:
11 = YEAR
10 = DAY
01 = HOURS
00 = SECONDS

Note 1: The RCFGCAL register is only affected by a POR.
2: A write to the RTCEN bit is only allowed when RTCWREN = 1.
3: This bit is read-only; it is cleared to '0' on a write to the lower half of the MINSEC register.

REGISTER 23-1: RCFGCAL: RTCC CALIBRATION/CONFIGURATION REGISTER(1) (CONTINUED)

bit 7-0 **CAL<7:0>:** RTC Drift Calibration bits

01111111 = Maximum positive adjustment; adds 127 RTC clock pulses every 15 seconds
.
.
.
01111111 = Minimum positive adjustment; adds 1 RTC clock pulse every 15 seconds
00000000 = No adjustment
11111111 = Minimum negative adjustment; subtracts 1 RTC clock pulse every 15 seconds
.
.
.
10000000 = Maximum negative adjustment; subtracts 128 RTC clock pulses every 15 seconds

Note 1: The RCFGCAL register is only affected by a POR.

2: A write to the RTCEN bit is only allowed when RTCWREN = 1.

3: This bit is read-only; it is cleared to '0' on a write to the lower half of the MINSEC register.

REGISTER 23-2: RTCPWC: RTCC POWER CONTROL REGISTER(1)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PWCEN	PWCPOL	PWCPRE	PWSPRE	RTCLK1(2)	RTCLK0(2)	RTCOUT1	RTCOUT0
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 **PWCEN:** Power Control Enable bit
1 = Power control is enabled
0 = Power control is disabled

bit 14 **PWCPOL:** Power Control Enable bit
1 = Power control is enabled
0 = Power control is disabled

bit 13 **PWCPRE:** Power Control/Stability Prescaler bit
1 = PWC stability window clock is divide-by-2 of source RTCC clock
0 = PWC stability window clock is divide-by-1 of source RTCC clock

bit 12 **PWSPRE:** Power Control Sample Prescaler bit
1 = PWC sample window clock is divide-by-2 of source RTCC clock
0 = PWC sample window clock is divide-by-1 of source RTCC clock

bit 11-10 **RTCLK<1:0>:** RTCC Clock Source Select bits(2)
11 = External power line (60 Hz)
10 = External power line source (50 Hz)
01 = Internal LPRC Oscillator
00 = External Secondary Oscillator (SOSC)

bit 9-8 **RTCOUT<1:0>:** RTCC Output Source Select bits
11 = Power control
10 = RTCC clock
01 = RTCC seconds clock
00 = RTCC alarm pulse

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

Note 1: The RTCPWC register is only affected by a POR.

2: When a new value is written to these register bits, the lower half of the MINSEC register should also be written to properly reset the clock prescalers in the RTCC.

REGISTER 23-3: ALCFGRPT: ALARM CONFIGURATION 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
ALRMEN	CHIME	AMASK3	AMASK2	AMASK1	AMASK0	ALRMPTR1	ALRMPTR0
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
ARPT7	ARPT6	ARPT5	ARPT4	ARPT3	ARPT2	ARPT1	ARPT0
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 **ALRMEN:** Alarm Enable bit

1 = Alarm is enabled (cleared automatically after an alarm event whenever ARPT<7:0> = 00h and CHIME = 0)
0 = Alarm is disabled

bit 14 **CHIME:** Chime Enable bit

1 = Chime is enabled; ARPT<7:0> bits are allowed to roll over from 00h to FFh
0 = Chime is disabled; ARPT<7:0> bits stop once they reach 00h

bit 13-10 **AMASK<3:0>:** Alarm Mask Configuration bits

0000 = Every half second
0001 = Every second
0010 = Every 10 seconds
0011 = Every minute
0100 = Every 10 minutes
0101 = Every hour
0110 = Once a day
0111 = Once a week
1000 = Once a month
1001 = Once a year (except when configured for February 29th, once every 4 years)
101x = Reserved – do not use
11xx = Reserved – do not use

bit 9-8 **ALRMPTR<1:0>:** Alarm Value Register Window Pointer bits

Points to the corresponding Alarm Value registers when reading the ALRMVALH and ALRMVALL registers. The ALRMPTR<1:0> value decrements on every read or write of ALRMVALH until it reaches '00'.

ALRMVAL<15:8>:
00 = ALRMMIN
01 = ALRMWD
10 = ALRMMNTH
11 = Unimplemented

ALRMVAL<7:0>:
00 = ALRMSEC
01 = ALRMHR
10 = ALRMDAY
11 = Unimplemented

bit 7-0 **ARPT<7:0>:** Alarm Repeat Counter Value bits

11111111 = Alarm will repeat 255 more times
.
.
.
00000000 = Alarm will not repeat

The counter decrements on any alarm event; it is prevented from rolling over from 00h to FFh unless CHIME = 1.

23.3.2 RTCVAL REGISTER MAPPINGS

REGISTER 23-4: YEAR: YEAR VALUE REGISTER(1)

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

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
YRTEN3	YRTEN2	YRTEN2	YRTEN1	YRONE3	YRONE2	YRONE1	YRONE0
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 **Unimplemented:** Read as '0'

bit 7-4 **YRTEN<3:0>:** Binary Coded Decimal Value of Year's Tens Digit bits
Contains a value from 0 to 9.

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

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

REGISTER 23-5: MTHDY: MONTH AND DAY VALUE REGISTER(1)

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	MTHTEN0	MTHONE3	MTHONE2	MTHONE1	MTHONE0
bit 15							bit 8

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	DAYTEN1	DAYTEN0	DAYONE3	DAYONE2	DAYONE1	DAYONE0
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 **MTHTEN0:** Binary Coded Decimal Value of Month's Tens Digit bit
Contains a value of '0' or '1'.

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

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

bit 5-4 **DAYTEN<1:0>:** Binary Coded Decimal Value of Day's Tens Digit bits
Contains a value from 0 to 3.

bit 3-0 **DAYONE<3:0>:** Binary Coded Decimal Value of Day'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 23-6: WKDYHR: WEEKDAY AND HOURS VALUE REGISTER(1)

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

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	HRTEN1	HRTEN0	HRONE3	HRONE2	HRONE1	HRONE0
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 **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 23-7: MINSEC: MINUTES AND SECONDS VALUE REGISTER

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

U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	SECTEN2	SECTEN1	SECTEN0	SECONE3	SECONE2	SECONE1	SECONE0
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 **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.

23.3.3 ALRMVAL REGISTER MAPPINGS

REGISTER 23-8: ALMTHDY: ALARM MONTH AND DAY VALUE REGISTER(1)

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	MTHTEN0	MTHONE3	MTHONE2	MTHONE1	MTHONE0
bit 15							bit 8

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	DAYTEN1	DAYTEN0	DAYONE3	DAYONE2	DAYONE1	DAYONE0
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 **MTHTEN0:** Binary Coded Decimal Value of Month's Tens Digit bit
Contains a value of '0' or '1'.

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

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

bit 5-4 **DAYTEN<1:0>:** Binary Coded Decimal Value of Day's Tens Digit bits
Contains a value from 0 to 3.

bit 3-0 **DAYONE<3:0>:** Binary Coded Decimal Value of Day'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 23-9: ALWDHR: ALARM WEEKDAY AND HOURS VALUE REGISTER(1)

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

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	HRTEN1	HRTEN0	HRONE3	HRONE2	HRONE1	HRONE0
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 **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 23-10: ALMINSEC: ALARM MINUTES AND SECONDS VALUE REGISTER

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

U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	SECTEN2	SECTEN1	SECTEN0	SECONE3	SECONE2	SECONE1	SECONE0
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 **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.

23.4 Calibration

The real-time crystal input can be calibrated using the periodic auto-adjust feature. When properly calibrated, the RTCC can provide an error of less than 3 seconds per month. This is accomplished by finding the number of error clock pulses and storing the value into the lower half of the RCFGCAL register. The 8-bit signed value loaded into the lower half of RCFGCAL is multiplied by four and will either be added or subtracted from the RTCC timer, once every minute. Refer to the steps below for RTCC calibration:

1. Using another timer resource on the device, the user must find the error of the 32.768 kHz crystal.
2. Once the error is known, it must be converted to the number of error clock pulses per minute.
3. a) If the oscillator is faster than ideal (negative result from Step 2), the RCFGCAL register value must be negative. This causes the specified number of clock pulses to be subtracted from the timer counter, once every minute.

   b) If the oscillator is slower than ideal (positive result from Step 2), the RCFGCAL register value must be positive. This causes the specified number of clock pulses to be subtracted from the timer counter, once every minute.

EQUATION 23-1:

(Ideal Frequency† – Measured Frequency) * 60 = Clocks per Minute

† Ideal Frequency = 32,768 Hz

Writes to the lower half of the RCFGCAL register should only occur when the timer is turned off, or immediately after the rising edge of the seconds pulse, except when SECONDS = 00, 15, 30 or 45. This is due to the auto-adjust of the RTCC at 15 second intervals.

Note:	It is up to the user to include, in the error value, the initial error of the crystal: drift due to temperature and drift due to crystal aging.

23.5 Alarm

- Configurable from half second to one year
- Enabled using the ALRMEN bit (ALCFGRPT<15>)
- One-time alarm and repeat alarm options available

23.5.1 CONFIGURING THE ALARM

The alarm feature is enabled using the ALRMEN bit. This bit is cleared when an alarm is issued. Writes to ALRMVAL should only take place when ALRMEN = 0.

As shown in Figure 23-2, the interval selection of the alarm is configured through the AMASK<3:0> bits (ALCFGRPT<13:10>). These bits determine which, and how many, digits of the alarm must match the clock value for the alarm to occur.

The alarm can also be configured to repeat based on a preconfigured interval. The amount of times this occurs, once the alarm is enabled, is stored in the ARPT<7:0> bits (ALCFGRPT<7:0>). When the value of the ARPTx bits equals 00h, and the CHIME bit (ALCFGRPT<14>) is cleared, the repeat function is disabled and only a single alarm will occur. The alarm can be repeated, up to 255 times by loading ARPT<7:0> with FFh.

After each alarm is issued, the value of the ARPTx bits is decremented by one. Once the value has reached 00h, the alarm will be issued one last time, after which, the ALRMEN bit will be cleared automatically and the alarm will turn off.

Indefinite repetition of the alarm can occur if the CHIME bit = 1. Instead of the alarm being disabled when the value of the ARPTx bits reaches 00h, it rolls over to FFh and continues counting indefinitely while CHIME is set.

23.5.2 ALARM INTERRUPT

At every alarm event, an interrupt is generated. In addition, an alarm pulse output is provided that operates at half the frequency of the alarm. This output is completely synchronous to the RTCC clock and can be used as a trigger clock to other peripherals.

Note:	Changing any of the registers, other than the RCFGCAL and ALCFGRPT registers, and the CHIME bit, while the alarm is enabled (ALRMEN = 1), can result in a false alarm event leading to a false alarm interrupt. To avoid a false alarm event, the timer and alarm values should only be changed while the alarm is disabled (ALRMEN = 0). It is recommended that the ALCFGRPT register and CHIME bit be changed when RTCSYNC = 0.

FIGURE 23-2: ALARM MASK SETTINGS

Alarm Mask Setting (AMASK<3:0>)	Day of the Week	Month	Day	Hours	Minutes	Seconds
0000 - Every half second 0001 - Every second						
0010 - Every 10 seconds						s
0011 - Every minute						s s
0100 - Every 10 minutes					m	s s
0101 - Every hour					m m	s s
0110 - Every day				h h	m m	s s
0111 - Every week	d			h h	m m	s s
1000 - Every month			d d	h h	m m	s s
1001 - Every year(1)		m m	d d	h h	m m	s s

Note 1: Annually, except when configured for February 29.

23.6 POWER CONTROL

The RTCC includes a power control feature that allows the device to periodically wake-up an external device, wait for the device to be stable before sampling wake-up events from that device, and then shut down the external device. This can be done completely autonomously by the RTCC, without the need to wake from the current lower power mode (Sleep, Deep Sleep, etc.).

To use this feature:

1. Enable the RTCC (RTCEN = 1).
2. Set the PWCEN bit (RTCPWC<15>).
3. Configure the RTCC pin to drive the PWC control signal (RTCOE = 1 and RTCOUT<1:0> = 11).

The polarity of the PWC control signal may be chosen using the PWCPOL bit (RTCPWC<14>). An active-low or active-high signal may be used with the appropriate external switch to turn on or off the power to one or more external devices. The active-low setting may also be used in conjunction with an open-drain setting on the RTCC pin, in order to drive the ground pin(s) of the external device directly (with the appropriate external VDD pull-up device), without the need for external switches. Finally, the CHIME bit should be set to enable the PWC periodicity.

23.7 RTCC VBAT OPERATION

The RTCC can operate in VBAT mode when there is a power loss on the VDD pin. The RTCC will continue to operate if the VBAT pin is powered on (it is usually connected to the battery).

Note:	It is recommended to connect the VBAT pin to VDD if the VBAT mode is not used (not connected to the battery).

The VBAT BOR can be enabled/disabled using the VBTBOR bit in the CW2 Configuration register (CW2<14>). If the VBTBOR enable bit is cleared, the VBAT BOR is always disabled and there will be no indication of a VBAT BOR. If the VBTBOR bit is set, the RTCC can receive a Reset and the RTCEN bit will get cleared; it can happen anywhere between 1.95-1.4V (typical).

24.0 32-BIT PROGRAMMABLE CYCLIC REDUNDANCY CHECK (CRC) GENERATOR

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*", **"32-Bit Programmable Cyclic Redundancy Check (CRC)"** (DS39729). The information in this data sheet supersedes the information in the FRM.

The 32-bit programmable CRC generator provides a hardware implemented method of quickly generating checksums for various networking and security applications. It offers the following features:

- User-programmable CRC polynomial equation, up to 32 bits
- Programmable shift direction (little or big-endian)
- Independent data and polynomial lengths
- Configurable interrupt output
- Data FIFO

Figure 24-1 displays a simplified block diagram of the CRC generator. A simple version of the CRC shift engine is displayed in Figure 24-2.

FIGURE 24-1: CRC BLOCK DIAGRAM

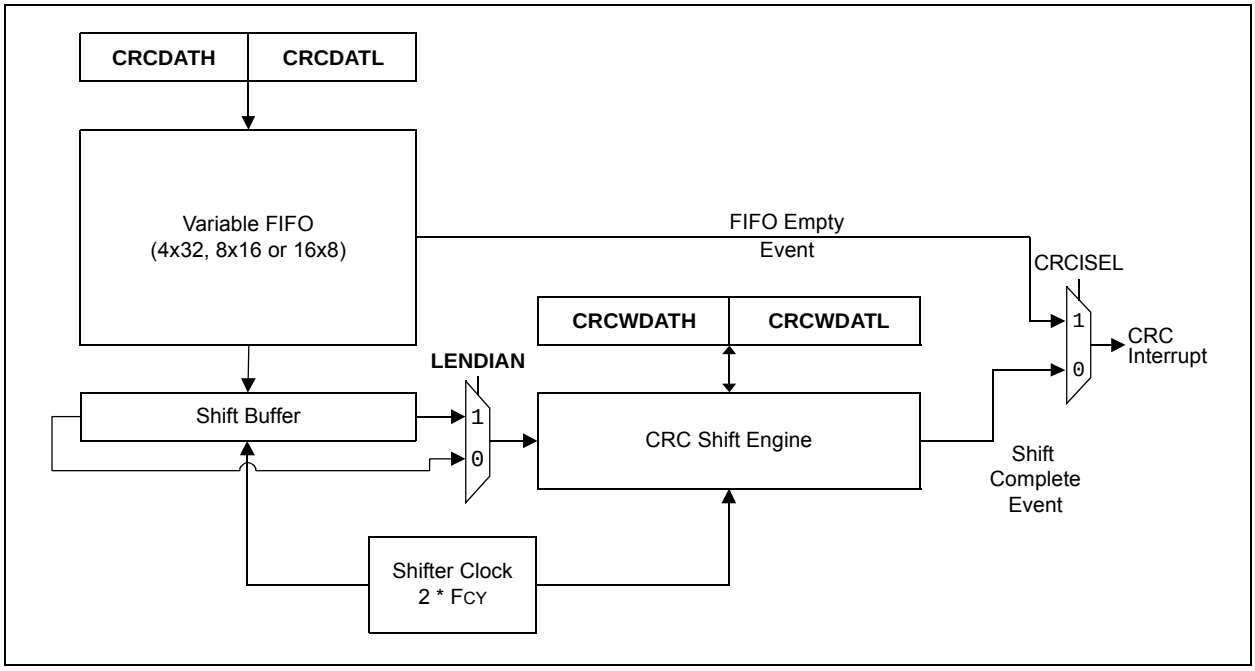
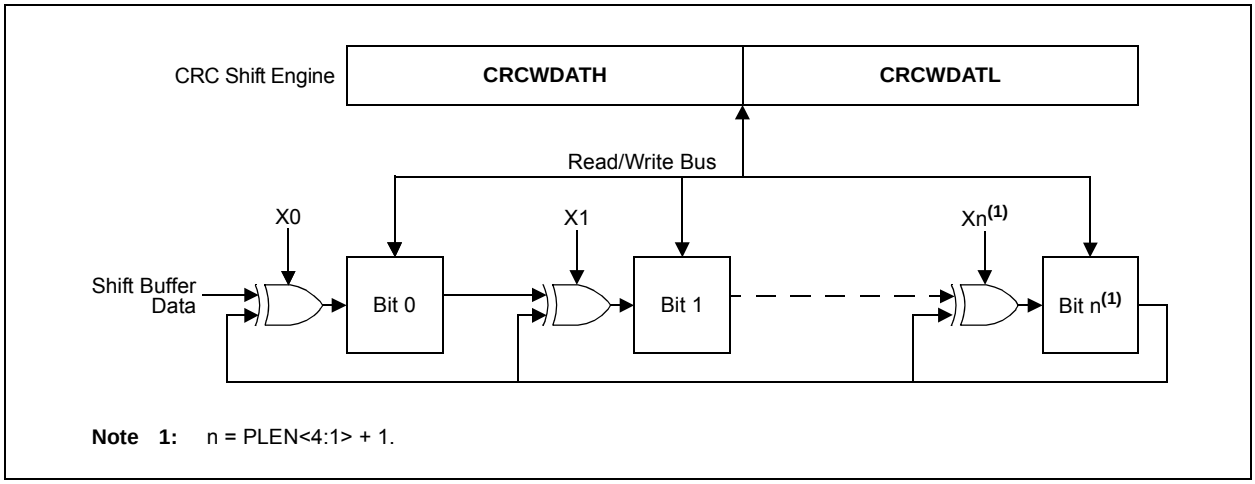


FIGURE 24-2: CRC SHIFT ENGINE DETAIL



Note 1: n = PLEN<4:1> + 1.

24.1 User Interface

24.1.1 POLYNOMIAL INTERFACE

The CRC module can be programmed for CRC polynomials of up to the 32nd 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 24-1: 16-BIT, 32-BIT CRC POLYNOMIALS

$$X16 + X12 + X5 + 1$$

and

$$X32+X26 + X23 + X22 + X16 + X12 + X11 + X10 + X8 + X7 + X5 + X4 + X2 + X + 1$$

To program these polynomials into the CRC generator, set the register bits, as shown in Table 24-1.

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 32nd bit will be used. Therefore, the X<31:1> bits do not have the 32nd bit.

24.1.2 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 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 becomes set. When the VWORDx bits reach zero, the CRCMPT bit 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 24-1: CRC SETUP EXAMPLES FOR 16 AND 32-BIT POLYNOMIALS

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

24.1.3 DATA SHIFT DIRECTION

The LENDIAN bit (CRCCON1<3>) is used to control the shift direction. By default, the CRC will shift data through the engine, MSb first. Setting LENDIAN (= 1) causes the CRC to shift data, LSb first. This setting allows better integration with various communication schemes and removes the overhead of reversing the bit order in software. Note that this only changes the direction the data is shifted into the engine. The result of the CRC calculation will still be a normal CRC result, not a reverse CRC result.

24.1.4 INTERRUPT OPERATION

The module generates an interrupt that is configurable by the user for either of two conditions.

If CRCISEL is '0', an interrupt is generated when the VWORD<4:0> bits make a transition from a value of '1' to '0'. If CRCISEL is '1', an interrupt will be generated after the CRC operation finishes and the module sets the CRCGO bit to '0'. Manually setting CRCGO to '0' will not generate an interrupt. Note that when an interrupt occurs, the CRC calculation would not yet be complete. The module will still need (PLENx + 1)/2 clock cycles after the interrupt is generated until the CRC calculation is finished.

24.1.5 TYPICAL OPERATION

To use the module for a typical CRC calculation:

1. Set the CRCEN bit to enable the module.
2. Configure the module for desired operation:
   a) Program the desired polynomial using the CRCXORL and CRCXORH registers, and the PLEN<4:0> bits.
   b) Configure the data width and shift direction using the DWIDTHx and LENDIAN bits.
   c) Select the desired Interrupt mode using the CRCISEL bit.
3. Preload the FIFO by writing to the CRCDATL and CRCDATH registers until the CRCFUL bit is set or no data is left.
4. Clear old results by writing 00h to CRCWDATL and CRCWDATH. The CRCWDAT registers can also be left unchanged to resume a previously halted calculation.
5. Set the CRCGO bit to start calculation.
6. Write remaining data into the FIFO as space becomes available.
7. When the calculation completes, CRCGO is automatically cleared. An interrupt will be generated if CRCISEL = 1.
8. Read CRCWDATL and CRCWDATH for the result of the calculation.

There are eight registers used to control programmable CRC operation:

- CRCCON1
- CRCCON2
- CRCXORL
- CRCXORH
- CRCDATL
- CRCDATH
- CRCWDATL
- CRCWDATH

The CRCCON1 and CRCCON2 registers (Register 24-1 and Register 24-2) control the operation of the module and configure the various settings.

The CRCXOR registers (Register 24-3 and Register 24-4) select the polynomial terms to be used in the CRC equation. The CRCDAT and CRCWDAT registers are each register pairs that serve as buffers for the double-word input data, and CRC processed output, respectively.

REGISTER 24-1: CRCCON1: CRC CONTROL 1 REGISTER

R/W-0	U-0	R/W-0	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC
CRCEN	—	CSIDL	VWORD4	VWORD3	VWORD2	VWORD1	VWORD0
bit 15							bit 8

R-0, HSC	R-1, HSC	R/W-0	R/W-0, HC	R/W-0	U-0	U-0	U-0
CRCFUL	CRCMPT	CRCISEL	CRCGO	LENDIAN	—	—	—
bit 7							bit 0

Legend:	HC = Hardware Clearable bit	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 **CRCEN:** CRC Enable bit
1 = Enables module
0 = Disables module; all state machines, pointers and CRCWDAT/CRCDATH registers are reset; other SFRs are NOT reset

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

bit 13 **CSIDL:** CRC Stop in Idle Mode bit
1 = Discontinues module operation when the device enters Idle mode
0 = Continues module operation in Idle mode

bit 12-8 **VWORD<4:0>:** Pointer Value bits
Indicates the number of valid words in the FIFO. Has a maximum value of 8 when PLEN<4:0> ≥ 7 or 16 when PLEN<4:0> ≤ 7.

bit 7 **CRCFUL:** CRC FIFO Full bit
1 = FIFO is full
0 = FIFO is not full

bit 6 **CRCMPT:** CRC FIFO Empty bit
1 = FIFO is empty
0 = FIFO is not empty

bit 5 **CRCISEL:** CRC Interrupt Selection bit
1 = Interrupt on FIFO is empty; the final word of data is still shifting through the CRC
0 = Interrupt on shift is complete and results are ready

bit 4 **CRCGO:** Start CRC bit
1 = Starts CRC serial shifter
0 = CRC serial shifter is turned off

bit 3 **LENDIAN:** Data Shift Direction Select bit
1 = Data word is shifted into the CRC, starting with the LSb (little endian)
0 = Data word is shifted into the CRC, starting with the MSb (big endian)

bit 2-0 **Unimplemented:** Read as '0'

REGISTER 24-2: CRCCON2: CRC CONTROL 2 REGISTER

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	DWIDTH4	DWIDTH3	DWIDTH2	DWIDTH1	DWIDTH0
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
—	—	—	PLEN4	PLEN3	PLEN2	PLEN1	PLEN0
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 **DWIDTH<4:0>:** Data Word Width Configuration bits
Configures the width of the data word (Data Word Width – 1).

bit 7-5 **Unimplemented:** Read as ‘0’

bit 4-0 **PLEN<4:0>:** Polynomial Length Configuration bits
Configures the length of the polynomial (Polynomial Length – 1).

REGISTER 24-3: CRCXORL: CRC XOR POLYNOMIAL REGISTER, LOW BYTE

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
X15	X14	X13	X12	X11	X10	X9	X8
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	U-0
X7	X6	X5	X4	X3	X2	X1	—
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 15-1 **X<15:1>:** XOR of Polynomial Term x^n Enable bits

bit 0 **Unimplemented:** Read as ‘0’

REGISTER 24-4: CRCXORH: CRC XOR POLYNOMIAL REGISTER, HIGH BYTE

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
X31	X30	X29	X28	X27	X26	X25	X24
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
X23	X22	X21	X20	X19	X18	X17	X16
bit 7							bit 0

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

bit 15-0 **X<31:16>:** XOR of Polynomial Term x^n Enable bits

25.0 OVERVIEW OF ADVANCED ANALOG FEATURES

The defining feature of PIC24FJ128GC010 family devices is the collection of analog peripherals, designed to extend the range of PIC24F microcontrollers into high-performance analog and mixed-signal applications. All devices include a set of new advanced modules and several existing analog peripherals, plus a common voltage reference for ease of use.

The analog block includes four new modules:

- 12-Bit High-Speed, Pipeline A/D Converter (described in **Section 26.0 "12-Bit High-Speed, Pipeline A/D Converter"**)
- 16-Bit Sigma-Delta A/D (described in **Section 27.0, 16-Bit Sigma-Delta Analog-to-Digital (A/D) Converter**)
- Dual 10-Bit Digital-to-Analog Converters (described in **Section 28.0, 10-Bit Digital-to-Analog Converter (DAC)**)
- Dual Operational Amplifiers (described in **Section 29.0, Dual Operational Amplifier Module**)

It also includes these legacy PIC24F analog modules:

- Triple Comparator module (described in **Section 30.0, Triple Comparator Module**) with independent voltage reference (described in **Section 31.0, Comparator Voltage Reference**)
- CTMU (described in **Section 32.0, Charge Time Measurement Unit (CTMU)**)

A high-level overview of the analog block and its integrating features is shown in Figure 25-1. For a more detailed diagram of each module and an explanation of its operation, please refer to the appropriate chapter.

25.1 Shared Analog Pins

Apart from the reserved differential inputs for the Sigma-Delta A/D Converter, PIC24FJ128GC010 family devices may have up to 50 analog input channels (in 100-pin devices). Because of the number of analog features available on the microcontroller, many of the inputs and outputs of the other advanced analog modules share pins with these channels.

Table 25-1 describes how features are multiplexed. Note that not all of these analog channels and their shared analog peripherals are available on all devices.

25.2 Internal Band Gap References

As an integrating feature, the analog block of PIC24FJ128GC010 family devices includes a common internal voltage reference source. This band gap provides several functions:

- A single, configurable internal reference source (BGBUF0) for all on-chip analog consumers
- Two additional and independently programmable band gap sources that can provide buffered internal references (BGBUF1 and BGBUF2) to external pins
- Independent configurability of all sources in Idle, Sleep and other low-power modes, allowing for flexibility in power consumption

The reference sources are controlled by three registers: BUFCON0 for the internal reference (Register 25-1), and BUFCON1 and BUFCON2 (Register 25-2) for the buffered references.

FIGURE 25-1: ANALOG BLOCK OVERVIEW

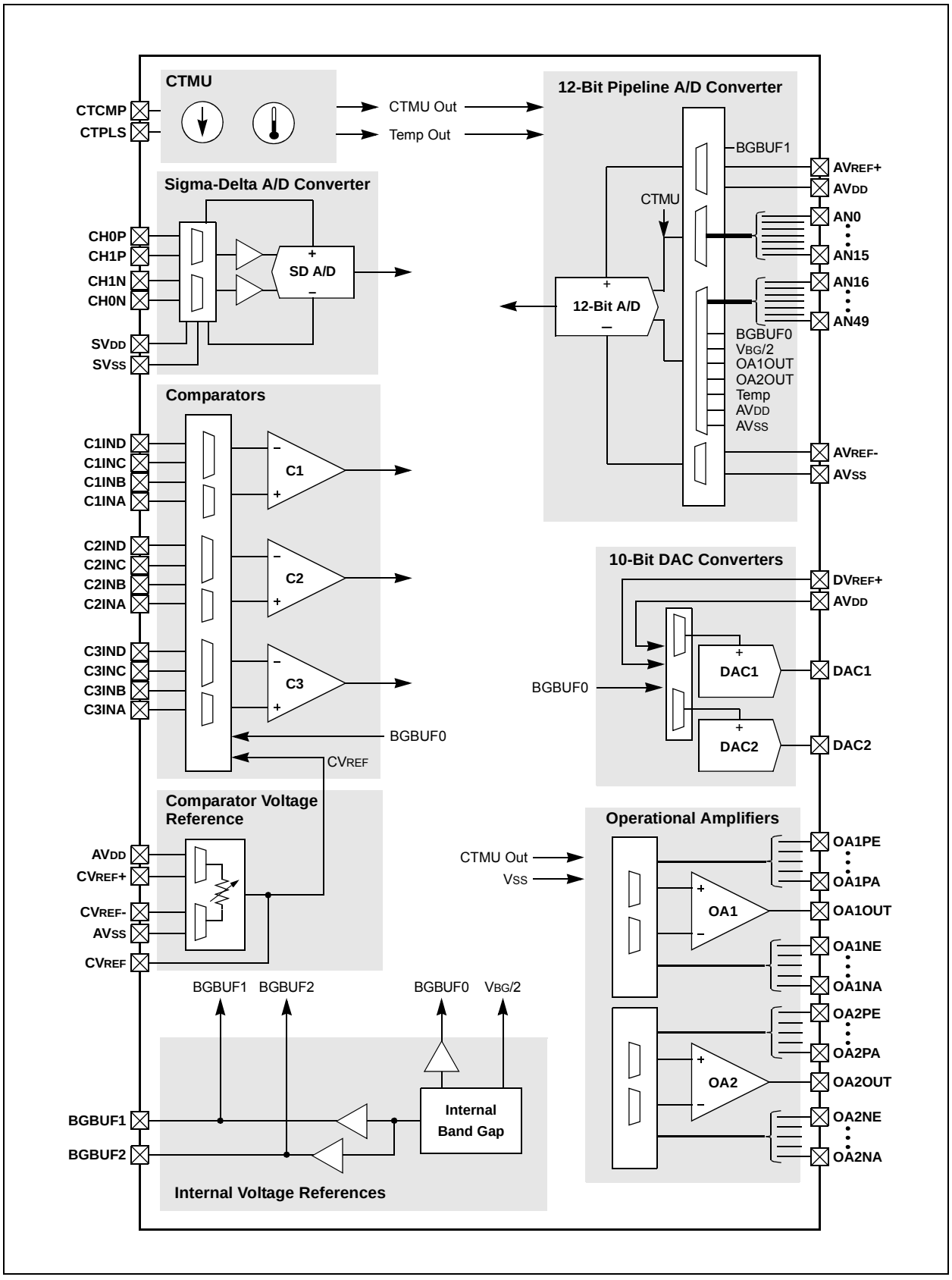


TABLE 25-1: SHARED ANALOG PINS

Analog Input Channel	Op Amp	Comparator	Comparator Reference	DAC	Band Gap	Other Analog
AN0	—	—	CVREF+	DVREF+	BGBUF1	AVREF+
AN1	OA2PB	—	CVREF-	—	—	AVREF-
AN2	OA2NC	C2INB	—	—	—	CTCMP
AN3	OA2OUT	C2INA	—	—	—	—
AN4	OA1NA	C1INB	—	—	—	—
AN5	OA1OUT	C1INA	—	—	—	—
AN6	OA1PD	—	—	—	—	—
AN9	OA1NC	—	—	—	—	—
AN10	OA2PC	—	CVREF	—	—	—
AN11	OA2ND	—	—	—	—	—
AN13	OA2PD	—	—	DAC2	—	—
AN14	OA2NE	—	—	—	—	CTPLS
AN17	OA1PB	C1IND	—	—	BGBUF2	—
AN18	OA1NE	C1INC	—	—	—	—
AN19	OA1NC	C2IND	—	—	—	—
AN20	—	C3INA	—	—	—	—
AN25	OA2NB	—	—	—	—	—
AN30	—	—	—	—	—	—
AN34	OA1PC	C3INB	—	—	—	—
AN41	—	C3IND	—	—	—	—
AN42	OA2PA	C3INC	—	—	—	—
AN43	OA2NA	—	—	—	—	—
AN44	OA2PE	—	—	—	—	—
AN47	OA1PE	—	—	—	—	—
AN48	OA1NB	—	—	—	—	—
AN49	OA1PA	C2INC	—	DAC1	—	—

Legend: Shaded cells are analog outputs.

REGISTER 25-1: BUFCON0: INTERNAL VOLTAGE REFERENCE CONTROL REGISTER

R/W-0	U-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
BUFEN	—	BUFSIDL	BUFSLP	—	—	—	—
bit 15							bit 8

U-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	BUFSTBY	—	—	—	—	BUFREF1(1)	BUFREF0(1)
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 **BUFEN:** Enable Buffer VREF Source bit
1 = Band gap and buffer are enabled
0 = Band gap and buffer are disabled

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

bit 13 **BUFSIDL:** Buffer Stop in Idle bit
1 = Buffer is disabled in Idle mode
0 = Buffer works normally in Idle mode

bit 12 **BUFSLP:** Buffer Sleep Enable bit
1 = Buffer is disabled in Sleep mode
0 = Buffer works normally in Sleep mode

bit 11-7 **Unimplemented:** Read as '0'

bit 6 **BUFSTBY:** Buffer Standby Enable bit
1 = Buffer in Low-Power Standby mode (output unknown or weak drive strength; allows quicker start-up than clearing BUFEN)
0 = Buffer output works normally

bit 5-2 **Unimplemented:** Read as '0'

bit 1-0 **BUFREF<1:0>:** Internal Voltage Reference Select bits(1)
11 = Reference output set at 3.072V
10 = Reference output set at 2.560V
01 = Reference output set at 2.048V
00 = Reference output set at 1.2V

Note 1: The BGBUF cannot "boost" the AVDD voltage to a higher level. Therefore, BUFREF<1:0> bits settings higher than the applied AVDD level are considered invalid.

REGISTER 25-2: BUFCONx: BAND GAP BUFFERS 1 AND 2 CONTROL REGISTERS

R/W-0	U-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
BUFEN	—	BUFSIDL	BUFSLP	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
BUFOE	BUFSTBY	—	—	—	—	BUFREF1	BUFREF0
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 **BUFEN:** Enable Buffer VREF Source bit
- 1 = Band gap and buffer are enabled
- 0 = Band gap and buffer are disabled

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

bit 13 **BUFSIDL:** Buffer Stop in Idle bit
- 1 = Buffer is disabled in Idle mode
- 0 = Buffer works normally in Idle mode

bit 12 **BUFSLP:** Buffer Sleep Enable bit
- 1 = Buffer is disabled in Sleep mode
- 0 = Buffer works normally in Sleep mode

bit 11-8 **Unimplemented:** Read as '0'

bit 7 **BUFOE:** Buffer Output Enable bit
- 1 = Buffer voltage is output to the corresponding pin
- 0 = Buffer voltage is not output to the pin

bit 6 **BUFSTBY:** Buffer Standby Enable bit
- 1 = Buffer in Low-Power Standby mode (output unknown or weak drive strength; allows quicker start-up than clearing BUFEN)
- 0 = Buffer output works normally

bit 5-2 **Unimplemented:** Read as '0'

bit 1-0 **BUFREF<1:0>:** Internal Voltage Reference Select bits
- 11 = Reference output set at 3.072V
- 10 = Reference output set at 2.560V
- 01 = Reference output set at 2.048V
- 00 = Reference output set at 1.2V

NOTES:

26.0 12-BIT HIGH-SPEED, PIPELINE A/D CONVERTER

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 Pipeline A/D Converter, refer to the *"PIC24F Family Reference Manual"*, **"12-Bit, High-Speed Pipeline A/D Converter"** (DS30686).

The 12-Bit Pipeline A/D Converter has the following key features:

- Conversion Speeds of up to 10 Msps
- Up to 50 Analog Single-Ended Input Channels or up to 15 Unique Differential Input Channel Pairs
- 12-Bit Conversion Resolution
- Multiple Internal Reference Input Channels
- External Voltage Reference Input Pins
- Automated Threshold Scan and Compare Operation to Pre-Evaluate Conversion Results
- Extended Automated and Fully Programmable Sampling Sequences from up to Four Different Lists
- Conversion Result Accumulation
- Selectable Conversion Trigger Source
- Internal 32-Word, Configurable Conversion Result Buffer
- Eight Options for Results Alignment
- Configurable Interrupt Generation
- Operation During CPU Sleep and Idle modes

The A/D Converter module is a pipelined 12-bit A/D Converter, capable of sampling up to once per A/D clock cycle. Its operation is enhanced with a wide range of automatic sampling options, tighter integration with other analog modules, result accumulation across many samples and a configurable results buffer.

A simplified block diagram for the module is shown in Figure 26-1.

26.1 Basic Operation

To perform a standard A/D conversion:

1. Configure the module:
   a) Configure port pins as analog inputs by setting the appropriate bits in the ANSx registers (see **Section 11.2 "Configuring Analog Port Pins (ANSx)"** for more information).
   b) Configure "global" ADCON1, ADCON2 and ADCON3 control settings, but do not set the ADON bit until all global settings are configured:
      - Configure A/D clock source/rate
      - Select A/D reference sources
      - Configure data formatting
      - Configure other settings
   c) Enable the A/D module by setting the ADON bit (ADCON1<15>).
   d) Wait until the ADREADY bit (ADSTATH<1>) becomes set, indicating the module is finished with internal calibration and initialization.
   e) Configure Sample List 0 settings, controlled by the ADL0CONH and ADL0CONL registers, but do not enable the sample list yet (SLEN):
      - Select the desired sample list interrupt generation settings
      - Select a Data Write mode (ex: write all results to buffer)
      - Configure analog sampling time (SAMC<4:0>)
      - Select a trigger source
      - Specify how many entries are in the sample list (SLSIZE<4:0>)
      - Configure other Sample List 0 specific settings
   f) Initialize the ADTBL0 register (and higher if SLSIZEx > 0) to select the analog channel(s) to be included in Sample List 0.
   g) Configure and enable A/D interrupts (if desired):
      - Clear the AD1IF and SL0IF bits
      - Select an interrupt priority
      - Enable AD1IE
   h) Enable Sample List 0 by setting the SLEN bit (in ADL0CONL<15>).
   i) Generate a trigger event for Sample List 0 (as configured in Step e).
   j) Wait for the SL0IF or top level AD1IF interrupt flag to assert, indicating that the A/D result(s) are now ready.
   k) Read the respective result(s) from the appropriate ADRESn register(s) (as configured based on the BUFORG setting).

26.2 Registers

The Pipeline A/D Converter uses a total of 116 registers. Of these, 75 registers control the module's operations; the remainder are data and result buffers.

Five "global" registers control overall module operation and provide module status:

- ADCON1 through ADCON3 (Register 26-1 through Register 26-3)
- ADSTATH and ADSTATL (Register 26-4 and Register 26-5)

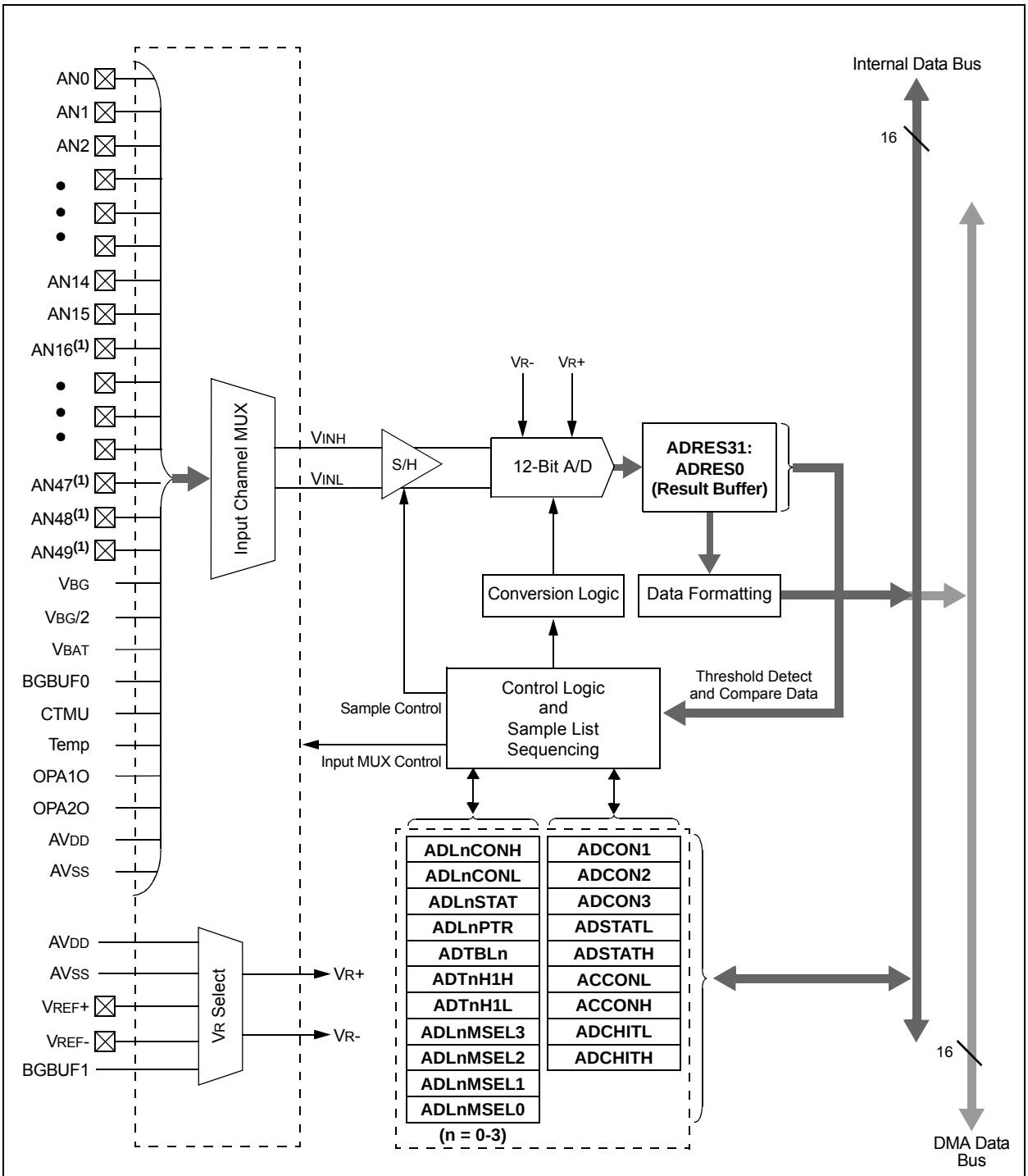
Twelve registers control accumulator and threshold detect operations:

- ACCONH and ACCONL (Register 26-11 and Register 26-12)
- ADCHITH and ADCHITL (Register 26-13 and Register 26-14)
- ADTHnH and ADTHnL (0 through 3) (prototypes, Register 26-15 and Register 26-16)

Sixty-four registers control sample list selection, configuration and execution:

- ADLnCONH and ADLnCONL (0 through 3) (prototypes, Register 26-6 and Register 26-7)
- ADLnSTAT (0 through 3) (prototype, Register 26-8)
- ADLnPTR (0 through 3) (prototype, Register 26-9)
- ADLTLBn (0 through 31) (prototype, Register 26-10)
- ADLnMSEL0 through ADLnMSEL3 (0 through 3) (prototypes, Register 26-17 through Register 26-20)

FIGURE 26-1: 12-BIT PIPELINE A/D CONVERTER BLOCK DIAGRAM



Note 1: Not all external analog inputs are implemented on all devices. See Table 1-3 for a list of implemented channels by pin count.

REGISTER 26-1: ADCON1: A/D CONTROL REGISTER 1

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADON	—	ADSIDL	ADSLP	FORM3	FORM2	FORM1	FORM0
bit 15							bit 8

R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0	R/W-0
PUMPEN	ADCAL(2)	—	—	—	—	—	PWRLVL
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 **ADON:** A/D Module Enable bit
1 = Module is enabled
0 = Module is disabled (registers are still readable and writable)

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

bit 13 **ADSIDL:** A/D Stop in Idle Control bit
1 = Halts when CPU is in Idle mode
0 = Continues to operate in CPU Idle mode

bit 12 **ADSLP:** A/D Suspend in Sleep Control bit
1 = Continues operation in Sleep mode
0 = Ignores triggers and clocks when CPU is in Sleep mode

bit 11-8 **FORM<3:0>:** Data Output Format bits
1xxx = Unimplemented, do not use
0111 = Signed Fractional (sddd dddd dddd 0000)
0110 = Fractional (dddd dddd dddd 0000)
0101 = Signed Integer (ssss sddd dddd dddd)
0100 = Integer (0000 dddd dddd dddd)
0011 = Signed Fractional (sddd dddd dddd 0000)
0010 = Fractional (dddd dddd dddd 0000)
0001 = Signed Integer (ssss sddd dddd dddd)
0000 = Integer, Raw Data (0000 dddd dddd dddd)

bit 7 **PUMPEN:** Analog Channel Switch Charge Pump Enable bit
1 = Charge pump for switches is enabled, reducing switch impedance(1)
0 = Charge pump for switches is disabled

bit 6 **ADCAL:** A/D Internal Analog Calibration bit(2)
1 = Initiates internal analog calibration
0 = No operation

bit 5-1 **Unimplemented:** Read as '0'

bit 0 **PWRLVL:** Power Level Select bit
1 = Full-Power mode; A/D clock rates from 1 MHz to 10 MHz are allowed
0 = Low-Power mode; A/D clock rates from 1 MHz to 2.5 MHz are allowed

Note 1: Use of the channel switch charge pump is recommended when AV_{DD} < 2.5V.

2: When set, ADCAL remains set for at least one T_{AD} and is then automatically cleared by hardware. Manually clearing the bit does not necessarily cancel the calibration routine. Calibration is complete when ADSTATH<1> = 1.

REGISTER 26-2: ADCON2: A/D CONTROL REGISTER 2

R/W-0	R/W-0	U-0	R/W-0	U-0	R/W-0	R/W-1	R/W-1
PVCFG1	PVCFG0	—	NVCFG0	—	BUFORG	r	r
bit 15							bit 8

R/W-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
r	r	—	—	—	—	REFPUMP(1)	r
bit 7							bit 0

Legend:	r = Reserved 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 15-14 **PVCFG<1:0>:** Converter Voltage Reference Configuration for ADREF+ bits
10 = BGBUF1 Internal Reference(2)
01 = External VREF+
00 = AVDD

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

bit 12 **NVCFG0:** Converter Voltage Reference Configuration for ADREF- bit
1 = External VREF-
0 = AVSS

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

bit 10 **BUFORG:** ADRES Result Buffer Organization Control bit
1 = Result buffer is organized as an indexed buffer; ADTBLn conversion result is stored in ADRESn (where n is the same number between 0-31)
0 = Result buffer is organized as a 32 result deep FIFO like buffer; results get stored in the sequential order that they are generated

bit 9-8 **Reserved:** Always write '11' to these bits for normal A/D operation

bit 7-6 **Reserved:** Always write '00' to these bits for normal A/D operation

bit 5-2 **Unimplemented:** Read as '0'

bit 1 **REFPUMP:** A/D Reference Charge Pump Control bit(1)
1 = Reference charge pump is enabled, to optimize internal operation with small references < (0.65 * AVDD)
0 = Reference charge pump is disabled

bit 0 **Reserved:** Always write '0' to this bit for normal A/D operation

Note 1: Never set the REFPUMP bit unless the magnitude of the A/D reference (ex: AVREF+ – AVREF-) is less than (0.65 * AVDD).

2: In order to use the BGBUF1 internal reference for the A/D, firmware must also configure and enable the buffer through the BUFCON1.

REGISTER 26-3: ADCON3: A/D CONTROL REGISTER 3

R/W-0	U-0	U-0	U-0	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC
ADRC(1)	—	—	—	SLEN3	SLEN2	SLEN1	SLEN0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADCS7(2)	ADCS6(2)	ADCS5(2)	ADCS4(2)	ADCS3(2)	ADCS2(2)	ADCS1(2)	ADCS0(2)
bit 7							bit 0

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

bit 15 **ADRC:** A/D Conversion Clock Source (TSRC) bit(1)
1 = Conversion clock derived from FRC (TSRC = TFRC)
0 = Conversion clock derived from system clock (TSRC = TSYS)

bit 14-12 **Unimplemented:** Read as '0'

bit 11 **SLEN3:** A/D Sample List 3 Enable bit
1 = Sampling for this list is enabled; triggers defined by ADL3CONL<12:8> are processed
0 = Sampling for this list is disabled

bit 10 **SLEN2:** A/D Sample List 2 Enable bit
1 = Sampling for this list is enabled; triggers defined by ADL2CONL<12:8> are processed
0 = Sampling for this list is disabled

bit 9 **SLEN1:** A/D Sample List 1 Enable bit
1 = Sampling for this list is enabled; triggers defined by ADL1CONL<12:8> are processed
0 = Sampling for this list is disabled

bit 8 **SLEN0:** A/D Sample List 0 Enable bit
1 = Sampling for this list is enabled; triggers defined by ADL0CONL<12:8> are processed
0 = Sampling for this list is disabled

bit 7-0 **ADCS<7:0>:** A/D Conversion Clock Prescaler bits(2)
TAD = TSRC · (2 · ADCS<7:0>)
Except When ADCS<7:0> = 00h:
TAD = TSRC
Otherwise:
00100001 and higher = Reserved
00100000 = 32 · TSRC
00011111 = 31 · TSRC
...
00000010 = 4 · TSRC
00000001 = 2 · TSRC

Note 1: This bit must be set for Sleep operation.
2: Final A/D clock frequency (1/TAD) must be at or between 1 MHz and 10 MHz.

REGISTER 26-4: ADSTATH: A/D STATUS HIGH REGISTER

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

U-0	U-0	U-0	U-0	U-0	R-0	R-0	R-0
—	—	—	—	—	PUMPST	ADREADY	ADBUSY
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 15-3 **Unimplemented:** Read as ‘0’

bit 2 **PUMPST:** A/D Boost Pump Status bit
1 = The A/D boost pump is active
0 = The A/D boost pump is Idle

bit 1 **ADREADY:** A/D Analog Ready bit
1 = The analog portion of the A/D is warmed up, internally calibrated and ready
0 = The analog portion of the A/D is not ready

bit 0 **ADBUSY:** A/D Busy bit
1 = A/D conversion is in progress
0 = A/D is Idle

REGISTER 26-5: ADSTATL: A/D STATUS LOW REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	SLOV
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	r	ACCIF	SL3IF(1)	SL2IF(1)	SL1IF(1)	SL0IF(1)
bit 7							bit 0

Legend:	r = Reserved 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 15-9 **Unimplemented:** Read as '0'

bit 8 **SLOV:** A/D Sample List Error Event bit
- 1 = A buffer overflow has occurred and data has been lost
- 0 = No buffer overflow has occurred

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

bit 5 **Reserved:** Maintain as '0' for normal A/D interrupt operation

bit 4 **ACCIF:** Accumulator Counter Interrupt Event bit
- 1 = Accumulator counter has counted down to zero
- 0 = Accumulator counter has not reached zero

bit 3 **SL3IF:** A/D Sample List 3 Interrupt Event bit(1)
- 1 = An interrupt event (defined by ADL3CONH<14:13>) has occurred in Sample List 3
- 0 = An interrupt event has not occurred

bit 2 **SL2IF:** A/D Sample List 2 Interrupt Event bit(1)
- 1 = An interrupt event (defined by ADL2CONH<14:13>) has occurred in Sample List 2
- 0 = An interrupt event has not occurred

bit 1 **SL1IF:** A/D Sample List 1 Interrupt Event bit(1)
- 1 = An interrupt event (defined by ADL1CONH<14:13>) has occurred in Sample List 1
- 0 = An interrupt event has not occurred

bit 0 **SL0IF:** A/D Sample List 0 Interrupt Event bit(1)
- 1 = An interrupt event (defined by ADL0CONH<14:13>) has occurred in Sample List 0
- 0 = An interrupt event has not occurred

Note 1: These bits mirror the ADLIF flag bits for the corresponding ADLnSTAT registers. Changes in the ADLIF bit are simultaneously reflected in the SLxIF bits.

REGISTER 26-6: ADLnCONH: A/D SAMPLE LIST n CONTROL HIGH REGISTER (n = 0 to 3)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ASEN	SLINT1	SLINT0	WM1	WM0	CM2	CM1	CM0
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	r	MULCHEN	SAMC4	SAMC3	SAMC2	SAMC1	SAMC0
bit 7							bit 0

Legend:	r = Reserved 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 **ASEN:** A/D Auto-Scan Enable bit
1 = Auto-Scan: Sample and convert all associated inputs sequentially on every trigger event
0 = Sequential Scan: Sample and convert the next associated input on trigger event

bit 14-13 **SLINT<1:0>:** Interrupt Trigger Control bits
When ASEN = 1:
11 = Interrupt after auto-scan completion only if a match occurred
10 = Interrupt after every match
01 = Interrupt after auto-scan completion
00 = No Interrupt
When ASEN = 0:
11 = Reserved
10 = Interrupt after all entries in a sample list have been converted (SLSIZE<4:0> + 1 samples)
01 = Interrupt after every sample
00 = No interrupt

bit 12-11 **WM<1:0>:** Internal Buffer Write Mode bits
11 = Reserved
10 = No conversion results saved (typically for threshold compare only)
01 = Conversion results saved when a match occurs (typically for threshold compare only)
00 = All conversion results saved to the ADRESn register associated with the conversion

bit 10-8 **CM<2:0>:** Threshold Compare Match bits
111 = Reserved
110 = Reserved
101 = Reserved
100 = Outside Window Match: A/D Result < Low Threshold Value or A/D Result >Threshold High Value
011 = Inside Window Match: Low Threshold Value < A/D Result < Threshold High Value
010 = Greater Than Match: A/D Result > Threshold Value.
001 = Less Than Match: A/D Result < Threshold Value
000 = Matching is disabled

bit 7 **CTMEN:** A/D CTMU Current Source Enable bit
1 = CTMU is enabled during sampling and used as a current source driving the selected analog input
0 = CTMU is not used as a current source driving selected analog input

bit 6 **Reserved:** Maintain as '0' for normal operation

bit 5 **MULCHEN:** Multiple Channel Enable bit
1 = Channels 15 to *n* are connected in parallel and scanned together
0 = Channels 15 to *n* in the scan list are sampled one at a time, as defined by the ASEN bit

REGISTER 26-6: ADLnCONH: A/D SAMPLE LIST n CONTROL HIGH REGISTER (n = 0 to 3) (CONTINUED)

bit 4-0 **SAMC<4:0>:** Sample/Hold Capacitor Charge Time (Acquisition Time) bits
11111 = 31 TAD
11110 = 30 TAD
...
00001 = 1 TAD
00000 = 0.5 TAD

REGISTER 26-7: ADLnCONL: A/D SAMPLE LIST n CONTROL LOW REGISTER (n = 0 to 3)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SLEN	SAMP(1)	SLENCLR	SLTSRC4	SLTSRC3	SLTSRC2	SLTSRC1	SLTSRC0
bit 15							bit 8

R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
THSRC	—	—	SLSIZE4	SLSIZE3	SLSIZE2	SLSIZE1	SLSIZE0
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 **SLEN:** A/D Trigger Control Enable bit
1 = Enabled: Selected trigger causes sampling of associated analog inputs
0 = Disabled: Selected trigger does NOT cause sampling of associated analog inputs

bit 14 **SAMP:** A/D Manual Conversion Trigger bit(1)
1 = Prepares to generate a trigger event (no generation yet)
0 = *See SLTSRC<4:0> = 00000, 00001 and 00010 descriptions*

bit 13 **SLENCLR:** A/D Trigger Clear bit
1 = ADTEN is cleared by hardware after a trigger is generated by this sample list
0 = ADTEN is only cleared by software

bit 12-8 **SLTSRC<4:0>:** Trigger Source Select bits
11111
...
10001
= Unimplemented, do not use
10000 = Timer1 A/D Trigger
01111 = Comparator 3
01110 = Comparator 2
01101 = Comparator 1
01100 = Input Capture 4
01011 = Input Capture 1
01010 = Output Compare 3
01001 = Output Compare 2
01000 = Output Compare 1
00111 = Internal Periodic Trigger Event; interval defined by the ADTMRPR register
00110 = CTMU
00101 = Timer2
00100 = Timer1 Sync
00011 = INT0
00010 = Manual Trigger Event: Triggers are generated on every A/D clock when SAMP = 0
00001 = Manual Trigger Event: Triggers are generated on every A/D clock when SAMP = 0 and ACCONH<7> = 1
00000 = Manual Trigger Event: A single trigger is generated when SAMP is manually cleared in firmware, creating a 1 to 0 transition

bit 7 **THSRC:** Threshold List Select bit
1 = Source used for threshold compare is the Sample List Threshold register
0 = Source used for threshold compare is the Buffer register

bit 6-5 **Unimplemented:** Read as '0'

Note 1: Applicable only with Manual Trigger modes (SLTSRC<4:0> = 00010, 00001 or 00000).

REGISTER 26-7: ADLnCONL: A/D SAMPLE LIST n CONTROL LOW REGISTER (n = 0 to 3) (CONTINUED)

bit 4-0 **SLSIZE<4:0>:** Sample List Size Select bits

Number of ADTBLn Registers (+ 1) Associated with this Sample List:

11111 = 32 ADTBLn registers associated with this sample list
11110 = 31 ADTBLn registers associated with this sample list
...
00010 = 3 ADTBLn registers associated with this sample list
00001 = 2 ADTBLn registers associated with this sample list
00000 = 1 ADTBLn register associated with this sample list

Note 1: Applicable only with Manual Trigger modes (SLTSRC<4:0> = 00010, 00001 or 00000).

REGISTER 26-8: ADLnSTAT: A/D SAMPLE LIST n STATUS REGISTER (n = 0 to 3)

R-0	R-0	U-0	U-0	U-0	U-0	U-0	U-0
ADTACT	LBUSY	—	—	—	—	—	—
bit 15							bit 8

R-0	U-0	R/W-0, HS	U-0	U-0	U-0	U-0	U-0
ADTDLY	—	ADLIF(1)	—	—	—	—	—
bit 7							bit 0

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

bit 15 **ADTACT:** A/D Trigger Event Active bit
1 = A trigger event is asserted
0 = A trigger event is not asserted

bit 14 **LBUSY:** Trigger Control Busy bit
1 = The A/D is converting a sample entry associated with this list's trigger
0 = The A/D is not busy with this trigger

bit 13-8 **Unimplemented:** Read as '0'

bit 7 **ADTDLY:** A/D Trigger Delayed Flag bit
1 = This trigger was delayed by a higher priority trigger
0 = This trigger was not delayed by a higher priority trigger

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

bit 5 **ADLIF:** A/D Sample List Interrupt Event Flag bit(1)
1 = An interrupt event (defined by ADLnCONH<14:13>) has occurred in Sample List *n*
0 = No interrupt event has occurred

bit 4-0 **Unimplemented:** Read as '0'

Note 1: ADLIF is mirrored by the corresponding SLxIF flag bit in the ADSTATL register. Setting or clearing this bit simultaneously changes the SLxIF.

REGISTER 26-9: ADLnPTR: A/D SAMPLE LIST n POINTER REGISTER (n = 0 to 3)

U-0	R/W-0, HSC	R/W-0, HSC	R/W-0, HSC	R/W-0, HSC	R/W-0	R/W-0	R/W-0
—	ADNEXT6	ADNEXT5	ADNEXT4	ADNEXT3	ADNEXT2	ADNEXT1	ADNEXT0
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:	U = Unimplemented bit, read as ‘0’		
R = Readable bit	W = Writable bit	HSC = Hardware Settable/Clearable bit	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as ‘0’

bit 14-8 **ADNEXT<6:0>:** Pointer to Next Entry on Sample List to be Converted bits
This value is added to the start of the sample list to determine the ADTBLn register to be used for the next trigger event.

bit 7-0 **Unimplemented:** Read as ‘0’

REGISTER 26-10: ADTBLn: A/D SAMPLE TABLE ENTRY n REGISTER (n = 0 to 31)

R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0	U-0
UCTMU	DIFF	—	—	—	—	—	—
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	ADCH6	ADCH5	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0
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 **UCTMU:** Enable CTMU During Entry Conversion bit
1 = CTMU is enabled during channel conversion for this entry
0 = CTMU is disabled during channel conversion for this entry

bit 14 **DIFF:** Differential Inputs Select bit
1 = Analog inputs are sampled as differential pairs for this entry
0 = Analog inputs are sampled as single-ended for this entry

bit 13-7 **Unimplemented:** Read as ‘0’

bit 6-0 **ADCH<6:0>:** A/D Channel Entry Select bits
See Table 26-1 for a complete description.

TABLE 26-1: CHANNEL ENTRY SELECT VALUES FOR ADCH<6:0>

ADCH<6:0>	Single-Ended (DIFF = 0)		Differential (DIFF = 1)		ADCH<6:0>	Single-Ended (DIFF = 0)		Differential (DIFF = 1)	
	AIN+(1)	AIN-	AIN+(1)	AIN-(1)		AIN+(1)	AIN-	AIN+(1)	AIN-
1111111(2)	VREF-	VREF-	VREF-	VREF-	0100001	AN33	VREF-	AN33	AN14
1111110(3)	VREF-	VREF+	VREF-	VREF+	0100000	AN32	VREF-	AN32	AN14
1111101(4)	VREF+	VREF-	VREF+	VREF-	0011111	AN31	VREF-	AN31	AN14
1111100(2)	VREF+	VREF+	VREF+	VREF+	0011110	AN30	VREF-	AN30	AN14
1110111	CTMU (Time)		CTMU (Time)		0011101	AN29	VREF-	AN29	AN14
1110110	Unimplemented				0011100	AN28	VREF-	AN28	AN14
. . .					0011011	AN27	VREF-	AN27	AN14
0111101					0011010	AN26	VREF-	AN26	AN14
0111100	Reserved				0011001	AN25	VREF-	AN25	AN14
0111011	OPA2	VREF-	OPA2	VREF-	0011000	AN24	VREF-	AN24	AN14
0111010	OPA1	VREF-	OPA1	VREF-	0010111	AN23	VREF-	AN23	AN14
0111001	Reserved				0010110	AN22	VREF-	AN22	AN14
0111000	VBG/2	VREF-	VBG/2	VREF-	0010101	AN21	VREF-	AN21	AN14
0110111	VBAT/2	VREF-	VBAT/2	VREF-	0010100	AN20	VREF-	AN20	AN14
0110110	AVDD	VREF-	AVDD	VREF-	0010011	AN19	VREF-	AN19	AN14
0110101	AVSS	VREF-	AVSS	VREF-	0010010	AN18	VREF-	AN18	AN14
0110100	BGBUF0	VREF-	BGBUF0	VREF-	0010001	AN17	VREF-	AN17	AN14
0110011	Unimplemented				0010000	AN16	VREF-	AN16	AN14
0110010	CTMU (Temp)		CTMU (Temp)		0001111	AN15	VREF-	AN15	AN14
0110001	AN49	VREF-	AN49	AN14	0001110	AN14	VREF-	AN15	AN14
0110000	AN48	VREF-	AN48	AN14	0001101	AN13	VREF-	AN13	AN12
0101111	AN47	VREF-	AN47	AN14	0001100	AN12	VREF-	AN13	AN12
0101110	AN46	VREF-	AN46	AN14	0001011	AN11	VREF-	AN11	AN10
0101101	AN45	VREF-	AN45	AN14	0001010	AN10	VREF-	AN11	AN10
0101100	AN44	VREF-	AN44	AN14	0001001	AN9	VREF-	AN9	AN8
0101011	AN43	VREF-	AN43	AN14	0001000	AN8	VREF-	AN9	AN8
0101010	AN42	VREF-	AN42	AN14	0000111	AN7	VREF-	AN7	AN6
0101001	AN41	VREF-	AN41	AN14	0000110	AN6	VREF-	AN7	AN6
0101000	AN40	VREF-	AN40	AN14	0000101	AN5	VREF-	AN5	AN4
0100111	AN39	VREF-	AN39	AN14	0000100	AN4	VREF-	AN5	AN4
0100110	AN38	VREF-	AN38	AN14	0000011	AN3	VREF-	AN3	AN2
0100101	AN37	VREF-	AN37	AN14	0000010	AN2	VREF-	AN3	AN2
0100100	AN36	VREF-	AN36	AN14	0000001	AN1	VREF-	AN1	AN0
0100011	AN35	VREF-	AN35	AN14	0000000	AN0	VREF-	AN1	AN0
0100010	AN34	VREF-	AN34	AN14					

Note 1: Not all external analog channels are available on all devices. See Table 1-3 for more information. For 64-pin devices, do not use values of ADCH<6:0> associated with unimplemented channels.

2: Single-ended conversion returns 000h + offset; differential conversions return 8000h + offset.

3: Single-ended conversion returns 000h; differential conversion returns 000h + offset and gain error.

4: Single-ended conversion returns FFFh + offset and gain error; differential conversion returns FFFh + offset and gain error.

REGISTER 26-11: ACCONH: A/D ACCUMULATOR CONTROL HIGH REGISTER

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

R/W-0, HC	R/W-0	U-0	U-0	U-0	U-0	U-0	U-0
ACEN(1)	ACIE	—	—	—	—	—	—
bit 7							bit 0

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

bit 15-8 **Unimplemented:** Read as '0'

bit 7 **ACEN:** Accumulator Enable bit(1)
- 1 = Accumulation is enabled; sample and convert the current sample list entry on trigger event and add to the contents of ACRESH/L
- 0 = The accumulation process has not started or is complete (cleared in hardware when accumulation COUNTx bits decrement to zero)

bit 6 **ACIE:** Accumulator Interrupt Enable bit
- 1 = An interrupt event is generated when the accumulator decrements to zero
- 0 = Accumulator interrupt events are disabled

bit 5-0 **Unimplemented:** Read as '0'

Note 1: To avoid unexpected or erroneous results, do not write to ACCONH or ACCONL while ACEN is set.

REGISTER 26-12: ACCONL: A/D ACCUMULATOR CONTROL LOW REGISTER

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	TBLSEL5	TBLSEL4	TBLSEL3	TBLSEL2	TBLSEL1	TBLSEL0
bit 15							bit 8

R/W-0, HSC	R/W-0, HSC	R/W-0, HSC	R/W-0, HSC	R/W-0, HSC	R/W-0, HSC	R/W-0, HSC	R/W-0, HSC
COUNT7	COUNT6	COUNT5	COUNT4	COUNT3	COUNT2	COUNT1	COUNT0
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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **TBLSEL<5:0>:** Pointer to ADTBLn Used to Select ANxx Channel to be Accumulated bits
The ANxx channel is designated by the ADTBLn register (where n = TBLSEL<5:0> value).

bit 7-0 **COUNT<7:0>**: Accumulations to be Completed Counter bits
Decrements on each accumulated sample. Before starting the accumulation process, preload this COUNTx bits field with the number of samples to accumulate (ex: To get a 9 sample sum, load COUNT with 9). Starting with a COUNT value of 0 will result in 256 samples being accumulated.

REGISTER 26-13: ADCHITH: A/D MATCH HIT HIGH REGISTER

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-0, HS	R/W-0, HS
CHH31	CHH30	CHH29	CHH28	CHH27	CHH26	CHH25	CHH24
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-0, HS	R/W-0, HS
CHH23	CHH22	CHH21	CHH20	CHH19	CHH18	CHH17	CHH16
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 31-0 **CHH<31:16>:** A/D Conversion Match Hit bits

1 = A threshold compare match has occurred on the corresponding sample list entry
0 = No match has occurred

REGISTER 26-14: ADCHITL: A/D MATCH HIT LOW REGISTER

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-0, HS	R/W-0, HS
CHH15	CHH14	CHH13	CHH12	CHH11	CHH10	CHH9	CHH8
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-0, HS	R/W-0, HS
CHH7	CHH6	CHH5	CHH4	CHH3	CHH2	CHH1	CHH0
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 31-0 **CHH<15:0>:** A/D Conversion Match Hit bits

1 = A threshold compare match has occurred on the corresponding sample list entry
0 = No match has occurred

REGISTER 26-15: ADTHnH: A/D SAMPLE TABLE n THRESHOLD VALUE HIGH REGISTER (n = 0 to 3)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
TH15	TH14	TH13	TH12	TH11	TH10	TH9	TH8
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
TH7	TH6	TH5	TH4	TH3	TH2	TH1	TH0
bit 7							bit 0

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

bit 15-0 **TH<15:0>:** High Threshold Value for Windowed Compare Operations bits (Sample Table n)
The value in 12-bit unsigned integer format only.

REGISTER 26-16: ADTHnL: A/D SAMPLE TABLE n THRESHOLD VALUE LOW REGISTER (n = 0 to 3)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
TH15	TH14	TH13	TH12	TH11	TH10	TH9	TH8
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
TH7	TH6	TH5	TH4	TH3	TH2	TH1	TH0
bit 7							bit 0

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

bit 15-0 **TH<15:0>:** Low Threshold Value for Windowed Compare Operations bits (Sample Table n)
Also serves as the comparison value for non-windowed threshold compare operations. The value in 12-bit unsigned integer format only.

REGISTER 26-17: ADLnMSEL3: A/D SAMPLE LIST n MULTI-CHANNEL SELECT REGISTER 3 (n = 0 to 3)

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	U-0	R/W-0	R/W-0
—	—	—	—	—	—	MSEL49	MSEL48
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 15-2 **Unimplemented:** Read as ‘0’

bit 1-0 **MSEL<49:48>:** A/D Channel Select bits

1 = Corresponding channel participates in multi-channel operations for Sample List n

0 = Channel does not participate in multi-channel operations

REGISTER 26-18: ADLnMSEL2: A/D SAMPLE LIST n MULTI-CHANNEL SELECT REGISTER 2 (n = 0 to 3)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
MSEL47	MSEL46	MSEL45	MSEL44	MSEL43	MSEL42	MSEL41	MSEL40
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
MSEL39	MSEL38	MSEL37	MSEL36	MSEL35	MSEL34	MSEL33	MSEL32
bit 7							bit 0

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

bit 15-0 **MSEL<47:32>:** A/D Channel Select bits

1 = Corresponding channel participates in multi-channel operations for Sample List n

0 = Channel does not participate in multi-channel operations

REGISTER 26-19: ADLnMSEL1: A/D SAMPLE LIST n MULTI-CHANNEL SELECT REGISTER 1 (n = 0 to 3)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
MSEL31	MSEL30	MSEL29	MSEL28	MSEL27	MSEL26	MSEL25	MSEL24
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
MSEL23	MSEL22	MSEL21	MSEL20	MSEL19	MSEL18	MSEL17	MSEL16
bit 7							bit 0

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

bit 15-0 **MSEL<31:16>:** A/D Channel Select bits

1 = Corresponding channel participates in multi-channel operations for Sample List n

0 = Channel does not participate in multi-channel operations

REGISTER 26-20: ADLnMSEL0: A/D SAMPLE LIST n MULTI-CHANNEL SELECT REGISTER 0 (n = 0 to 3)

R/W-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
MSEL15	—	—	—	—	—	—	—
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

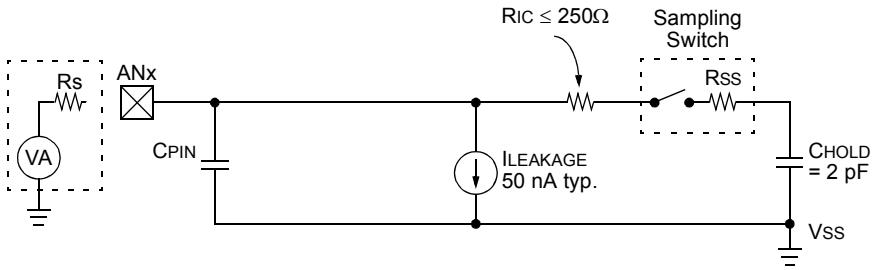
bit 15 **MSEL15:** A/D Channel Select bits

1 = Corresponding channel participates in multi-channel operations for Sample List n

0 = Channel does not participate in multi-channel operations

bit 14-0 **Unimplemented:** Read as ‘0’

FIGURE 26-2: 12-BIT A/D CONVERTER SINGLE-ENDED ANALOG INPUT MODEL

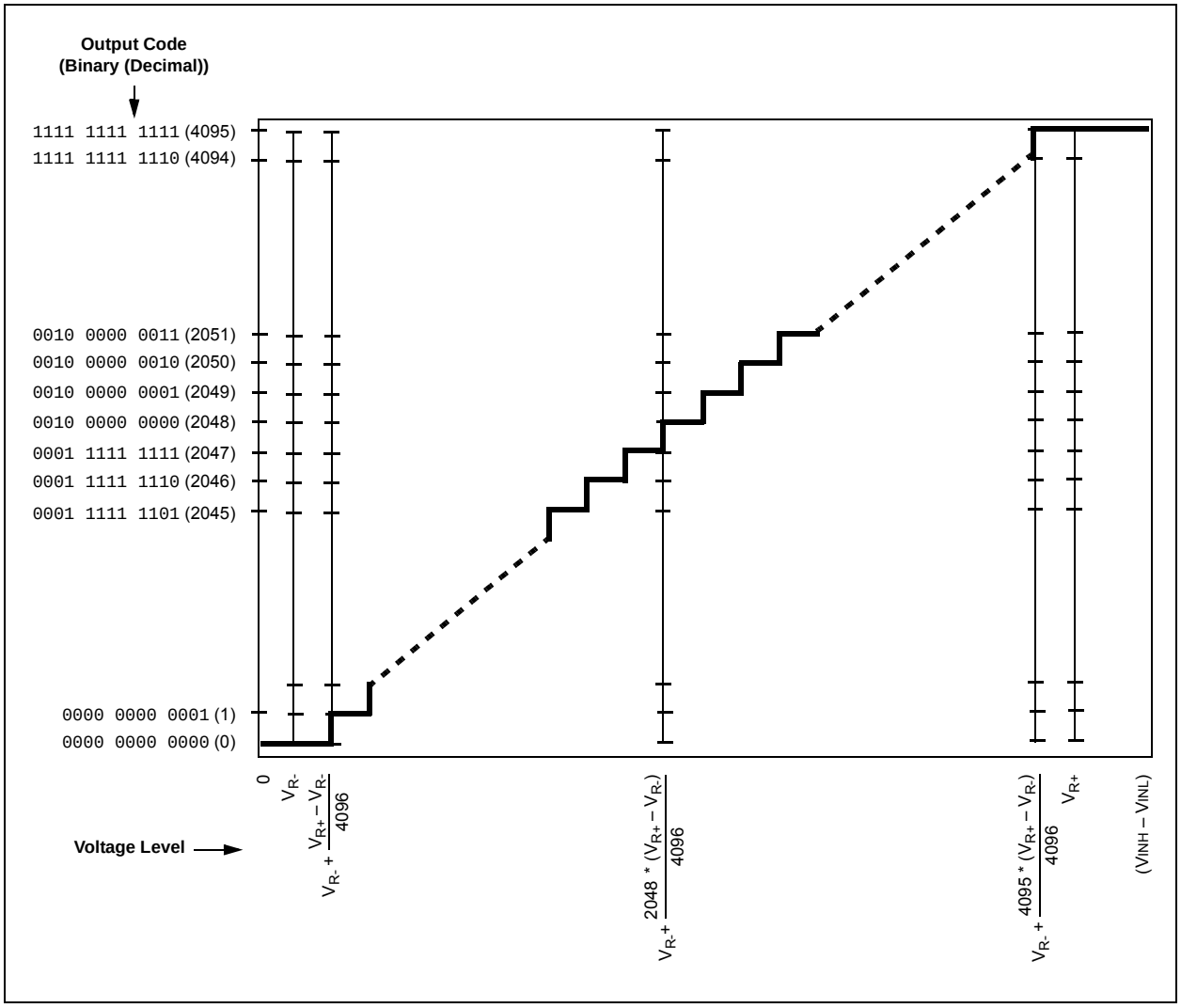


Legend:	CIN	= Pin Capacitance + Channel MUX Capacitance(2)
	VT	= Threshold Voltage
	ILEAKAGE	= Leakage Current at the pin due to various junctions
	RIC	= Interconnect Resistance
	RSS	= Sampling Switch Resistance
	CHOLD	= Sample/Hold Capacitance (from DAC)

Note 1: The CIN value depends on the device package and is not tested. The effect of CPIN is negligible if Rs ≤ 1.5 kΩ.

2: The input channel multiplexer will add capacitance based on the input channel selected. Selecting Channels AN0-AN14 will add <5 pF and selecting Channels AN15 and above will add ~16 pF.

FIGURE 26-3: 12-BIT A/D TRANSFER FUNCTION



27.0 16-BIT SIGMA-DELTA ANALOG-TO-DIGITAL (A/D) CONVERTER

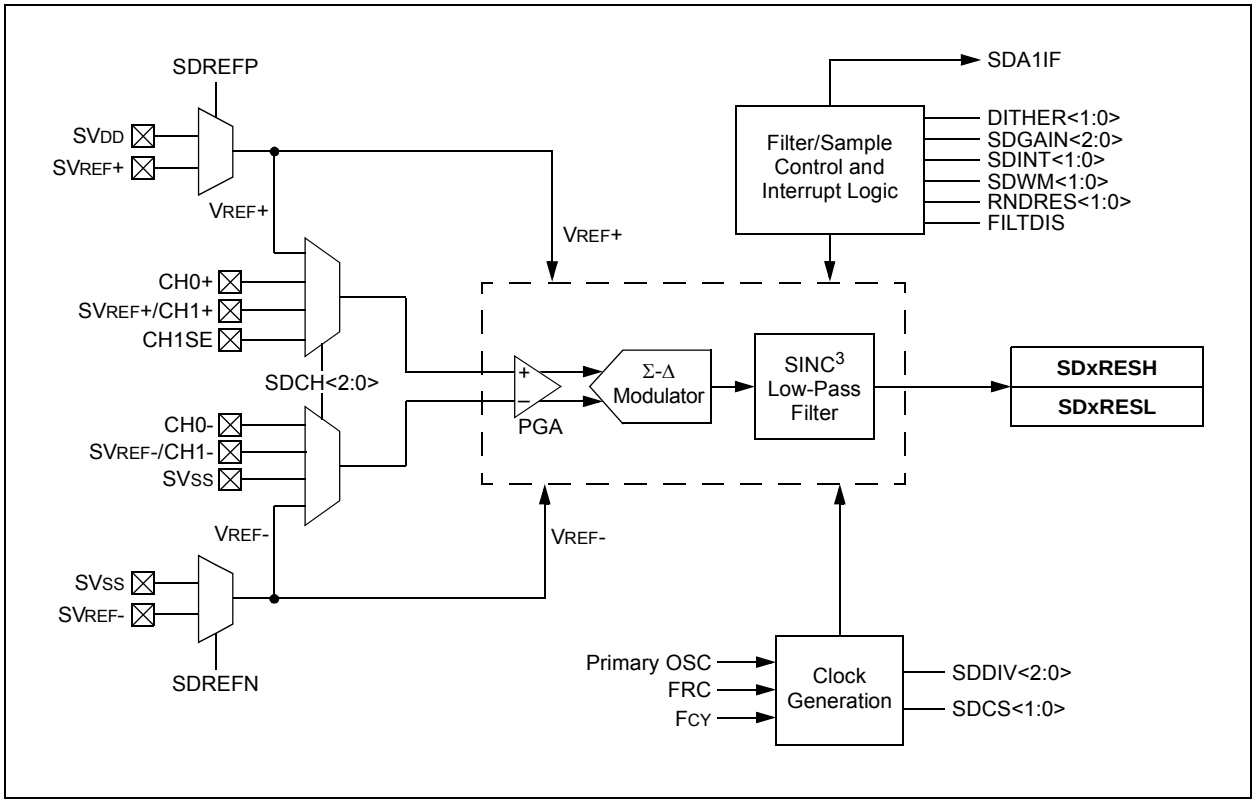
Note: This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the "*PIC24F Family Reference Manual*", **"16-Bit Sigma-Delta A/D Converter"** (DS30687). The information in this data sheet supersedes the information in the FRM.

The Sigma-Delta A/D Converter employs sigma-delta modulation techniques to convert analog signals to a digital equivalent. This method achieves exceptional resolution and output code stability, which can significantly exceed that of conventional 10-bit or 12-bit SAR-based A/Ds. A block diagram of the 16-bit Sigma-Delta A/D is shown in Figure 27-1.

Key features include:

- Adjustable sampling rates
- Configurable A/D data rates between 976 samples per second (highest quality) and 62.5 ksps (highest speed)
- Two differential input channels
- Programmable gain amplifier input
- User-selectable clock sources
- User-selectable oversampling, dithering and data rounding
- Self-measurement of internal offset and gain error
- Operation in Idle and Sleep modes
- Independent module Reset option

FIGURE 27-1: SIGMA-DELTA A/D CONVERTER BLOCK DIAGRAM



27.1 Important Differences Compared to Conventional A/Ds

In principle, the Sigma-Delta A/D Converter does what most other A/Ds do: it samples an analog input voltage and generates a digital output code representing the analog voltage. There are, however, a number of differences when comparing a Sigma-Delta Converter to conventional A/D Converters, such as the Successive Approximation Register (SAR) design that is popular on many of today's microcontrollers.

The most important differences that are noticeable at the application level include:

- Readily achieved resolution/quality of result
- Analog channel sampling methodology
- Uncorrected offset error
- Uncorrected gain error

27.1.1 RESULT QUALITY AND OVERSAMPLING

In a typical application, involving switching digital circuitry, oscillators, clocks and other noise sources common in a microcontroller-based circuit, it is often difficult to reduce the high-frequency noise floor below some arbitrary value. For A/Ds, which perform instantaneous "snapshot" based sampling (e.g., charging a Sample-and-Hold capacitor in a conventional SAR-based A/D), this noise floor ultimately restricts the maximum achievable stable result resolution.

To achieve higher effective stable resolution and to minimize the effects of high-frequency noise, the Sigma-Delta A/D Converter implements inherent oversampling in the design. This oversampling has an effect similar to low-pass filtering the analog signal and voltage references to the A/D. Therefore, when the converter generates a result, the output code represents the average voltage of the signal or reference being measured over a specific time window, rather than an instantaneous snapshot in time (like that of the SAR-based A/D). This sampling method enables the Sigma-Delta A/D Converter to generate stable results at significantly higher resolution than is typically achievable with conventional A/D designs.

The design of this Sigma-Delta A/D Converter allows user-configurable Oversampling Ratios (OSRs), between 16 and 1024. The lowest settings provide the fastest results, but they sacrifice result code accuracy. The highest OSR settings provide the best quality and most stable results, but generate results at a much slower rate.

27.1.2 UNCORRECTED OFFSET ERROR

When uncorrected, the Sigma-Delta A/D Converter typically has more LSBs worth of offset error than conventional SAR-based A/Ds. This is partly due to the high resolution and small size of each LSB. Additionally, internal or external input circuitry, such as the internal input gain stage, can also introduce some offset error.

Fortunately, the Sigma-Delta A/D Converter implements a feature that allows it to measure its own internal offset error. This feature is controlled by the VOSCAL bit (SD1CON1<4>). Once the application firmware has measured the internal offset error, the digital output code can be saved in the firmware, and subsequently subtracted from all future A/D measurements on the regular input channel(s). This procedure significantly improves the absolute accuracy of the A/D and is recommended for most applications.

27.1.3 UNCORRECTED GAIN ERROR

When uncorrected, Sigma-Delta A/D Converters typically exhibit high gain error compared to other A/D designs. To obtain high absolute accuracy from the Sigma-Delta A/D Converter, it is necessary to compensate for both offset error and gain error. Gain error can be corrected by first removing the offset error, then multiplying the resulting code with a suitable gain error correction factor.

One of the input channel settings, selectable in the SD1CON3 register, allows the A/D to measure its own references. When a measurement (with a gain of 1) is performed on this channel, the result code can be corrected for offset error (using the method described in **Section 27.1.2 "Uncorrected Offset Error"**) and then used to calculate the gain error correction factor. Once the gain error correction factor is known, it can be saved and stored in the firmware, so that it may be used later to correct for gain error when performing measurements on the other A/D input channels.

REGISTER 27-1: SD1CON1: S/D CONTROL REGISTER 1

R/W-0	U-0	R/W-0	R/W-0	r-0	R/W-0	R/W-0	R/W-0
SDON	—	SDSIDL	SDRST	r	SDGAIN2	SDGAIN1	SDGAIN0
bit 15							bit 8

R/W-0	R/W-0	U-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
DITHER1	DITHER0	—	VOSCAL	—	SDREFN	SDREFP	PWRLVL
bit 7							bit 0

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

bit 15 **SDON:** S/D Module Enable bit
1 = Module is enabled
0 = Module is disabled

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

bit 13 **SDSIDL:** S/D Stop in Idle Mode bit
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode

bit 12 **SDRST:** S/D Reset bit
1 = Resets all S/D module circuitry (analog section remains in Reset as long as bit is set)
0 = Releases from Reset (Run mode)

bit 11 **Reserved:** Maintain as '0' for proper operation

bit 10-8 **SDGAIN<2:0>:** S/D Gain Control bits
11x = Reserved
101 = 32
100 = 16
011 = 8
010 = 4
001 = 2
000 = 1

bit 7-6 **DITHER<1:0>:** Dither Mode Select bits
11 = High dither (preferred with higher Oversampling Ratio (OSR) and positive reference well below SV_{DD})
10 = Medium dither (preferred for low to medium OSR and positive reference well below SV_{DD})
01 = Low dither (preferred when the positive reference is at or near SV_{DD})
00 = No dither

bit 5 **Unimplemented:** Read as '0'

bit 4 **VOSCAL:** Internal Offset Measurement Enable bit
1 = Converter is configured to sample its own internal offset error
0 = Converter is configured for normal operation

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

bit 2 **SDREFN:** S/D Negative Reference Source Select bit
1 = SV_{REF-} pin
0 = SV_{SS} pin

bit 1 **SDREFP:** S/D Positive Reference Source Select bit
1 = SV_{REF+} pin
0 = SV_{DD} pin

bit 0 **PWRLVL:** Analog Amplifier Bandwidth Select bit
1 = 2x bandwidth (higher power consumption compared to normal bandwidth)
0 = Normal bandwidth

REGISTER 27-2: SD1CON2: S/D CONTROL REGISTER 2

R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
CHOP1	CHOP0	SDINT1	SDINT0	—	—	SDWM1	SDWM0
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	U-0	U-0	HS/C-0
—	—	—	RNDRES1	RNDRES0	—	—	SDRDY
bit 7							bit 0

Legend:	C = Clearable bit	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 15-14 **CHOP<1:0>:** Chopping Enable bits
11 = Chopping is enabled (recommended setting, improves result quality)
10 = Reserved
01 = Reserved
00 = Chopping is disabled

bit 13-12 **SDINT<1:0>:** S/D Interrupt Event Generation Select bits
11 = Interrupt on every sample clock
10 = Interrupt on every fifth sample clock
01 = Interrupt when New Result < Old Result
00 = Interrupt when New Result > Old Result

bit 11-10 **Unimplemented:** Read as '0'

bit 9-8 **SDWM<1:0>:** S/D Output Result Register Write bits
11 = Reserved; do not use
10 = SD1RESH/SD1RESL is never updated (used for threshold compare operations)
01 = SD1RESH/SD1RESL is updated on every interrupt
00 = SD1RESH/SD1RESL is updated on every interrupt when SDRDY = 0

bit 7-5 **Unimplemented:** Read as '0'

bit 4-3 **RNDRES<1:0>:** Round Data Control bits
11 = Round result to 8 bits
10 = Round result to 16 bits
01 = Round result to 24 bits
00 = No Rounding

bit 2-1 **Unimplemented:** Read as '0'

bit 0 **SDRDY:** S/D Filter Data Ready bit (set by hardware)
1 = Sync filter delay is satisfied (clear this bit in software)
0 = Sync filter delay is not satisfied yet

REGISTER 27-3: SD1CON3: S/D CONTROL REGISTER 3

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SDDIV2(1)	SDDIV1(1)	SDDIV0(1)	SDOSR2	SDOSR1	SDOSR0	SDCS1	SDCS0
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	SDCH2	SDCH1	SDCH0
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 **SDDIV<2:0>:** S/D Input Clock Divider/Postscaler Ratio bits(1)
- 111 = Reserved
- 110 = 64
- 101 = 32
- 100 = 16
- 011 = 8
- 010 = 4
- 001 = 2
- 000 = 1 (No divider, clock selected by SDCS<1:0> provided directly to A/D)

bit 12-10 **SDOSR<2:0>:** S/D Oversampling Ratio (OSR) Selection bits
- 111 = Reserved
- 110 = 16 (fastest result, lowest quality)
- 101 = 32
- 100 = 64
- 011 = 128
- 010 = 256
- 001 = 512
- 000 = 1024 (slowest result, best quality)

bit 9-8 **SDCS<1:0>:** S/D A/D Module Clock Source Select bits
- 11 = Reserved
- 10 = Primary Oscillator (OSCI/CLKI)
- 01 = FRC (8 MHz)(2)
- 00 = System clock (FOSC/2)

bit 7-3 **Unimplemented:** Read as '0'

bit 2-0 **SDCH<2:0>:** S/D Analog Channel Input Select bits (positive input/negative input)
- 1xx = Reserved
- 011 = Measures the reference selected by SDREFP/SDREFN (used for gain error measurements)
- 010 = CH1SE/SVSS (single-ended measurement of CH1SE)
- 001 = CH1+/CH1- (Differential Channel 1)
- 000 = CH0+/CH0- (Differential Channel 0)

Note 1: To avoid overclocking or underclocking the module, set SDDIV<2:0> to obtain an A/D clock frequency (input frequency selected by SDCS<1:0> source, divided by selected SDDIVx ratio) at or between 1 MHz and 4 MHz.

2: 8 MHz FRC output is used directly, prior to the FRCDIV postscaler.

NOTES:

28.0 10-BIT DIGITAL-TO-ANALOG CONVERTER (DAC)

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*", **"10-Bit Digital-to-Analog Converter (DAC)"** (DS39615). The information in this data sheet supersedes the information in the FRM.

PIC24FJ128GC010 family devices include two 10-bit Digital-to-Analog Converters (DACs) for generating analog outputs from digital data. A simplified block diagram for a single DAC is shown in Figure 28-1. Both of the DACs are identical.

The DAC generates an analog output voltage based on the digital input code, according to the formula:

$$V_{DAC} = \frac{V_{DACREF} \times DACxDAT}{1024}$$

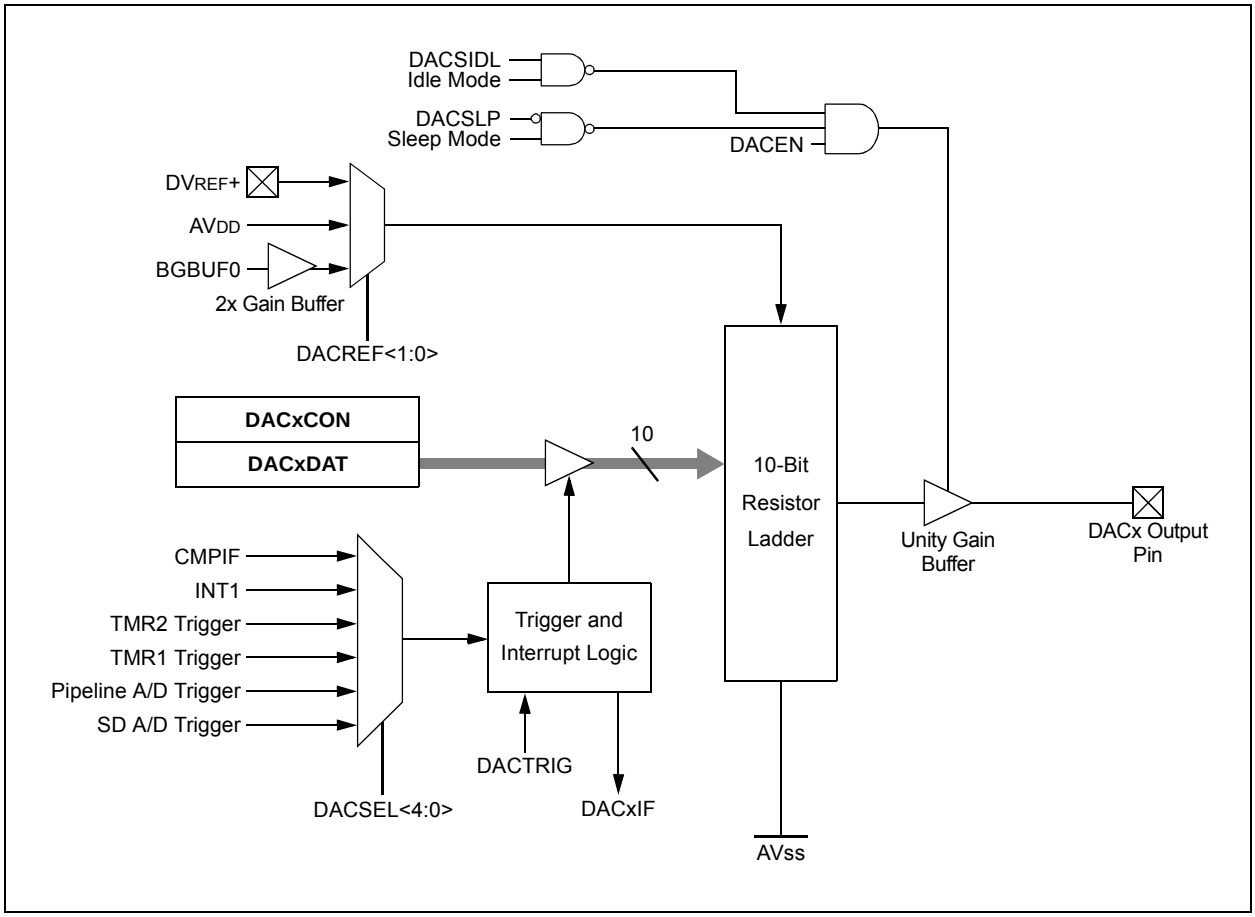
where V_{DAC} is the analog output voltage and V_{DACREF} is the reference voltage selected by DACREF<1:0>.

Each DAC includes these features:

- Precision 10-bit resistor ladder for high accuracy
- Fast settling time, supporting 1 Msps effective sampling rates
- Buffered output voltage
- Three user-selectable voltage reference options
- Multiple conversion trigger options, plus a manual convert-on-write option
- Left and right justified input data options
- User-selectable Sleep and Idle mode operation

When using the DAC, it is required to set the ANSx and TRISx bits for the DACx output pin to configure it as an analog output. See **Section 11.2 "Configuring Analog Port Pins (ANSx)"** for more information.

FIGURE 28-1: SINGLE DAC SIMPLIFIED BLOCK DIAGRAM



REGISTER 28-1: DACxCON: DACx CONTROL REGISTER

R/W-0	U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0
DACEN	—	DACSIDL	DACSLP	DACFM	—	—	DACTRIG
bit 15							bit 8

U-1	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	DACTSEL4	DACTSEL3	DACTSEL2	DACTSEL1	DACTSEL0	DACREF1	DACREF0
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 **DACEN:** DAC Enable bit
1 = Module is enabled
0 = Module is disabled

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

bit 13 **DACSIDL:** DAC Peripheral Stop in Idle Mode bit
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode

bit 12 **DACSLP:** DAC Enable Peripheral During Sleep bit
1 = DAC continues to output the most recent value of DACxDAT during Sleep mode
0 = DAC is powered down in Sleep mode; DACx output pin is controlled by the TRISx and LATx bits

bit 11 **DACFM:** DAC Data Format Select bit
1 = Data is left justified (data stored in DACxDAT<15:5>)
0 = Data is right justified (data stored in DACxDAT<9:0>)

bit 10-9 **Unimplemented:** Read as '0'

bit 8 **DACTRIG:** DAC Trigger Input Enable bit
1 = Analog output value updates when the selected (by DACTSEL<4:0>) event occurs
0 = Analog output value updates as soon as DACxDAT is written (DAC trigger is ignored)

bit 7 **Unimplemented:** Recommended to write as '1' for code compatibility across device families

bit 6-2 **DACTSEL<4:0>:** DAC Trigger Source Select bits
11x = Unimplemented
101 = S/D A/D interrupt
100 = Pipeline A/D interrupt
011 = Timer1 interrupt
010 = Timer2 interrupt
001 = INT1
000 = Comparator 1 trigger

bit 1-0 **DACREF<1:0>:** DAC Reference Source Select bits
11 = 2.4V Internal Band Gap (2 • BGBUF0)(1)
10 = AV_{DD}
01 = DV_{REF+}
00 = Reference not connected (lowest power but no DAC functionality)

Note 1: User must also enable Band Gap Buffer 0 and set BUFCON0<1:0> to '00' to obtain this voltage. See Register 25-1 for details.

29.0 DUAL OPERATIONAL AMPLIFIER 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 *"PIC24F Family Reference Manual"*, **"Operational Amplifier (Op Amp)"** (DS30505). Device-specific information in this data sheet supersedes the information in the *"PIC24F Family Reference Manual"*.

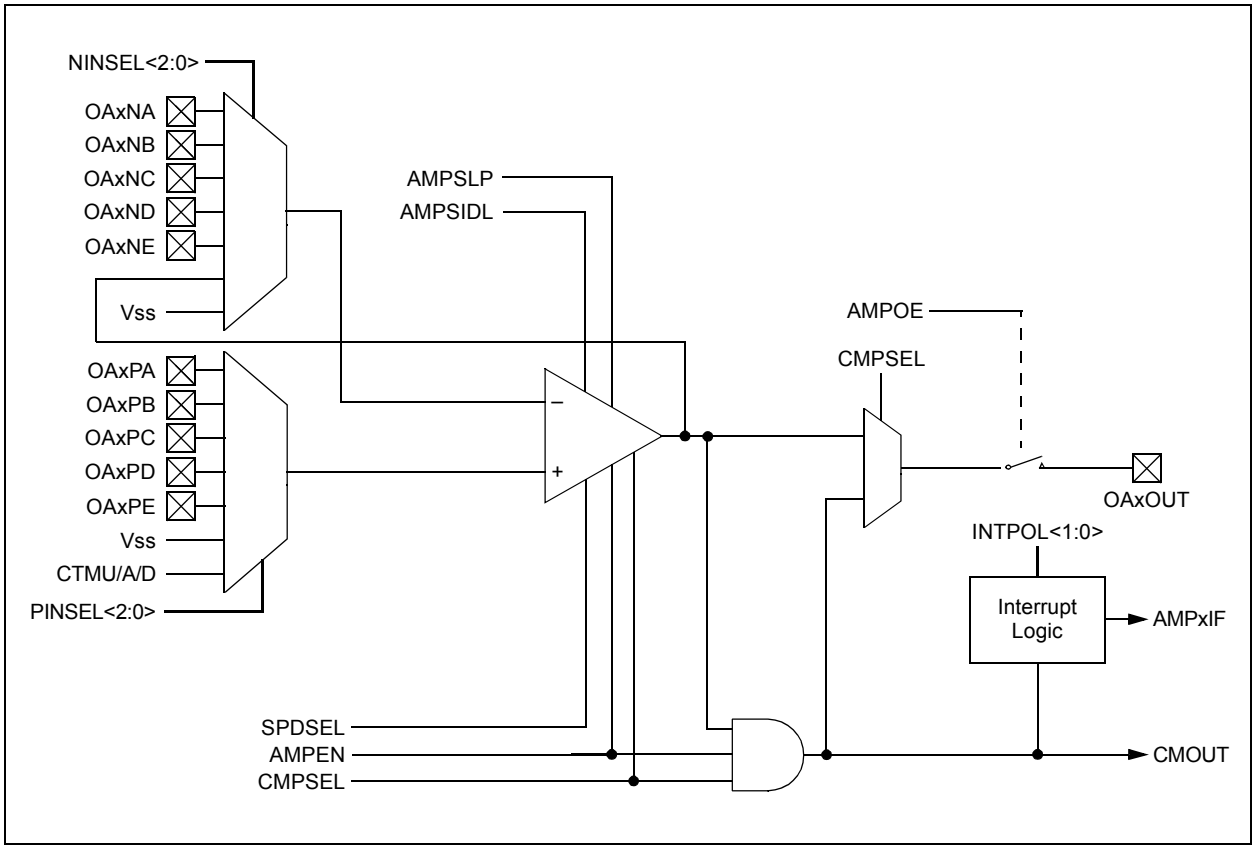
PIC24FJ128GC010 family devices include two operational amplifiers to complement the microcontroller's other analog features. They may be used to provide analog signal conditioning, either as stand-alone devices or in addition to other analog peripherals. They may also be configured to operate as digital comparators in addition to the triple comparator module (see **Section 30.0 "Triple Comparator Module"** for more information).

The two op amps are functionally identical; the block diagram for a single amplifier is shown in Figure 29-1. Each op amp has these features:

- Configurable as either an operational amplifier or a comparator
- Internal unity-gain buffer option
- Six input options each on the inverting and non-inverting amplifier inputs
- Rail-to-rail input and output capabilities
- User-configurable interrupt with comparator operation, including four interrupt options
- User-selectable option for regular or low-power operation
- User-selectable operation in Idle and Sleep modes

When using the op amps, it is recommended to set the ANSx and TRISx bits of both the input and output pins to configure them as analog pins. See **Section 11.2 "Configuring Analog Port Pins (ANSx)"** for more information.

FIGURE 29-1: SINGLE OPERATIONAL AMPLIFIER BLOCK DIAGRAM



REGISTER 29-1: AMPxCON: OP AMP x CONTROL REGISTER

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R-0	R-x	R/W-0
AMPEN	—	AMPSIDL	AMPSLP	INTPOL1	INTPOL0	CMOUT	CMPSEL
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
SPDSEL	AMPOE	NINSEL2	NINSEL1	NINSEL0	PINSEL2	PINSEL1	PINSEL0
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 **AMPEN:** Op Amp Control Module Enable bit
1 = Module is enabled
0 = Module is disabled

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

bit 13 **AMPSIDL:** Op Amp Peripheral Stop in Idle Mode bit
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode

bit 12 **AMPSLP:** Op Amp Peripheral Enabled in Sleep Mode bit
1 = Continues module operation when device enters Sleep mode
0 = Discontinues module operation in Sleep mode

bit 11-10 **INTPOL<1:0>:** Interrupt Mode Select bits
When CMPSEL = 1:
11 = Interrupt occurs on any change
10 = Interrupt occurs on negative edge
01 = Interrupt occurs on positive edge
00 = Interrupts are disabled
When CMPSEL = 0:
Op amp interrupts are not generated.

bit 9 **CMOUT:** Comparator Mode Output State bit
When CMPSEL = 1:
1 = Non-inverting input is greater than the inverting input
0 = Non-inverting input is less than the inverting input
When CMPSEL = 0:
Op Amp mode (no digital state information is generated).

bit 8 **CMPSEL:** Op Amp Mode Select bit
1 = Configured as a comparator
0 = Configured as an op amp

bit 7 **SPDSEL:** Op Amp/Comparator Power/Speed Select bit
1 = Higher power and bandwidth (faster response time)
0 = Lower power and bandwidth (slower response time)

bit 6 **AMPOE:** Amplifier Output Enable bit
1 = Amplifier or comparator output is sent to OAxOUT pin
0 = Amplifier or comparator output is not sent to OAxOUT (pin is controlled by TRISx and LATx bits)

REGISTER 29-1: AMPxCON: OP AMP x CONTROL REGISTER

bit 5-3 **NINSEL<2:0>:** Op Amp Inverting Input Select bits
111 = Reserved; do not use
110 = Op Amp output (voltage follower configuration)
101 = OAxNE
100 = OAxND
011 = OAxNC
010 = OAxNB
001 = OAxNA
000 = Vss

bit 2-0 **PINSEL<2:0>:** Op Amp Non-Inverting Input Select bits
111 = Reserved; do not use
110 = Connected between CTMU output and Pipeline A/D
101 = OAxPE
100 = OAxPD
011 = OAxPC
010 = OAxPB
001 = OAxPA
000 = Vss

NOTES:

30.0 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 "*PIC24F Family Reference Manual*", **"Scalable Comparator Module"** (DS39734). The information in this data sheet supersedes the information in the FRM.

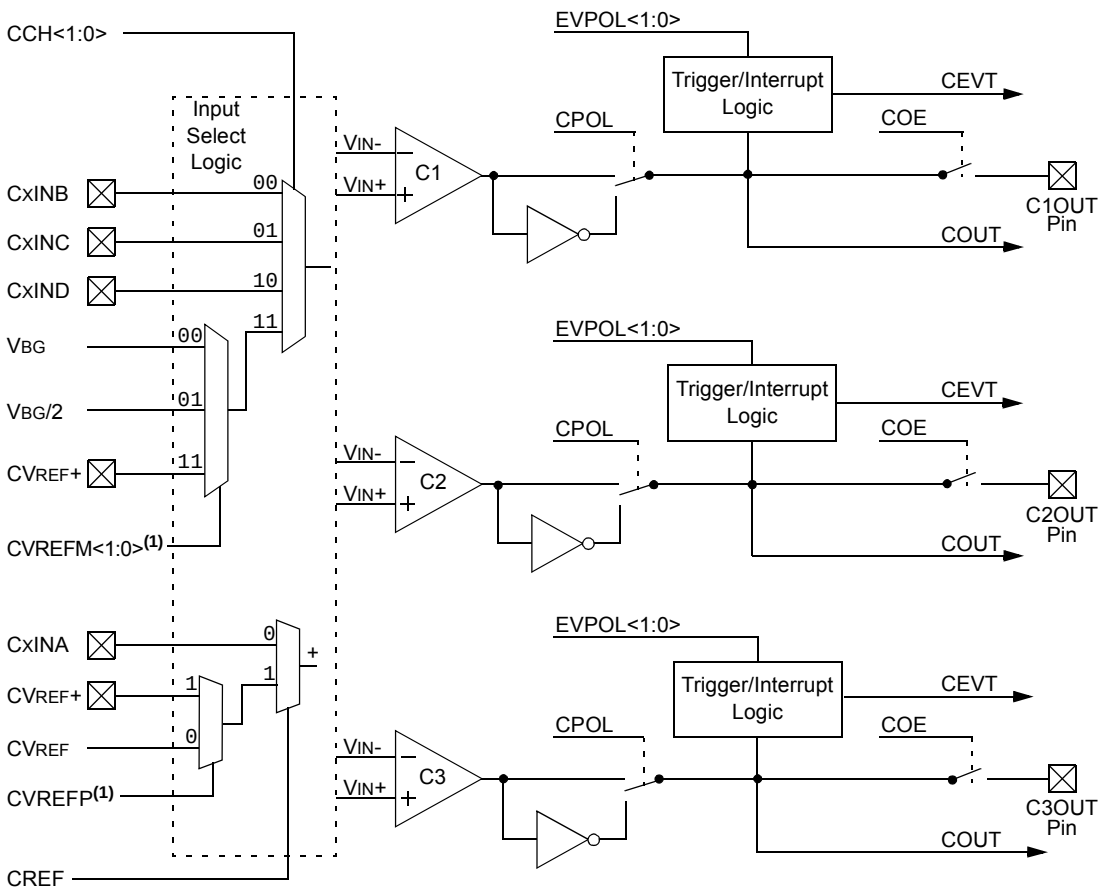
The triple comparator module provides three dual input comparators. The inputs to the comparator can be configured to use any one of five external analog inputs (CxINA, CxINB, CxINC, CxIND and VREF+) and a voltage reference input from one of the internal band gap references or the comparator voltage 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 30-1. Diagrams of the possible individual comparator configurations are shown in Figure 30-2.

Each comparator has its own control register, CMxCON (Register 30-1), for enabling and configuring its operation. The output and event status of all three comparators is provided in the CMSTAT register (Register 30-2).

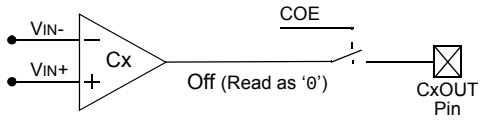
FIGURE 30-1: TRIPLE COMPARATOR MODULE BLOCK DIAGRAM



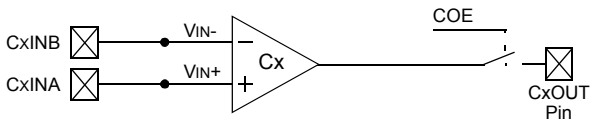
Note 1: Refer to the CVRCON register (Register 31-1) for bit details.

FIGURE 30-2: INDIVIDUAL COMPARATOR CONFIGURATIONS WHEN CREF = 0

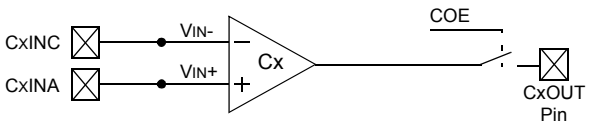
Comparator Off
CEN = 0, CREF = x, CCH<1:0> = xx



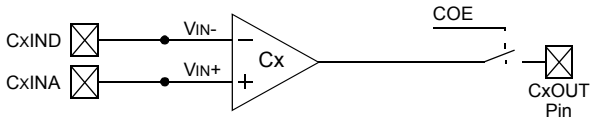
Comparator CxINB > CxINA Compare
CEN = 1, CCH<1:0> = 00, CVREFM<1:0> = xx



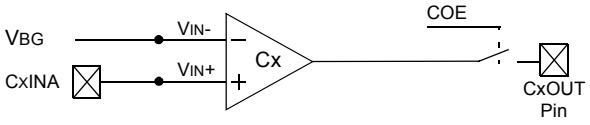
Comparator CxINC > CxINA Compare
CEN = 1, CCH<1:0> = 01, CVREFM<1:0> = xx



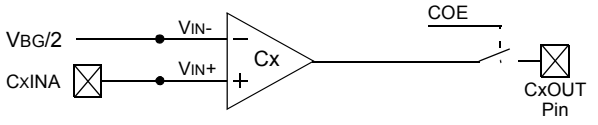
Comparator CxIND > CxINA Compare
CEN = 1, CCH<1:0> = 10, CVREFM<1:0> = xx



Comparator VBG > CxINA Compare
CEN = 1, CCH<1:0> = 11, CVREFM<1:0> = 00



Comparator VBG > CxINA Compare
CEN = 1, CCH<1:0> = 11, CVREFM<1:0> = 01



Comparator CxIND > CxINA Compare
CEN = 1, CCH<1:0> = 11, CVREFM<1:0> = 11

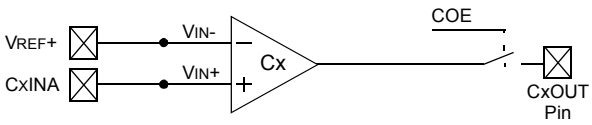


FIGURE 30-3: INDIVIDUAL COMPARATOR CONFIGURATIONS WHEN CREF = 1 AND CVREFP = 0

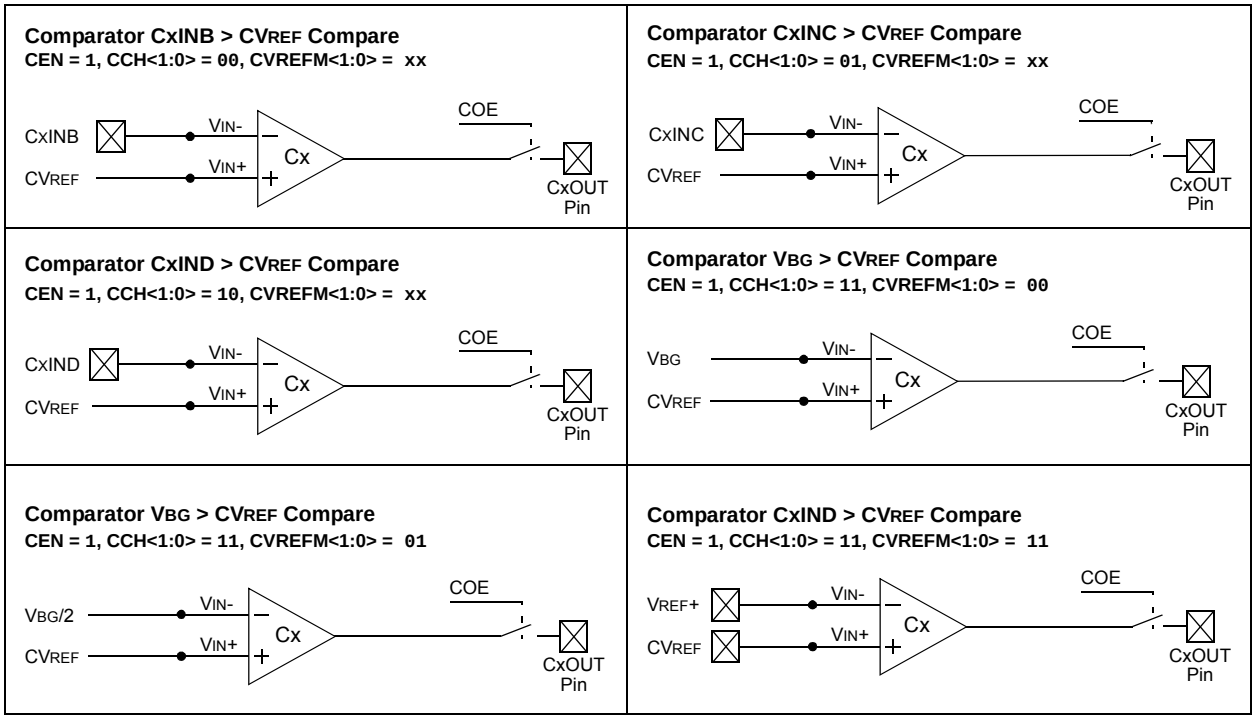
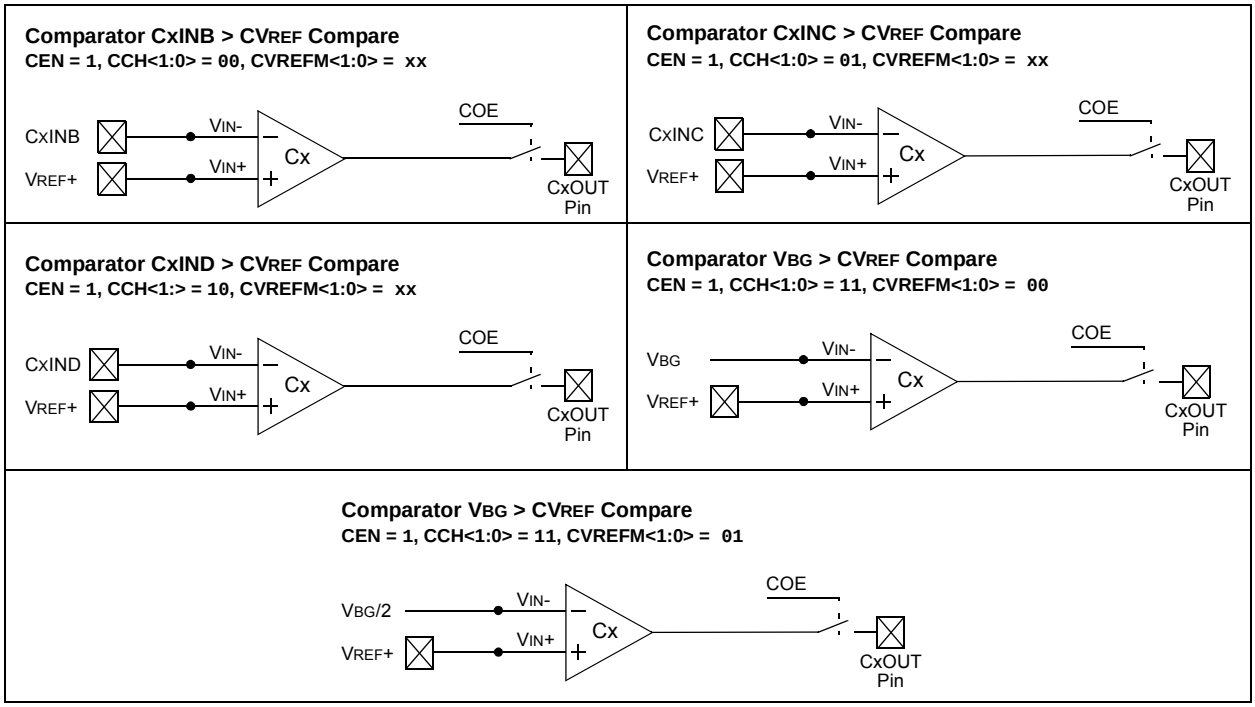


FIGURE 30-4: INDIVIDUAL COMPARATOR CONFIGURATIONS WHEN CREF = 1 AND CVREFP = 1



REGISTER 30-1: CMxCON: COMPARATOR x CONTROL REGISTERS (COMPARATORS 1 THROUGH 3)

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	R/W-0, HS	R-0, HSC
CON	COE	CPOL	—	—	—	CEVT	COUT
bit 15							bit 8

R/W-0	R/W-0	U-0	R/W-0	U-0	U-0	R/W-0	R/W-0
EVPOL1	EVPOL0	—	CREF	—	—	CCH1	CCH0
bit 7							bit 0

Legend:	HS = Hardware Settable bit	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 **CON:** Comparator Enable bit
1 = Comparator is enabled
0 = Comparator is disabled

bit 14 **COE:** Comparator Output Enable bit
1 = Comparator output is present on the CxOUT pin
0 = Comparator output is internal only

bit 13 **CPOL:** Comparator Output Polarity Select bit
1 = Comparator output is inverted
0 = Comparator output is not inverted

bit 12-10 **Unimplemented:** Read as '0'

bit 9 **CEVT:** Comparator Event bit
1 = Comparator event that is defined by EVPOL<1:0> has occurred; subsequent triggers and interrupts are disabled until the bit is cleared
0 = Comparator event has not occurred

bit 8 **COUT:** Comparator Output bit
When CPOL = 0:
1 = VIN+ > VIN-
0 = VIN+ < VIN-
When CPOL = 1:
1 = VIN+ < VIN-
0 = VIN+ > VIN-

bit 7-6 **EVPOL<1:0>:** Trigger/Event/Interrupt Polarity Select bits
11 = Trigger/event/interrupt is generated on any change of the comparator output (while CEVT = 0)
10 = Trigger/event/interrupt is generated on transition of the comparator output:
If CPOL = 0 (non-inverted polarity):
High-to-low transition only.
If CPOL = 1 (inverted polarity):
Low-to-high transition only.
01 = Trigger/event/interrupt is generated on transition of comparator output:
If CPOL = 0 (non-inverted polarity):
Low-to-high transition only.
If CPOL = 1 (inverted polarity):
High-to-low transition only.
00 = Trigger/event/interrupt generation is disabled

bit 5 **Unimplemented:** Read as '0'

REGISTER 30-1: CMxCON: COMPARATOR x CONTROL REGISTERS (COMPARATORS 1 THROUGH 3) (CONTINUED)

bit 4 **CREF:** Comparator Reference Select bits (non-inverting input)
1 = Non-inverting input connects to the internal CVREF voltage
0 = Non-inverting input connects to the CxINA pin

bit 3-2 **Unimplemented:** Read as '0'

bit 1-0 **CCH<1:0>:** Comparator Channel Select bits
11 = Inverting input of the comparator connects to the internal selectable reference voltage specified by the CVREFM<1:0> bits in the CVRCON register
10 = Inverting input of the comparator connects to the CxIND pin
01 = Inverting input of the comparator connects to the CxINC pin
00 = Inverting input of the comparator connects to the CxINB pin

REGISTER 30-2: CMSTAT: COMPARATOR MODULE STATUS REGISTER

R/W-0	U-0	U-0	U-0	U-0	R-0, HSC	R-0, HSC	R-0, HSC
CMIDL	—	—	—	—	C3EVT	C2EVT	C1EVT
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R-0, HSC	R-0, HSC	R-0, HSC
—	—	—	—	—	C3OUT	C2OUT	C1OUT
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 **CMIDL:** Comparator Stop in Idle Mode bit
1 = Discontinues operation of all comparators when device enters Idle mode
0 = Continues operation of all enabled comparators in Idle mode

bit 14-11 **Unimplemented:** Read as '0'

bit 10 **C3EVT:** Comparator 3 Event Status bit (read-only)
Shows the current event status of Comparator 3 (CM3CON<9>).

bit 9 **C2EVT:** Comparator 2 Event Status bit (read-only)
Shows the current event status of Comparator 2 (CM2CON<9>).

bit 8 **C1EVT:** Comparator 1 Event Status bit (read-only)
Shows the current event status of Comparator 1 (CM1CON<9>).

bit 7-3 **Unimplemented:** Read as '0'

bit 2 **C3OUT:** Comparator 3 Output Status bit (read-only)
Shows the current output of Comparator 3 (CM3CON<8>).

bit 1 **C2OUT:** Comparator 2 Output Status bit (read-only)
Shows the current output of Comparator 2 (CM2CON<8>).

bit 0 **C1OUT:** Comparator 1 Output Status bit (read-only)
Shows the current output of Comparator 1 (CM1CON<8>).

NOTES:

31.0 COMPARATOR VOLTAGE REFERENCE

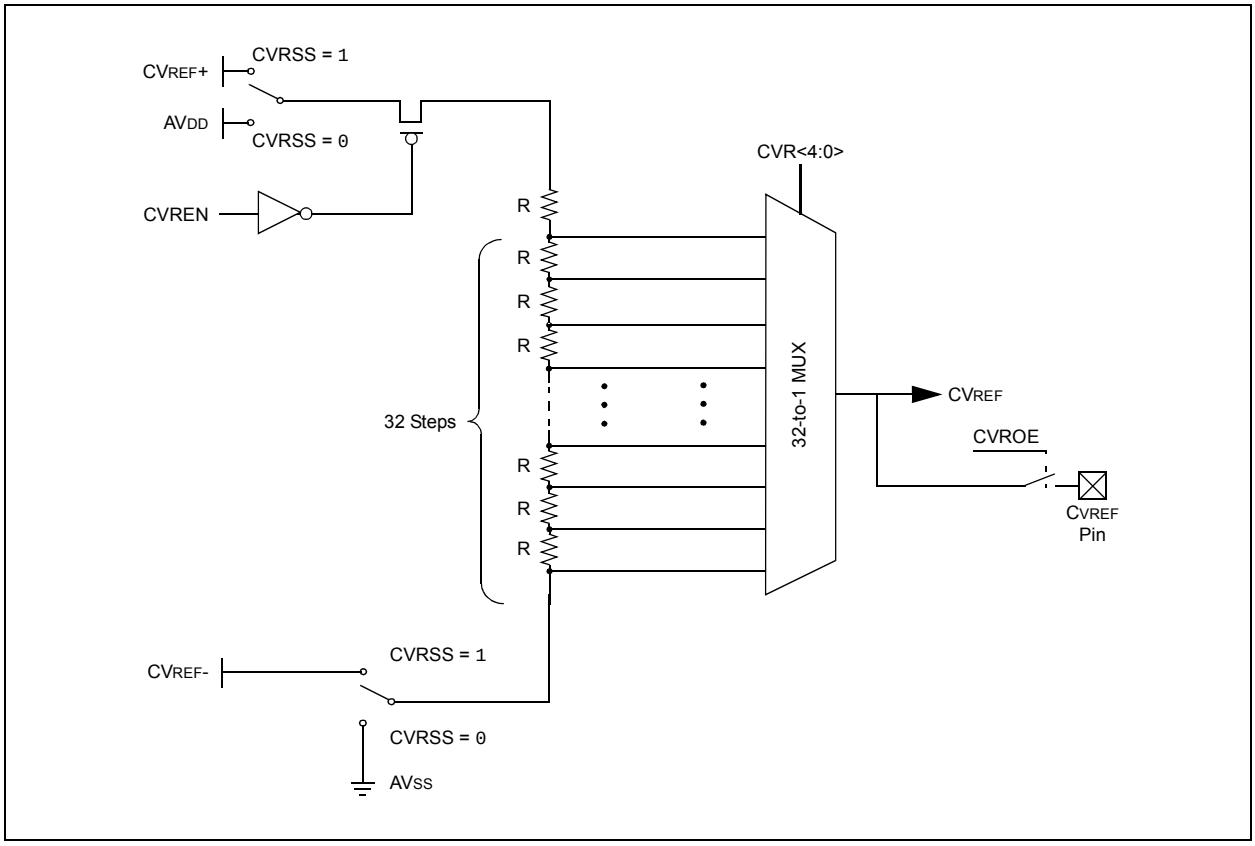
Note: This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the "*PIC24F Family Reference Manual*", **"Dual Comparator Module"** (DS39710). The information in this data sheet supersedes the information in the FRM.

31.1 Configuring the Comparator Voltage Reference

The voltage reference module is controlled through the CVRCON register (Register 31-1). The comparator voltage reference provides a range of output voltages, with 32 distinct levels. The comparator reference supply voltage can come from either VDD and VSS, or the external CVREF+ and CVREF- pins. The voltage source is selected by the CVRSS bit (CVRCON<5>).

The settling time of the comparator voltage reference must be considered when changing the CVREF output.

FIGURE 31-1: COMPARATOR VOLTAGE REFERENCE BLOCK DIAGRAM



REGISTER 31-1: CVRCON: COMPARATOR VOLTAGE REFERENCE CONTROL REGISTER

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	CVREFP	CVREFM1	CVREFM0
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
CVREN	CVROE	CVRSS	CVR4	CVR3	CVR2	CVR1	CVR0
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 **CVREFP:** Comparator Voltage Reference Select bit (valid only when CREF is '1')
1 = VREF+ is used as a reference voltage to the comparators
0 = The CVRx (5-bit DAC) within this module provides the reference voltage to the comparators

bit 9-8 **CVREFM<1:0>:** Comparator Voltage Band Gap Reference Source Select bits
(valid only when CCH<1:0> = 11)
00 = Band gap voltage is provided as an input to the comparators
01 = Band gap voltage, divided by two, is provided as an input to the comparators
10 = Reserved
11 = VREF+ pin is provided as an input to the comparators

bit 7 **CVREN:** Comparator Voltage Reference Enable bit
1 = CVREF circuit is powered on
0 = CVREF circuit is powered down

bit 6 **CVROE:** Comparator VREF Output Enable bit
1 = CVREF voltage level is output on the CVREF pin
0 = CVREF voltage level is disconnected from the CVREF pin

bit 5 **CVRSS:** Comparator VREF Source Selection bit
1 = Comparator reference source, CVRSRC = VREF+ – VREF-
0 = Comparator reference source, CVRSRC = AVDD – AVSS

bit 4-0 **CVR<4:0>:** Comparator VREF Value Selection bits
CVREF = (CVR<4:0>/32) • (CVRSRC)

32.0 CHARGE TIME MEASUREMENT UNIT (CTMU)

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 Charge Measurement Unit, refer to the *"PIC24F Family Reference Manual"*, **"Charge Time Measurement Unit (CTMU) with Threshold Detect"** (DS39743).

The Charge Time Measurement Unit (CTMU) is a flexible analog module that provides charge measurement, accurate differential time measurement between pulse sources and asynchronous pulse generation. Its key features include:

- Thirteen external edge input trigger sources
- Polarity control for each edge source
- Control of edge sequence
- Control of response to edge levels or edge transitions
- Time measurement resolution of one nanosecond
- Accurate current source suitable for capacitive measurement

Together with other on-chip analog modules, the CTMU can be used to precisely measure time, measure capacitance, measure relative changes in capacitance or generate output pulses that are independent of the system clock. The CTMU module is ideal for interfacing with capacitive-based touch sensors.

The CTMU is controlled through three registers: CTMUCON1, CTMUCON2 and CTMUICON. CTMUCON1 enables the module and controls the mode of operation of the CTMU, as well as controlling edge sequencing. CTMUCON2 controls edge source selection and edge source polarity selection. The CTMUICON register selects the current range of current source and trims the current.

32.1 Measuring Capacitance

The CTMU module measures capacitance by generating an output pulse with a width equal to the time between edge events on two separate input channels. The pulse edge events to both input channels can be selected from four sources: two internal peripheral modules (OC1 and Timer1) and up to 13 external pins (CTEDG1 through CTEDG13). This pulse is used with the module's precision current source to calculate capacitance according to the relationship:

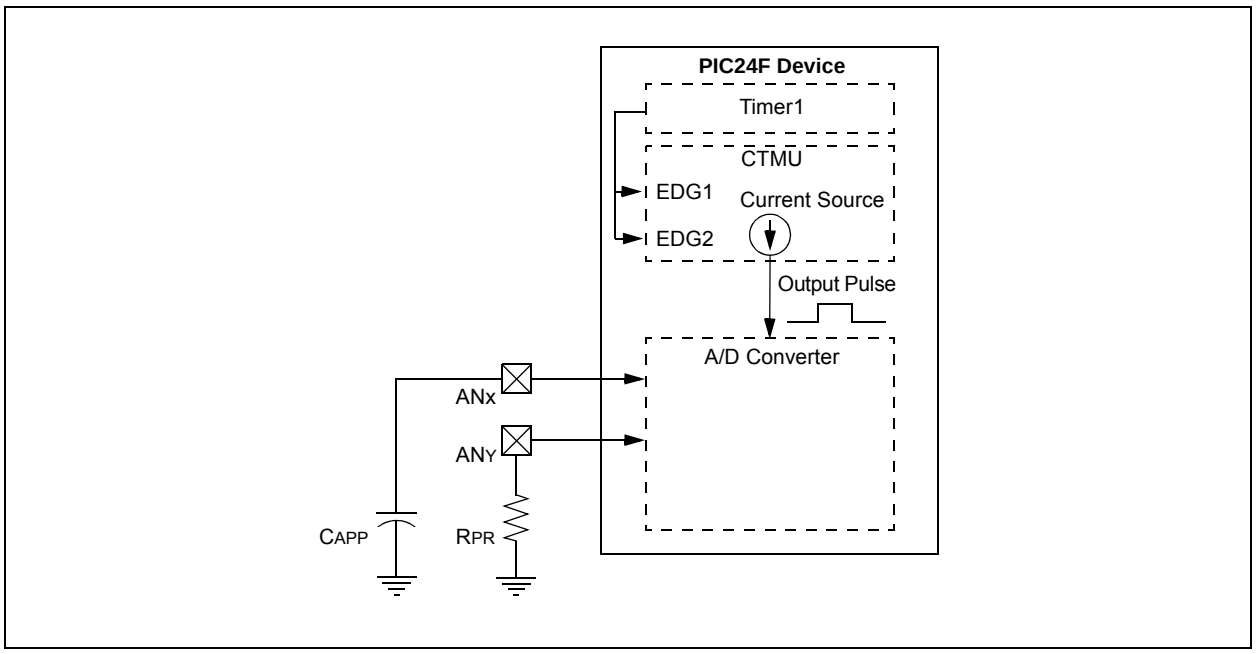
EQUATION 32-1:

$$I = C \cdot \frac{dV}{dT}$$

For capacitance measurements, the A/D Converter samples an external capacitor (CAPP) on one of its input channels, after the CTMU output's pulse. A precision resistor (RPR) provides current source calibration on a second A/D channel. After the pulse ends, the converter determines the voltage on the capacitor. The actual calculation of capacitance is performed in software by the application.

Figure 32-1 illustrates the external connections used for capacitance measurements, and how the CTMU and A/D modules are related in this application. This example also shows the edge events coming from Timer1, but other configurations using external edge sources are possible. A detailed discussion on measuring capacitance and time with the CTMU module is provided in the "*PIC24F Family Reference Manual",* **"Charge Time Measurement Unit (CTMU)"**.

FIGURE 32-1: TYPICAL CONNECTIONS AND INTERNAL CONFIGURATION FOR CAPACITANCE MEASUREMENT



32.2 Measuring Time

Time measurements on the pulse width can be similarly performed using the A/D module's Internal Capacitor (C_{AD}) and a precision resistor for current calibration. Figure 32-2 displays the external connections used for time measurements, and how the CTMU and A/D modules are related in this application. This example also shows both edge events coming from the external CTEDGx pins, but other configurations using internal edge sources are possible.

32.3 Pulse Generation and Delay

The CTMU module can also generate an output pulse with edges that are not synchronous with the device's system clock. More specifically, it can generate a pulse with a programmable delay from an edge event input to the module.

When the module is configured for pulse generation delay by setting the TGEN bit (CTMUCON1<12>), the internal current source is connected to the B input of Comparator 2. A capacitor (C_{DELAY}) is connected to the Comparator 2 pin, C2INB, and the Comparator Voltage Reference, CV_{REF}, is connected to C2INA. CV_{REF} is then configured for a specific trip point. The module begins to charge C_{DELAY} when an edge event is detected. When C_{DELAY} charges above the CV_{REF} trip point, a pulse is output on CTPLS. The length of the pulse delay is determined by the value of C_{DELAY} and the CV_{REF} trip point.

Figure 32-3 illustrates the external connections for pulse generation, as well as the relationship of the different analog modules required. While CTED1 is shown as the input pulse source, other options are available. A detailed discussion on pulse generation with the CTMU module is provided in the "*PIC24F Family Reference Manual*".

FIGURE 32-2: TYPICAL CONNECTIONS AND INTERNAL CONFIGURATION FOR TIME MEASUREMENT

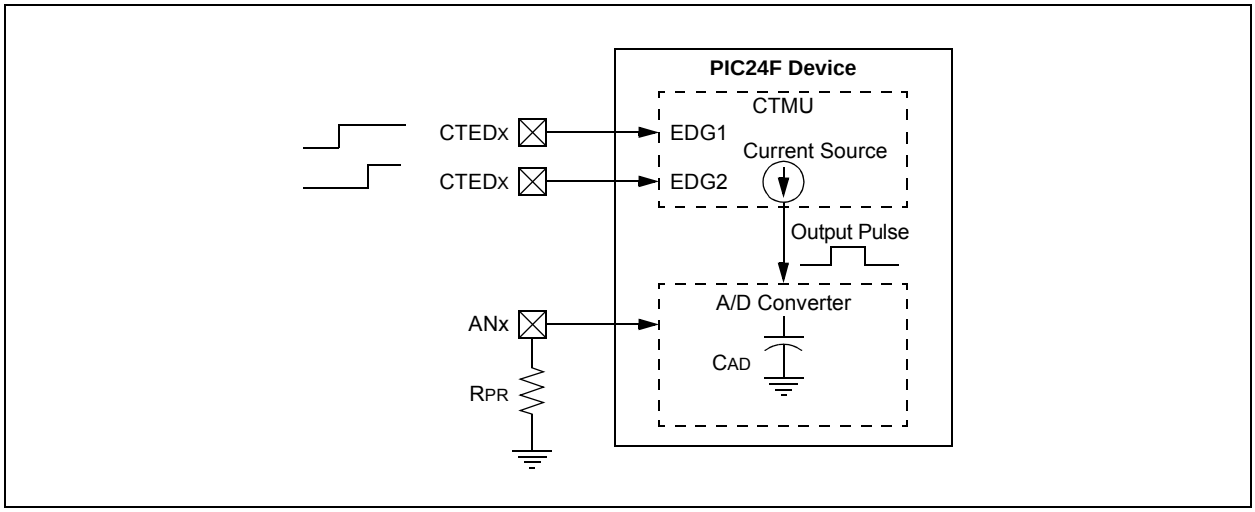
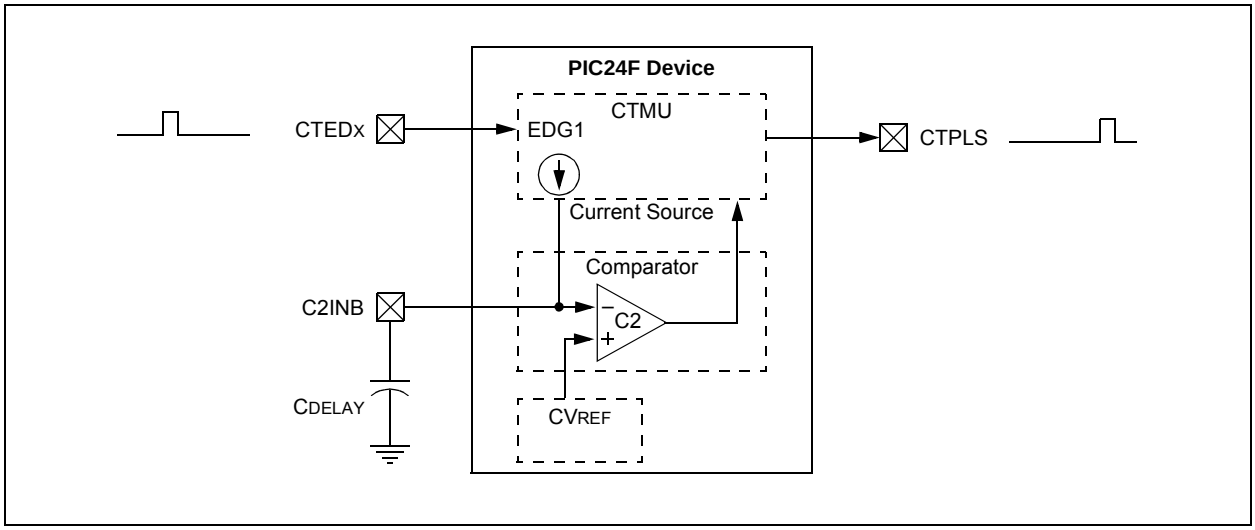


FIGURE 32-3: TYPICAL CONNECTIONS AND INTERNAL CONFIGURATION FOR PULSE DELAY GENERATION



REGISTER 32-1: CTMUCON1: CTMU CONTROL REGISTER 1

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CTMUEN	—	CTMUSIDL	TGEN	EDGEN	EDGSEQEN	IDISSEN	CTTRIG
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 **CTMUEN:** CTMU Enable bit
1 = Module is enabled
0 = Module is disabled

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

bit 13 **CTMUSIDL:** CTMU Stop in Idle Mode bit
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode

bit 12 **TGEN:** Time Generation Enable bit
1 = Enables edge delay generation
0 = Disables edge delay generation

bit 11 **EDGEN:** Edge Enable bit
1 = Edges are not blocked
0 = Edges are blocked

bit 10 **EDGSEQEN:** Edge Sequence Enable bit
1 = Edge 1 event must occur before Edge 2 event can occur
0 = No edge sequence is needed

bit 9 **IDISSEN:** Analog Current Source Control bit
1 = Analog current source output is grounded
0 = Analog current source output is not grounded

bit 8 **CTTRIG:** CTMU Trigger Control bit
1 = Trigger output is enabled
0 = Trigger output is disabled

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

REGISTER 32-2: CTMUCON2: CTMU CONTROL REGISTER 2

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
EDG1MOD	EDG1POL	EDG1SEL3	EDG1SEL2	EDG1SEL1	EDG1SEL0	EDG2STAT	EDG1STAT
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0
EDG2MOD	EDG2POL	EDG2SEL3	EDG2SEL2	EDG2SEL1	EDG2SEL0	—	—
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 **EDG1MOD:** Edge 1 Edge-Sensitive Select bit
1 = Input is edge-sensitive
0 = Input is level-sensitive

bit 14 **EDG1POL:** Edge 1 Polarity Select bit
1 = Edge 1 is programmed for a positive edge response
0 = Edge 1 is programmed for a negative edge response

bit 13-10 **EDG1SEL<3:0>:** Edge 1 Source Select bits
1111 = Edge 1 source is Comparator 3 output
1110 = Edge 1 source is Comparator 2 output
1101 = Edge 1 source is Comparator 1 output
1100 = Edge 1 source is IC3
1011 = Edge 1 source is IC2
1010 = Edge 1 source is IC1
1001 = Edge 1 source is CTED8
1000 = Edge 1 source is CTED7(1)
0111 = Edge 1 source is CTED6
0110 = Edge 1 source is CTED5
0101 = Edge 1 source is CTED4
0100 = Edge 1 source is CTED3(1)
0011 = Edge 1 source is CTED1
0010 = Edge 1 source is CTED2
0001 = Edge 1 source is OC1
0000 = Edge 1 source is Timer1

bit 9 **EDG2STAT:** Edge 2 Status bit
Indicates the status of Edge 2 and can be written to control current source.
1 = Edge 2 has occurred
0 = Edge 2 has not occurred

bit 8 **EDG1STAT:** Edge 1 Status bit
Indicates the status of Edge 1 and can be written to control current source.
1 = Edge 1 has occurred
0 = Edge 1 has not occurred

bit 7 **EDG2MOD:** Edge 2 Edge-Sensitive Select bit
1 = Input is edge-sensitive
0 = Input is level-sensitive

bit 6 **EDG2POL:** Edge 2 Polarity Select bit
1 = Edge 2 is programmed for a positive edge
0 = Edge 2 is programmed for a positive edge

Note 1: Edge sources, CTED3, CTED7, CTED10 and CTED11, are available in 100-pin devices only.

REGISTER 32-2: CTMUCON2: CTMU CONTROL REGISTER 2 (CONTINUED)

bit 5-2 **EDG2SEL<3:0>:** Edge 2 Source Select bits
1111 = Edge 2 source is Comparator 3 output
1110 = Edge 2 source is Comparator 2 output
1101 = Edge 2 source is Comparator 1 output
1100 = Unimplemented Do not use
1011 = Edge 2 source is IC3
1010 = Edge 2 source is IC2
1001 = Edge 2 source is IC1
1000 = Edge 2 source is CTED13
0111 = Edge 2 source is CTED12
0110 = Edge 2 source is CTED11(1)
0101 = Edge 2 source is CTED10(1)
0100 = Edge 2 source is CTED9
0011 = Edge 2 source is CTED1
0010 = Edge 2 source is CTED2
0001 = Edge 2 source is OC1
0000 = Edge 2 source is Timer1

bit 1-0 **Unimplemented:** Read as '0'

Note 1: Edge sources, CTED3, CTED7, CTED10 and CTED11, are available in 100-pin devices only.

REGISTER 32-3: CTMUICON: CTMU CURRENT CONTROL 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
ITRIM5	ITRIM4	ITRIM3	ITRIM2	ITRIM1	ITRIM0	IRNG1	IRNG0
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 15-10 **ITRIM<5:0>:** Current Source Trim bits
011111 = Maximum positive change from nominal current
011110
.
.
.
000001 = Minimum positive change from nominal current
000000 = Nominal current output specified by IRNG<1:0>
111111 = Minimum negative change from nominal current
.
.
.
100010
100001 = Maximum negative change from nominal current

bit 9-8 **IRNG<1:0>:** Current Source Range Select bits
11 = 100 × Base Current
10 = 10 × Base Current
01 = Base current level (0.55 μA nominal)
00 = 1000 × Base Current

bit 7-0 **Unimplemented:** Read as ‘0’

NOTES:

33.0 HIGH/LOW-VOLTAGE DETECT (HLVD)

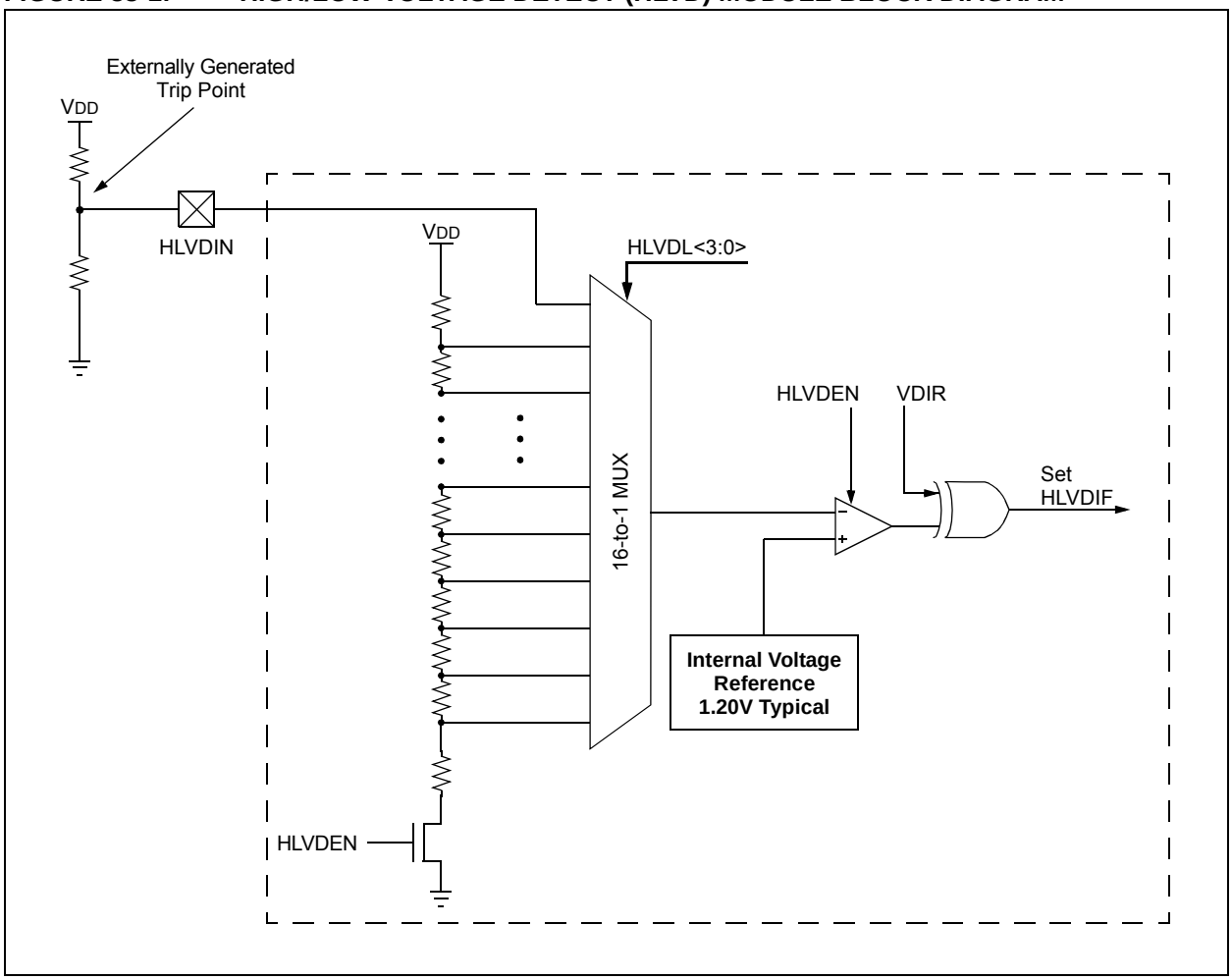
Note:	This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information on the High/Low-Voltage Detect, refer to the *"PIC24F Family Reference Manual"*, **"High-Level Integration with Programmable High/Low-Voltage Detect (HLVD)"** (DS39725).

The High/Low-Voltage Detect (HLVD) module is a programmable circuit that allows the user to specify both the device voltage trip point and the direction of change.

An interrupt flag is set if the device experiences an excursion past the trip point in the direction of change. If the interrupt is enabled, the program execution will branch to the interrupt vector address and the software can then respond to the interrupt.

The HLVD Control register (see Register 33-1) completely controls the operation of the HLVD module. This allows the circuitry to be "turned off" by the user under software control, which minimizes the current consumption for the device.

FIGURE 33-1: HIGH/LOW-VOLTAGE DETECT (HLVD) MODULE BLOCK DIAGRAM



REGISTER 33-1: HLVDCON: HIGH/LOW-VOLTAGE DETECT CONTROL REGISTER

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
HLVDEN	—	LSIDL	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
VDIR	BGVST	IRVST	—	HLVDL3	HLVDL2	HLVDL1	HLVDL0
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 **HLVDEN:** High/Low-Voltage Detect Power Enable bit
1 = HLVD is enabled
0 = HLVD is disabled

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

bit 13 **LSIDL:** HLVD Stop in Idle Mode bit
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode

bit 12-8 **Unimplemented:** Read as '0'

bit 7 **VDIR:** Voltage Change Direction Select bit
1 = Event occurs when voltage equals or exceeds trip point (HLVDL<3:0>)
0 = Event occurs when voltage equals or falls below trip point (HLVDL<3:0>)

bit 6 **BGVST:** Band Gap Voltage Stable Flag bit
1 = Indicates that the band gap voltage is stable
0 = Indicates that the band gap voltage is unstable

bit 5 **IRVST:** Internal Reference Voltage Stable Flag bit
1 = Internal reference voltage is stable; the High-Voltage Detect logic generates the interrupt flag at the specified voltage range
0 = Internal reference voltage is unstable; the High-Voltage Detect logic will not generate the interrupt flag at the specified voltage range and the HLVD interrupt should not be enabled

bit 4 **Unimplemented:** Read as '0'

bit 3-0 **HLVDL<3:0>:** High/Low-Voltage Detection Limit bits
1111 = External analog input is used (input comes from the HLVDIN pin)
1110 = Trip Point 1(1)
1101 = Trip Point 2(1)
1100 = Trip Point 3(1)
.
.
.
0100 = Trip Point 11(1)
00xx = Unused

Note 1: For the actual trip point, see **Section 37.0 "Electrical Characteristics"**.

34.0 SPECIAL FEATURES

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 following sections of the "*PIC24F Family Reference Manual*". The information in this data sheet supersedes the information in the FRMs.

- **"Watchdog Timer (WDT)"** (DS39697)
- **"High-Level Device Integration"** (DS39719)
- **"Programming and Diagnostics"** (DS39716)

PIC24FJ128GC010 family devices include several features intended to maximize application flexibility and reliability, and minimize cost through elimination of external components. These are:

- Flexible Configuration
- Watchdog Timer (WDT)
- Code Protection
- JTAG Boundary Scan Interface
- In-Circuit Serial Programming™
- In-Circuit Emulation

34.1 Configuration Bits

The Configuration bits can be programmed (read as '0'), or left unprogrammed (read as '1'), to select various device configurations. These bits are mapped, starting at program memory location, F80000h. A detailed explanation of the various bit functions is provided in Register 34-1 through Register 34-6.

Note that address, F80000h, is beyond the user program memory space. In fact, it belongs to the configuration memory space (800000h-FFFFFFh), which can only be accessed using table reads and table writes.

34.1.1 CONSIDERATIONS FOR CONFIGURING PIC24FJ128GC010 FAMILY DEVICES

In PIC24FJ128GC010 family devices, the configuration bytes are implemented as volatile memory. This means that configuration data must be programmed each time the device is powered up. Configuration data is stored in the four words at the top of the on-chip program memory space, known as the Flash Configuration Words. Their specific locations are shown in Table 34-1. These are packed representations of the actual device Configuration bits, whose actual locations are distributed among several locations in configuration space. The configuration data is automatically loaded from the Flash Configuration Words to the proper Configuration registers during device Resets.

Note: Configuration data is reloaded on all types of device Resets.

When creating applications for these devices, users should always specifically allocate the location of the Flash Configuration Word for configuration data. This is to make certain that program code is not stored in this address when the code is compiled.

The upper byte of all Flash Configuration Words in program memory should always be '0000 0000'. This makes them appear to be NOP instructions in the remote event that their locations are ever executed by accident. Since Configuration bits are not implemented in the corresponding locations, writing '0's to these locations has no effect on device operation.

Note: Performing a page erase operation on the last page of program memory clears the Flash Configuration Words, enabling code protection as a result. Therefore, users should avoid performing page erase operations on the last page of program memory.

TABLE 34-1: FLASH CONFIGURATION WORD LOCATIONS FOR PIC24FJ128GC010 FAMILY DEVICES

Device	Configuration Word Addresses			
	1	2	3	4
PIC24FJ64GC0XX	ABFEh	ABFCh	ABFAh	ABF8h
PIC24FJ128GC0XX	157FEh	157FCh	157FAh	157F8h

REGISTER 34-1: CW1: FLASH CONFIGURATION WORD 1

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

r-x	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1
r	JTAGEN	GCP	GWRP	$\overline{\text{DEBUG}}$	$\overline{\text{LPCFG}}$	ICS1	ICS0
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
FWDTEN1	FWDTEN0	WINDIS	FWPSA	WDTPS3	WDTPS2	WDTPS1	WDTPS0
bit 7							bit 0

Legend:	r = Reserved bit	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 23-16 **Unimplemented:** Read as '1'

bit 15 **Reserved:** The value is unknown; program as '0'

bit 14 **JTAGEN:** JTAG Port Enable bit
1 = JTAG port is enabled
0 = JTAG port is disabled

bit 13 **GCP:** General Segment Program Memory Code Protection bit
1 = Code protection is disabled
0 = Code protection is enabled for the entire program memory space

bit 12 **GWRP:** General Segment Code Flash Write Protection bit
1 = Writes to program memory are allowed
0 = Writes to program memory are not allowed

bit 11 **$\overline{\text{DEBUG}}$:** Background Debugger Enable bit
1 = Device resets into Operational mode
0 = Device resets into Debug mode

bit 10 **$\overline{\text{LPCFG}}$:** Low-Voltage/Retention Regulator Configuration bit
1 = Low-voltage/retention regulator is always disabled
0 = Low-power, low-voltage/retention regulator is enabled and controlled in firmware by the RETEN bit

bit 9-8 **ICS<1:0>:** Emulator Pin Placement Select bits
11 = Emulator functions are shared with PGEC1/PGED1
10 = Emulator functions are shared with PGEC2/PGED2
01 = Emulator functions are shared with PGEC3/PGED3
00 = Reserved; do not use

bit 7-6 **FWDTEN<1:0>:** Watchdog Timer Configuration bits
11 = WDT is always enabled; SWDTEN bit has no effect
10 = WDT is enabled and controlled in firmware by the SWDTEN bit
01 = WDT is enabled only in Run mode and disabled in Sleep modes; SWDTEN bit is disabled
00 = WDT is disabled; SWDTEN bit is disabled

bit 5 **WINDIS:** Windowed Watchdog Timer Disable bit
1 = Standard Watchdog Timer is enabled
0 = Windowed Watchdog Timer is enabled (FWDTEN<1:0> must not be '00')

REGISTER 34-1: CW1: FLASH CONFIGURATION WORD 1 (CONTINUED)

bit 4 **FWPSA:** WDT Prescaler Ratio Select bit

1 = Prescaler ratio of 1:128
0 = Prescaler ratio of 1:32

bit 3-0 **WDTPS<3:0>:** Watchdog Timer Postscaler Select bits

1111 = 1:32,768
1110 = 1:16,384
1101 = 1:8,192
1100 = 1:4,096
1011 = 1:2,048
1010 = 1:1,024
1001 = 1:512
1000 = 1:256
0111 = 1:128
0110 = 1:64
0101 = 1:32
0100 = 1:16
0011 = 1:8
0010 = 1:4
0001 = 1:2
0000 = 1:1

REGISTER 34-2: CW2: FLASH CONFIGURATION WORD 2

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

R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1
IESO	VBTBOR	WDTCMX	ALTCVREF(1)	ALTADREF(1)	FNOSC2	FNOSC1	FNOSC0
bit 15							bit 8

R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	r-0	R/PO-1	R/PO-1
FCKSM1	FCKSM0	OSCIOFCN	WDTCLK1	WDTCLK0	r	POSCMD1	POSCMD0
bit 7							bit 0

Legend:	r = Reserved bit	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 23-16 **Unimplemented:** Read as '1'

bit 15 **IESO:** Internal External Switchover bit
1 = IESO mode (Two-Speed Start-up) is enabled
0 = IESO mode (Two-Speed Start-up) is disabled

bit 14 **VBTBOR:** VBAT BOR Enable bit
1 = VBAT BOR is enabled
0 = VBAT BOR is disabled

bit 13 **WDTCMX:** WDT Clock Multiplex Control bit
1 = Enables WDT clock multiplexing
0 = Disables clock multiplexing

bit 12 **ALTCVREF:** External CVREF+/CVREF- Location Select bit(1)
1 = CVREF+/CVREF- are mapped to RA9/RA10, respectively
0 = CVREF+/CVREF- are mapped to RB0/RB1, respectively

bit 11 **ALTADREF:** External AVREF+/AVREF- Location Select bit(1)
1 = AVREF+/AVREF- are mapped to RA9/RA10, respectively
0 = AVREF+/AVREF- are mapped to RB0/RB1, respectively

bit 10-8 **FNOSC<2:0>:** Initial Oscillator Select bits
111 = Fast RC Oscillator with Postscaler (FRCDIV)
110 = Reserved
101 = Low-Power RC Oscillator (LPRC)
100 = Secondary Oscillator (SOSC)
011 = Primary Oscillator with PLL module (XTPLL, HSPLL, ECPLL)
010 = Primary Oscillator (XT, HS, EC)
001 = Fast RC Oscillator with Postscaler and PLL module (FRCPLL)
000 = Fast RC Oscillator (FRC)

bit 7-6 **FCKSM<1:0>:** Clock Switching and Fail-Safe Clock Monitor Configuration bits
1x = Clock switching and Fail-Safe Clock Monitor are disabled
01 = Clock switching is enabled, Fail-Safe Clock Monitor is disabled
00 = Clock switching is enabled, Fail-Safe Clock Monitor is enabled

Note 1: These bits should be treated as reserved on the 64-pin devices (PIC24FJ64GC006 and PIC24FJ128GC006) and should always be programmed to '0'. The AVREF+/CVREF+ and AVREF-/CVREF- functions are located on RB0 and RB1 on these devices.

2: The 31 kHz FRC source is used when a Windowed WDT mode is selected and the LPRC is not being used as the system clock. The LPRC is used when the device is in Sleep mode and in all other cases.

REGISTER 34-2: CW2: FLASH CONFIGURATION WORD 2 (CONTINUED)

bit 5 **OSCIOFCN:** OSCO Pin Configuration bit

If POSCMD<1:0> = 11 or 00:
1 = OSCO/CLKO/RC15 functions as CLKO (FOSC/2)
0 = OSCO/CLKO/RC15 functions as port I/O (RC15)

If POSCMD<1:0> = 10 or 01:
OSCIOFCN has no effect on OSCO/CLKO/RC15.

bit 4-3 **WDTCLK<1:0>:** WDT Clock Source Select bits

When WDTCMX = 1:
11 = LPRC
10 = Either the 31 kHz FRC source or LPRC, depending on device configuration(2)
01 = SOSC input
00 = System clock when active, LPRC while in Sleep mode

When WDTCMX = 0:
LPRC is always the WDT clock source.

bit 2 **Reserved:** Configure as '0'

bit 1-0 **POSCMD<1:0>:** Primary Oscillator Configuration bits

11 = Primary Oscillator mode is disabled
10 = HS Oscillator mode is selected (HS mode is used if crystal ≥ 10 MHz)
01 = XT Oscillator mode is selected (XT mode is used if crystal < 10 MHz)
00 = EC Oscillator mode is selected

Note 1: These bits should be treated as reserved on the 64-pin devices (PIC24FJ64GC006 and PIC24FJ128GC006) and should always be programmed to '0'. The AVREF+/CVREF+ and AVREF-/CVREF- functions are located on RB0 and RB1 on these devices.

2: The 31 kHz FRC source is used when a Windowed WDT mode is selected and the LPRC is not being used as the system clock. The LPRC is used when the device is in Sleep mode and in all other cases.

REGISTER 34-3: CW3: FLASH CONFIGURATION WORD 3

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

R/PO-1	R/PO-1	R/PO-1	R/PO-1	r-1	R/PO-1	R/PO-1	R/PO-1
WPEND	WPCFG	WPDIS	BOREN	r	WDTWIN1	WDTWIN0	SOSCSEL
bit 15							bit 8

r-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1
r	WPFP6(3)	WPFP5	WPFP4	WPFP3	WPFP2	WPFP1	WPFP0
bit 7							bit 0

Legend:	PO = Program Once bit	r = Reserved 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 23-16 **Unimplemented:** Read as '1'

bit 15 **WPEND:** Segment Write Protection End Page Select bit
- 1 = Protected program memory segment upper boundary is at the last page of program memory; the lower boundary is the code page specified by WPFP<6:0>
- 0 = Protected program memory segment lower boundary is at the bottom of the program memory (000000h); upper boundary is the code page specified by WPFP<6:0>

bit 14 **WPCFG:** Configuration Word Code Page Write Protection Select bit
- 1 = Last page (at the top of program memory) and Flash Configuration Words are not write-protected(1)
- 0 = Last page and Flash Configuration Words are write-protected provided WPDIS = 0

bit 13 **WPDIS:** Segment Write Protection Disable bit
- 1 = Segmented program memory write protection is disabled
- 0 = Segmented program memory write protection is enabled; protected segment is defined by the WPEND, WPCFG and WPFPx Configuration bits

bit 12 **BOREN:** Brown-out Reset Enable bit
- 1 = BOR is enabled (all modes except Deep Sleep)
- 0 = BOR is disabled

bit 11 **Reserved:** Always maintain as '1'

bit 10-9 **WDTWIN<1:0>:** Watchdog Timer Window Width Select bits
- 11 = 25%
- 10 = 37.5%
- 01 = 50%
- 00 = 75%

bit 8 **SOSCSEL:** SOSC Selection bit
- 1 = SOSC circuit is selected
- 0 = Digital (SCLKI) mode(2)

bit 7 **Reserved:** Always maintain as '1'

Note 1: Regardless of WPCFG status, if WPEND = 1 or if WPFP<6:0> bits correspond to the Configuration Word page, the Configuration Word page is protected.

2: Ensure that the SCLKI pin is made a digital input while using this configuration (see Table 11-1).

3: For the 64K devices (PIC24FJ64GC0XX), maintain WPFP6 as '0'.

REGISTER 34-3: CW3: FLASH CONFIGURATION WORD 3 (CONTINUED)

bit 6-0 **WPFP<6:0>:** Write-Protected Code Segment Boundary Page bits(3)
Designates the 512 instruction words page boundary of the protected code segment.
If WPEND = 1:
Specifies the lower page boundary of the code-protected segment; the last page being the last implemented page in the device.
If WPEND = 0:
Specifies the upper page boundary of the code-protected segment; Page 0 being the lower boundary.

Note 1: Regardless of WPCFG status, if WPEND = 1 or if WPFP<6:0> bits correspond to the Configuration Word page, the Configuration Word page is protected.

2: Ensure that the SCLKI pin is made a digital input while using this configuration (see Table 11-1).

3: For the 64K devices (PIC24FJ64GC0XX), maintain WPFP6 as '0'.

REGISTER 34-4: CW4: FLASH CONFIGURATION WORD 4

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

R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1
IOL1WAY	I2C2SEL	PLLDIV3	PLLDIV2	PLLDIV1	PLLDIV0	RTCBAT	DSSWEN
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
DSWDTEN	DSBOREN	DSWDTOSC	DSWDPS4	DSWDPS3	DSWDPS2	DSWDPS1	DSWDPS0
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 23-16 **Unimplemented:** Read as '1'

bit 15 **IOL1WAY:** IOLOCK One-Way Set Enable bit
1 = The IOLOCK bit (OSCCON<6>) can be set once, provided the unlock sequence has been completed. Once set, the Peripheral Pin Select registers cannot be written to a second time.
0 = The IOLOCK bit can be set and cleared as needed, provided the unlock sequence has been completed

bit 14 **I2C2SEL:** Alternate I2C2 Location Select bit
For PIC24FJXXXGC010 Devices:
1 = I2C2 functions; SCL2 and SDA2 are mapped to RA2 and RA3, respectively
0 = I2C2 functions; SCL2 and SDA2 are mapped to RF5 and RF4, respectively
For PIC24FJXXXGC006 Devices:
Reserved, maintain as '1'.

bit 13-10 **PLLDIV<3:0:>** USB 96 MHz PLL Prescaler Select bits
1111 = PLL disabled
1110
.... = Reserved, do not use
1000
0111 = Oscillator input divided by 12 (48 MHz input)
0110 = Oscillator input divided by 8 (32 MHz input)
0101 = Oscillator input divided by 6 (24 MHz input)
0100 = Oscillator input divided by 5 (20 MHz input)
0011 = Oscillator input divided by 4 (16 MHz input)
0010 = Oscillator input divided by 3 (12 MHz input)
0001 = Oscillator input divided by 2 (8 MHz input)
0000 = Oscillator input used directly (4 MHz input)

bit 9 **RTCBAT:** V_{BAT} RTCC Operation Select bit
1 = RTCC operation continues when the device is in V_{BAT} mode
0 = RTCC operation stops when the device is in V_{BAT} mode

bit 8 **DSSWEN:** Deep Sleep Software Control Select bit
1 = Deep Sleep operation is enabled and controlled by the DSEN bit
0 = Deep Sleep operation is disabled

bit 7 **DSWDTEN:** Deep Sleep Watchdog Timer Enable bit
1 = Deep Sleep WDT is enabled
0 = Deep Sleep WDT is disabled

REGISTER 34-4: CW4: FLASH CONFIGURATION WORD 4 (CONTINUED)

bit 6 **DSBOREN:** Deep Sleep Brown-out Reset Enable bit

1 = BOR is enabled in Deep Sleep mode
0 = BOR is disabled in Deep Sleep mode (remains active in other Sleep modes)

bit 5 **DSWDTOSC:** Deep Sleep Watchdog Timer Clock Select bit

1 = Clock source is LPRC
0 = Clock source is SOSC

bit 4-0 **DSWDPS<4:0>:** Deep Sleep Watchdog Timer Postscaler Select bits

11111 = 1:68,719,476,736 (25.7 days)
11110 = 1:34,359,738,368(12.8 days)
11101 = 1:17,179,869,184 (6.4 days)
11100 = 1:8,589,934592 (77.0 hours)
11011 = 1:4,294,967,296 (38.5 hours)
11010 = 1:2,147,483,648 (19.2 hours)
11001 = 1:1,073,741,824 (9.6 hours)
11000 = 1:536,870,912 (4.8 hours)
10111 = 1:268,435,456 (2.4 hours)
10110 = 1:134,217,728 (72.2 minutes)
10101 = 1:67,108,864 (36.1 minutes)
10100 = 1:33,554,432 (18.0 minutes)
10011 = 1:16,777,216 (9.0 minutes)
10010 = 1:8,388,608 (4.5 minutes)
10001 = 1:4,194,304 (135.3s)
10000 = 1:2,097,152 (67.7s)
01111 = 1:1,048,576 (33.825s)
01110 = 1:524,288 (16.912s)
01101 = 1:262,114 (8.456s)
01100 = 1:131,072 (4.228s)
01011 = 1:65,536 (2.114s)
01010 = 1:32,768 (1.057s)
01001 = 1:16,384 (528.5 ms)
01000 = 1:8,192 (264.3 ms)
00111 = 1:4,096 (132.1 ms)
00110 = 1:2,048 (66.1 ms)
00101 = 1:1,024 (33 ms)
00100 = 1:512 (16.5 ms)
00011 = 1:256 (8.3 ms)
00010 = 1:128 (4.1 ms)
00001 = 1:64 (2.1 ms)
00000 = 1:32 (1 ms)

REGISTER 34-5: DEVID: DEVICE ID REGISTER

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

R	R	R	R	R	R	R	R
FAMID7	FAMID6	FAMID5	FAMID4	FAMID3	FAMID2	FAMID1	FAMID0
bit 15							bit 8

R	R	R	R	R	R	R	R
DEV7	DEV6	DEV5	DEV4	DEV3	DEV2	DEV1	DEV0
bit 7							bit 0

Legend:	R = Readable bit	U = Unimplemented bit

bit 23-16 **Unimplemented:** Read as '1'

bit 15-8 **FAMID<7:0>:** Device Family Identifier bits

0100 1000 = PIC24FJ128GC010 family

bit 7-0 **DEV<7:0>:** Individual Device Identifier bits

1000 1000 = PIC24FJ64GC006
1000 1001 = PIC24FJ128GC006
1000 0100 = PIC24FJ64GC010
1000 0101 = PIC24FJ128GC010

REGISTER 34-6: DEVREV: DEVICE REVISION REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 23							bit 16

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	R	R	R	R
—	—	—	—	REV3	REV2	REV1	REV0
bit 7							bit 0

Legend:	R = Readable bit	U = Unimplemented bit

bit 23-4 **Unimplemented:** Read as '0'

bit 3-0 **REV<3:0>:** Device Revision Identifier bits

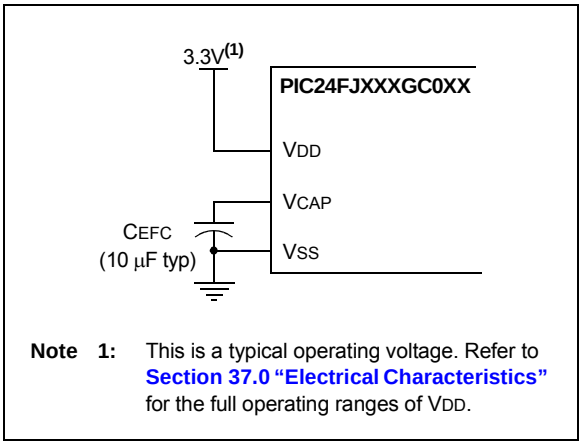
34.2 On-Chip Voltage Regulator

All PIC24FJ128GC010 family devices power their core digital logic at a nominal 1.8V. This may create an issue for designs that are required to operate at a higher typical voltage, such as 3.3V. To simplify system design, all devices in the PIC24FJ128GC010 family incorporate an on-chip regulator that allows the device to run its core logic from VDD.

This regulator is always enabled. It provides a constant voltage (1.8V nominal) to the digital core logic, from a VDD of 2.0V all the way up to the device's VDDMAX. It does not have the capability to boost VDD levels. In order to prevent "brown-out" conditions when the voltage drops too low for the regulator, the Brown-out Reset occurs. Then the regulator output follows VDD with a typical voltage drop of 300 mV.

A low-ESR capacitor (such as ceramic) must be connected to the VCAP pin (Figure 34-1). This helps to maintain the stability of the regulator. The recommended value for the filter capacitor (CEFC) is provided in **Section 37.1 "DC Characteristics"**.

FIGURE 34-1: CONNECTIONS FOR THE ON-CHIP REGULATOR



Note 1: This is a typical operating voltage. Refer to **Section 37.0 "Electrical Characteristics"** for the full operating ranges of VDD.

34.2.1 ON-CHIP REGULATOR AND POR

The voltage regulator requires a small amount of time to transition from a disabled or standby state into normal operating mode. During this time, designated as TVREG, code execution is disabled. TVREG is applied every time the device resumes operation after any power-down, including Sleep mode. TVREG is determined by the status of the PMSLP bit (RCON<8>). Refer to **Section 37.0 "Electrical Characteristics"** for more information on TVREG.

Note: For more information, see **Section 37.0 "Electrical Characteristics"**. The Information in this data sheet supersedes the information in the "*PIC24F Family Reference Manual*".

34.2.2 VOLTAGE REGULATOR STANDBY MODE

The on-chip regulator always consumes a small incremental amount of current over IDD/IPD, including when the device is in Sleep mode, even though the core digital logic does not require power. To provide additional savings in applications where power resources are critical, the regulator can be made to enter Standby mode on its own whenever the device goes into Sleep mode. This feature is controlled by the PMSLP bit (RCON<8>). Clearing the PMSLP bit enables the Standby mode. When waking up from Standby mode, the regulator needs to wait for TVREG to expire before wake-up.

34.2.3 LOW-VOLTAGE/RETENTION REGULATOR

When power-saving modes, such as Sleep and Deep Sleep are used, PIC24FJ128GC010 family devices may use a separate low-power, low-voltage/retention regulator to power critical circuits. This regulator, which operates at 1.2V nominal, maintains power to data RAM and the RTCC while all other core digital logic is powered down. It operates only in Sleep, Deep Sleep and VBAT modes.

The low-voltage/retention regulator is described in more detail in **Section 10.1.3 "Low-Voltage/Retention Regulator"**.

34.3 Watchdog Timer (WDT)

For PIC24FJ128GC010 family devices, the WDT is driven by the LPRC oscillator. When the WDT is enabled, the clock source is also enabled.

The nominal WDT clock source from LPRC is 31 kHz. This feeds a prescaler that can be configured for either 5-bit (divide-by-32) or 7-bit (divide-by-128) operation. The prescaler is set by the FWPSA Configuration bit. With a 31 kHz input, the prescaler yields a nominal WDT Time-out period (TWDT) of 1 ms in 5-bit mode or 4 ms in 7-bit mode.

A variable postscaler divides down the WDT prescaler output and allows for a wide range of time-out periods. The postscaler is controlled by the WDTPS<3:0> Configuration bits (CW1<3:0>), which allows the selection of a total of 16 settings, from 1:1 to 1:32,768. Using the prescaler and postscaler time-out periods, ranges from 1 ms to 131 seconds can be achieved.

The WDT, prescaler and postscaler are reset:

- On any device Reset
- On the completion of a clock switch, whether invoked by software (i.e., setting the OSWEN bit after changing the NOSCx bits) or by hardware (i.e., Fail-Safe Clock Monitor)
- When a PWRSAV instruction is executed (i.e., Sleep or Idle mode is entered)
- When the device exits Sleep or Idle mode to resume normal operation
- By a CLRWDT instruction during normal execution

If the WDT is enabled, it will continue to run during Sleep or Idle modes. When the WDT time-out occurs, the device will wake the device and code execution will continue from where the PWRSAV instruction was executed. The corresponding SLEEP or IDLE (RCON<3:2>) bit will need to be cleared in software after the device wakes up.

The WDT Flag bit, WDTO (RCON<4>), is not automatically cleared following a WDT time-out. To detect subsequent WDT events, the flag must be cleared in software.

Note: The CLRWDT and PWRSAV instructions clear the prescaler and postscaler counts when executed.

34.3.1 WINDOWED OPERATION

The Watchdog Timer has an optional Fixed Window mode of operation. In this Windowed mode, CLRWDT instructions can only reset the WDT during the last 1/4 of the programmed WDT period. A CLRWDT instruction executed before that window causes a WDT Reset, similar to a WDT time-out.

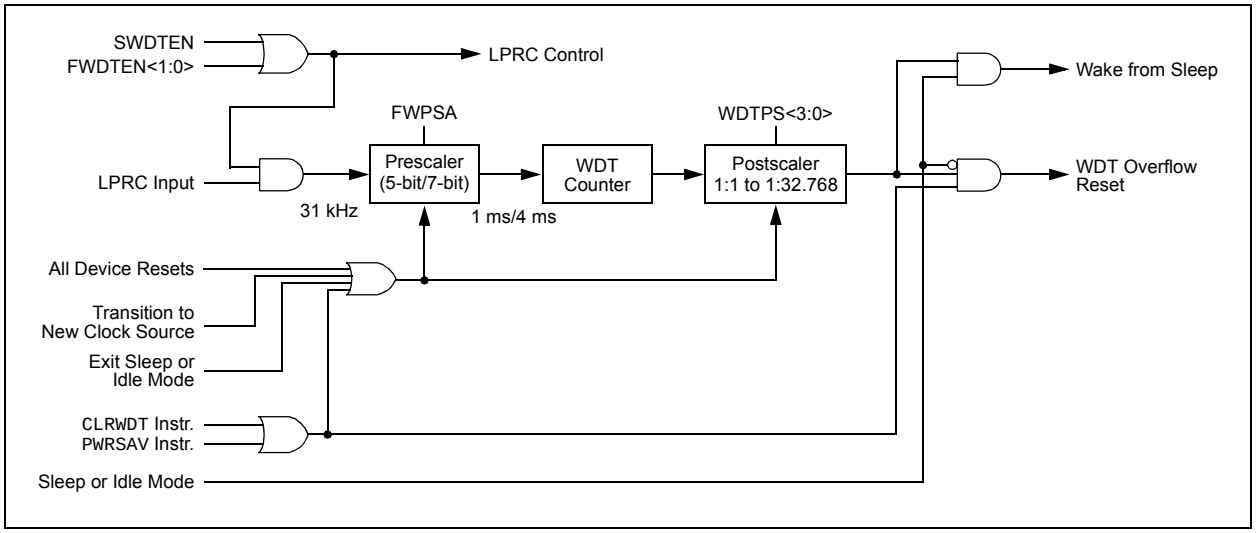
Windowed WDT mode is enabled by programming the WINDIS Configuration bit (CW1<5>) to '0'.

34.3.2 CONTROL REGISTER

The WDT is enabled or disabled by the FWDTEN<1:0> Configuration bits. When the Configuration bits, FWDTEN<1:0> = 11, the WDT is always enabled.

The WDT can be optionally controlled in software when the Configuration bits, FWDTEN<1:0> = 10. When FWDTEN<1:0> = 00, the Watchdog Timer is always disabled. The WDT is enabled in software by setting the SWDTEN control bit (RCON<5>). The SWDTEN control bit is cleared on any device Reset. The software WDT option allows the user to enable the WDT for critical code segments and disable the WDT during non-critical segments for maximum power savings.

FIGURE 34-2: WDT BLOCK DIAGRAM



34.4 Program Verification and Code Protection

PIC24FJ128GC010 family devices provide two complimentary methods to protect application code from overwrites and erasures. These also help to protect the device from inadvertent configuration changes during run time.

34.4.1 GENERAL SEGMENT PROTECTION

For all devices in the PIC24FJ128GC010 family, the on-chip program memory space is treated as a single block, known as the General Segment (GS). Code protection for this block is controlled by one Configuration bit, GCP. This bit inhibits external reads and writes to the program memory space. It has no direct effect in normal execution mode.

Write protection is controlled by the GWRP bit in the Configuration Word. When GWRP is programmed to '0', internal write and erase operations to program memory are blocked.

34.4.2 CODE SEGMENT PROTECTION

In addition to global General Segment protection, a separate subrange of the program memory space can be individually protected against writes and erases. This area can be used for many purposes where a separate block of write and erase-protected code is needed, such as bootloader applications. Unlike common boot block implementations, the specially protected segment in the PIC24FJ128GC010 family devices can be located by the user anywhere in the program space and configured in a wide range of sizes.

Code segment protection provides an added level of protection to a designated area of program memory by disabling the NVM safety interlock whenever a write or erase address falls within a specified range. It does not override General Segment protection controlled by the GCP or GWRP bit. For example, if GCP and GWRP are enabled, enabling segmented code protection for the bottom half of program memory does not undo General Segment protection for the top half.

The size and type of protection for the segmented code range are configured by the WPFPx, WPEND, WPCFG and WPDIS bits in Configuration Word 3. Code segment protection is enabled by programming the WPDIS bit (= 0). The WPFPx bits specify the size of the segment to be protected by specifying the 512-word code page that is the start or end of the protected segment. The specified region is inclusive, therefore, this page will also be protected.

The WPEND bit determines if the protected segment uses the top or bottom of the program space as a boundary. Programming WPEND (= 0) sets the bottom of program memory (000000h) as the lower boundary of the protected segment. Leaving WPEND unprogrammed (= 1) protects the specified page through the last page of implemented program memory, including the Configuration Word locations.

A separate bit, WPCFG, is used to protect the last page of program space, including the Flash Configuration Words. Programming WPCFG (= 0) protects the last page in addition to the pages selected by the WPEND and WPFP<6:0> bits' setting. This is useful in circumstances where write protection is needed for both the code segment in the bottom of the memory and the Flash Configuration Words.

The various options for segment code protection are shown in Table 34-2.

TABLE 34-2: CODE SEGMENT PROTECTION CONFIGURATION OPTIONS

Segment Configuration Bits			Write/Erase Protection of Code Segment
WPDIS	WPEND	WPCFG	
1	x	x	No additional protection is enabled; all program memory protection is configured by GCP and GWRP.
0	1	x	Addresses from the first address of the code page are defined by WPFP<6:0> through the end of implemented program memory (inclusive); erase/write-protected, including Flash Configuration Words.
0	0	1	Address, 000000h through the last address of the code page, is defined by WPFP<6:0> (inclusive); write/erase protected.
0	0	0	Address, 000000h through the last address of code page, is defined by WPFP<6:0> (inclusive); erase/write-protected and the last page, including Flash Configuration Words, are erase/write-protected.

34.4.3 CONFIGURATION REGISTER PROTECTION

The Configuration registers are protected against inadvertent or unwanted changes or reads in two ways. The primary protection method is the same as that of the RP registers – shadow registers contain a complimentary value which is constantly compared with the actual value.

To safeguard against unpredictable events, Configuration bit changes resulting from individual cell level disruptions (such as ESD events) will cause a parity error and trigger a device Reset.

The data for the Configuration registers is derived from the Flash Configuration Words in program memory. When the GCP bit is set, the source data for device configuration is also protected as a consequence. Even if General Segment protection is not enabled, the device configuration can be protected by using the appropriate code segment protection setting.

34.5 JTAG Interface

PIC24FJ128GC010 family devices implement a JTAG interface, which supports boundary scan device testing.

34.6 In-Circuit Serial Programming

PIC24FJ128GC010 family microcontrollers can be serially programmed while in the end application circuit. This is simply done with two lines for clock (PGECx) and data (PGEDx), and three other lines for power (V_{DD}), ground (V_{SS}) and $\overline{MCLR}$. This allows customers to manufacture boards with unprogrammed devices and then program the microcontroller just before shipping the product. This also allows the most recent firmware or a custom firmware to be programmed.

34.7 In-Circuit Debugger

When MPLAB® ICD 3 is selected as a debugger, the in-circuit debugging functionality is enabled. This function allows simple debugging functions when used with MPLAB IDE. Debugging functionality is controlled through the PGECx (Emulation/Debug Clock) and PGEDx (Emulation/Debug Data) pins.

To use the in-circuit debugger function of the device, the design must implement ICSP connections to $\overline{MCLR}$, V_{DD}, V_{SS} and the PGECx/PGEDx pin pair, designated by the ICSx Configuration bits. In addition, when the feature is enabled, some of the resources are not available for general use. These resources include the first 80 bytes of data RAM and two I/O pins.

35.0 DEVELOPMENT SUPPORT

The PIC® microcontrollers (MCU) and dsPIC® digital signal controllers (DSC) are supported with a full range of software and hardware development tools:

- Integrated Development Environment
  - MPLAB® X IDE Software
- Compilers/Assemblers/Linkers
  - MPLAB XC Compiler
  - MPASM™ Assembler
  - MPLINK™ Object Linker/ MPLIB™ Object Librarian
  - MPLAB Assembler/Linker/Librarian for Various Device Families
- Simulators
  - MPLAB X SIM Software Simulator
- Emulators
  - MPLAB REAL ICE™ In-Circuit Emulator
- In-Circuit Debuggers/Programmers
  - MPLAB ICD 3
  - PICkit™ 3
- Device Programmers
  - MPLAB PM3 Device Programmer
- Low-Cost Demonstration/Development Boards, Evaluation Kits and Starter Kits
- Third-party development tools

35.1 MPLAB X Integrated Development Environment Software

The MPLAB X IDE is a single, unified graphical user interface for Microchip and third-party software, and hardware development tool that runs on Windows®, Linux and Mac OS® X. Based on the NetBeans IDE, MPLAB X IDE is an entirely new IDE with a host of free software components and plug-ins for high-performance application development and debugging. Moving between tools and upgrading from software simulators to hardware debugging and programming tools is simple with the seamless user interface.

With complete project management, visual call graphs, a configurable watch window and a feature-rich editor that includes code completion and context menus, MPLAB X IDE is flexible and friendly enough for new users. With the ability to support multiple tools on multiple projects with simultaneous debugging, MPLAB X IDE is also suitable for the needs of experienced users.

Feature-Rich Editor:

- Color syntax highlighting
- Smart code completion makes suggestions and provides hints as you type
- Automatic code formatting based on user-defined rules
- Live parsing

User-Friendly, Customizable Interface:

- Fully customizable interface: toolbars, toolbar buttons, windows, window placement, etc.
- Call graph window

Project-Based Workspaces:

- Multiple projects
- Multiple tools
- Multiple configurations
- Simultaneous debugging sessions

File History and Bug Tracking:

- Local file history feature
- Built-in support for Bugzilla issue tracker

35.2 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

35.3 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

35.4 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

35.5 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

35.6 MPLAB X SIM Software Simulator

The MPLAB X SIM Software Simulator allows code development in a PC-hosted environment by simulating the PIC MCUs and dsPIC DSCs on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a comprehensive stimulus controller. Registers can be logged to files for further run-time analysis. The trace buffer and logic analyzer display extend the power of the simulator to record and track program execution, actions on I/O, most peripherals and internal registers.

The MPLAB X SIM Software Simulator fully supports symbolic debugging using the MPLAB XC Compilers, and the MPASM and MPLAB Assemblers. The software simulator offers the flexibility to develop and debug code outside of the hardware laboratory environment, making it an excellent, economical software development tool.

35.7 MPLAB REAL ICE In-Circuit Emulator System

The MPLAB REAL ICE In-Circuit Emulator System is Microchip's next generation high-speed emulator for Microchip Flash DSC and MCU devices. It debugs and programs all 8, 16 and 32-bit MCU, and DSC devices with the easy-to-use, powerful graphical user interface of the MPLAB X IDE.

The emulator is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with either a connector compatible with in-circuit debugger systems (RJ-11) or with the new high-speed, noise tolerant, Low-Voltage Differential Signal (LVDS) interconnection (CAT5).

The emulator is field upgradable through future firmware downloads in MPLAB X IDE. MPLAB REAL ICE offers significant advantages over competitive emulators including full-speed emulation, run-time variable watches, trace analysis, complex breakpoints, logic probes, a ruggedized probe interface and long (up to three meters) interconnection cables.

35.8 MPLAB ICD 3 In-Circuit Debugger System

The MPLAB ICD 3 In-Circuit Debugger System is Microchip's most cost-effective, high-speed hardware debugger/programmer for Microchip Flash DSC and MCU devices. It debugs and programs PIC Flash microcontrollers and dsPIC DSCs with the powerful, yet easy-to-use graphical user interface of the MPLAB IDE.

The MPLAB ICD 3 In-Circuit Debugger probe is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with a connector compatible with the MPLAB ICD 2 or MPLAB REAL ICE systems (RJ-11). MPLAB ICD 3 supports all MPLAB ICD 2 headers.

35.9 PICkit 3 In-Circuit Debugger/ Programmer

The MPLAB PICkit 3 allows debugging and programming of PIC and dsPIC Flash microcontrollers at a most affordable price point using the powerful graphical user interface of the MPLAB IDE. The MPLAB PICkit 3 is connected to the design engineer's PC using a full-speed USB interface and can be connected to the target via a Microchip debug (RJ-11) connector (compatible with MPLAB ICD 3 and MPLAB REAL ICE). The connector uses two device I/O pins and the Reset line to implement in-circuit debugging and In-Circuit Serial Programming™ (ICSP™).

35.10 MPLAB PM3 Device Programmer

The MPLAB PM3 Device Programmer is a universal, CE compliant device programmer with programmable voltage verification at V_{DDMIN} and V_{DDMAX} for maximum reliability. It features a large LCD display (128 x 64) for menus and error messages, and a modular, detachable socket assembly to support various package types. The ICSP cable assembly is included as a standard item. In Stand-Alone mode, the MPLAB PM3 Device Programmer can read, verify and program PIC devices without a PC connection. It can also set code protection in this mode. The MPLAB PM3 connects to the host PC via an RS-232 or USB cable. The MPLAB PM3 has high-speed communications and optimized algorithms for quick programming of large memory devices, and incorporates an MMC card for file storage and data applications.

35.11 Demonstration/Development Boards, Evaluation Kits, and Starter Kits

A wide variety of demonstration, development and evaluation boards for various PIC MCUs and dsPIC DSCs allows quick application development on fully functional systems. Most boards include prototyping areas for adding custom circuitry and provide application firmware and source code for examination and modification.

The boards support a variety of features, including LEDs, temperature sensors, switches, speakers, RS-232 interfaces, LCD displays, potentiometers and additional EEPROM memory.

The demonstration and development boards can be used in teaching environments, for prototyping custom circuits and for learning about various microcontroller applications.

In addition to the PICDEM™ and dsPICDEM™ demonstration/development board series of circuits, Microchip has a line of evaluation kits and demonstration software for analog filter design, KEELOQ® security ICs, CAN, IrDA®, PowerSmart battery management, SEEVAL® evaluation system, Sigma-Delta ADC, flow rate sensing, plus many more.

Also available are starter kits that contain everything needed to experience the specified device. This usually includes a single application and debug capability, all on one board.

Check the Microchip web page (www.microchip.com) for the complete list of demonstration, development and evaluation kits.

35.12 Third-Party Development Tools

Microchip also offers a great collection of tools from third-party vendors. These tools are carefully selected to offer good value and unique functionality.

- Device Programmers and Gang Programmers from companies, such as SoftLog and CCS
- Software Tools from companies, such as Gimpel and Trace Systems
- Protocol Analyzers from companies, such as Saleae and Total Phase
- Demonstration Boards from companies, such as MikroElektronika, Digilent® and Olimex
- Embedded Ethernet Solutions from companies, such as EZ Web Lynx, WIZnet and IPLogika®

36.0 INSTRUCTION SET SUMMARY

Note: This chapter is a brief summary of the PIC24F Instruction Set Architecture (ISA) and is not intended to be a comprehensive reference source.

The PIC24F instruction set adds many enhancements to the previous PIC® MCU instruction sets, while maintaining an easy migration from previous PIC MCU instruction sets. Most instructions are a single program memory word. Only three instructions require two program memory locations.

Each single-word instruction is a 24-bit word divided into an 8-bit opcode, which specifies the instruction type and one or more operands, which further specify the operation of the instruction. The instruction set is highly orthogonal and is grouped into four basic categories:

- Word or byte-oriented operations
- Bit-oriented operations
- Literal operations
- Control operations

Table 36-1 shows the general symbols used in describing the instructions. The PIC24F instruction set summary in Table 36-2 lists all the instructions, along with the status flags affected by each instruction.

Most word or byte-oriented W register instructions (including barrel shift instructions) have three operands:

- The first source operand, which is typically a register, ‘Wb’, without any address modifier
- The second source operand, which is typically a register, ‘Ws’, with or without an address modifier
- The destination of the result, which is typically a register, ‘Wd’, with or without an address modifier

However, word or byte-oriented file register instructions have two operands:

- The file register specified by the value, ‘f’
- The destination, which could either be the file register, ‘f’, or the W0 register, which is denoted as ‘WREG’

Most bit-oriented instructions (including simple rotate/shift instructions) have two operands:

- The W register (with or without an address modifier) or file register (specified by the value of ‘Ws’ or ‘f’)
- The bit in the W register or file register (specified by a literal value or indirectly by the contents of register, ‘Wb’)

The literal instructions that involve data movement may use some of the following operands:

- A literal value to be loaded into a W register or file register (specified by the value of ‘k’)
- The W register or file register where the literal value is to be loaded (specified by ‘Wb’ or ‘f’)

However, literal instructions that involve arithmetic or logical operations use some of the following operands:

- The first source operand, which is a register, ‘Wb’, without any address modifier
- The second source operand, which is a literal value
- The destination of the result (only if not the same as the first source operand), which is typically a register, ‘Wd’, with or without an address modifier

The control instructions may use some of the following operands:

- A program memory address
- The mode of the table read and table write instructions

All instructions are a single word, except for certain double-word instructions, which were made double-word instructions so that all the required information is available in these 48 bits. In the second word, the 8 MSbs are ‘0’s. If this second word is executed as an instruction (by itself), it will execute as a NOP.

Most single-word instructions are executed in a single instruction cycle, unless a conditional test is true or the Program Counter is changed as a result of the instruction. In these cases, the execution takes two instruction cycles, with the additional instruction cycle(s) executed as a NOP. Notable exceptions are the BRA (unconditional/computed branch), indirect CALL/GOTO, all table reads and writes, and RETURN/RETFIE instructions, which are single-word instructions but take two or three cycles.

Certain instructions that involve skipping over the subsequent instruction require either two or three cycles if the skip is performed, depending on whether the instruction being skipped is a single-word or two-word instruction. Moreover, double-word moves require two cycles. The double-word instructions execute in two instruction cycles.

TABLE 36-1: SYMBOLS USED IN OPCODE DESCRIPTIONS

Field	Description
#text	Means literal defined by "text"
(text)	Means "content of text"
[text]	Means "the location addressed by text"
{ }	Optional field or operation
<n:m>	Register bit field
.b	Byte mode selection
.d	Double-Word mode selection
.S	Shadow register select
.w	Word mode selection (default)
bit4	4-bit Bit Selection field (used in word addressed instructions) ∈ {0...15}
C, DC, N, OV, Z	MCU Status bits: Carry, Digit Carry, Negative, Overflow, Sticky Zero
Expr	Absolute address, label or expression (resolved by the linker)
f	File register address ∈ {0000h...1FFFh}
lit1	1-bit unsigned literal ∈ {0,1}
lit4	4-bit unsigned literal ∈ {0...15}
lit5	5-bit unsigned literal ∈ {0...31}
lit8	8-bit unsigned literal ∈ {0...255}
lit10	10-bit unsigned literal ∈ {0...255} for Byte mode, {0:1023} for Word mode
lit14	14-bit unsigned literal ∈ {0...16383}
lit16	16-bit unsigned literal ∈ {0...65535}
lit23	23-bit unsigned literal ∈ {0...8388607}; LSB must be '0'
None	Field does not require an entry, may be blank
PC	Program Counter
Slit10	10-bit signed literal ∈ {-512...511}
Slit16	16-bit signed literal ∈ {-32768...32767}
Slit6	6-bit signed literal ∈ {-16...16}
Wb	Base W register ∈ {W0..W15}
Wd	Destination W register ∈ { Wd, [Wd], [Wd++], [Wd--], [++Wd], [--Wd] }
Wdo	Destination W register ∈ { Wnd, [Wnd], [Wnd++], [Wnd--], [++Wnd], [--Wnd], [Wnd+Wb] }
Wm,Wn	Dividend, Divisor working register pair (direct addressing)
Wn	One of 16 working registers ∈ {W0..W15}
Wnd	One of 16 destination working registers ∈ {W0..W15}
Wns	One of 16 source working registers ∈ {W0..W15}
WREG	W0 (working register used in file register instructions)
Ws	Source W register ∈ { Ws, [Ws], [Ws++], [Ws--], [++Ws], [--Ws] }
Wso	Source W register ∈ { Wns, [Wns], [Wns++], [Wns--], [++Wns], [--Wns], [Wns+Wb] }

TABLE 36-2: INSTRUCTION SET OVERVIEW

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>	1	1	None
	BSW.Z	Ws,Wb	Write Z bit to Ws<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 36-2: INSTRUCTION SET OVERVIEW (CONTINUED)

Assembly Mnemonic	Assembly Syntax		Description	# of Words	# of Cycles	Status Flags Affected
BTSS	BTSS	f,#bit4	Bit Test f, Skip if Set	1	1 (2 or 3)	None
	BTSS	Ws,#bit4	Bit Test Ws, Skip if Set	1	1 (2 or 3)	None
BTST	BTST	f,#bit4	Bit Test f	1	1	Z
	BTST.C	Ws,#bit4	Bit Test Ws to C	1	1	C
	BTST.Z	Ws,#bit4	Bit Test Ws to Z	1	1	Z
	BTST.C	Ws,Wb	Bit Test Ws<Wb> to C	1	1	C
	BTST.Z	Ws,Wb	Bit Test Ws<Wb> to Z	1	1	Z
BTSTS	BTSTS	f,#bit4	Bit Test then Set f	1	1	Z
	BTSTS.C	Ws,#bit4	Bit Test Ws to C, then Set	1	1	C
	BTSTS.Z	Ws,#bit4	Bit Test Ws to Z, then Set	1	1	Z
CALL	CALL	lit23	Call Subroutine	2	2	None
	CALL	Wn	Call Indirect Subroutine	1	2	None
CLR	CLR	f	f = 0x0000	1	1	None
	CLR	WREG	WREG = 0x0000	1	1	None
	CLR	Ws	Ws = 0x0000	1	1	None
CLRWDT	CLRWDT		Clear Watchdog Timer	1	1	WDTO, Sleep
COM	COM	f	$f = \overline{f}$	1	1	N, Z
	COM	f,WREG	$WREG = \overline{f}$	1	1	N, Z
	COM	Ws,Wd	$Wd = \overline{Ws}$	1	1	N, Z
CP	CP	f	Compare f with WREG	1	1	C, DC, N, OV, Z
	CP	Wb,#lit5	Compare Wb with lit5	1	1	C, DC, N, OV, Z
	CP	Wb,Ws	Compare Wb with Ws (Wb – Ws)	1	1	C, DC, N, OV, Z
CP0	CP0	f	Compare f with 0x0000	1	1	C, DC, N, OV, Z
	CP0	Ws	Compare Ws with 0x0000	1	1	C, DC, N, OV, Z
CPB	CPB	f	Compare f with WREG, with Borrow	1	1	C, DC, N, OV, Z
	CPB	Wb,#lit5	Compare Wb with lit5, with Borrow	1	1	C, DC, N, OV, Z
	CPB	Wb,Ws	Compare Wb with Ws, with Borrow ($Wb – Ws – \overline{C}$)	1	1	C, DC, N, OV, Z
CPSEQ	CPSEQ	Wb,Wn	Compare Wb with Wn, Skip if =	1	1 (2 or 3)	None
CPSGT	CPSGT	Wb,Wn	Compare Wb with Wn, Skip if >	1	1 (2 or 3)	None
CPSLT	CPSLT	Wb,Wn	Compare Wb with Wn, Skip if <	1	1 (2 or 3)	None
CPSNE	CPSNE	Wb,Wn	Compare Wb with Wn, Skip if ≠	1	1 (2 or 3)	None
DAW	DAW.B	Wn	Wn = Decimal Adjust Wn	1	1	C
DEC	DEC	f	f = f –1	1	1	C, DC, N, OV, Z
	DEC	f,WREG	WREG = f –1	1	1	C, DC, N, OV, Z
	DEC	Ws,Wd	Wd = Ws – 1	1	1	C, DC, N, OV, Z
DEC2	DEC2	f	f = f – 2	1	1	C, DC, N, OV, Z
	DEC2	f,WREG	WREG = f – 2	1	1	C, DC, N, OV, Z
	DEC2	Ws,Wd	Wd = Ws – 2	1	1	C, DC, N, OV, Z
DISI	DISI	#lit14	Disable Interrupts for k Instruction Cycles	1	1	None
DIV	DIV.SW	Wm,Wn	Signed 16/16-bit Integer Divide	1	18	N, Z, C, OV
	DIV.SD	Wm,Wn	Signed 32/16-bit Integer Divide	1	18	N, Z, C, OV
	DIV.UW	Wm,Wn	Unsigned 16/16-bit Integer Divide	1	18	N, Z, C, OV
	DIV.UD	Wm,Wn	Unsigned 32/16-bit Integer Divide	1	18	N, Z, C, OV
EXCH	EXCH	Wns,Wnd	Swap Wns with Wnd	1	1	None
FF1L	FF1L	Ws,Wnd	Find First One from Left (MSb) Side	1	1	C
FF1R	FF1R	Ws,Wnd	Find First One from Right (LSb) Side	1	1	C

TABLE 36-2: INSTRUCTION SET OVERVIEW (CONTINUED)

Assembly Mnemonic	Assembly Syntax		Description	# of Words	# of Cycles	Status Flags Affected
GOTO	GOTO	Expr	Go to Address	2	2	None
	GOTO	Wn	Go to Indirect	1	2	None
INC	INC	f	f = f + 1	1	1	C, DC, N, OV, Z
	INC	f,WREG	WREG = f + 1	1	1	C, DC, N, OV, Z
	INC	Ws,Wd	Wd = Ws + 1	1	1	C, DC, N, OV, Z
INC2	INC2	f	f = f + 2	1	1	C, DC, N, OV, Z
	INC2	f,WREG	WREG = f + 2	1	1	C, DC, N, OV, Z
	INC2	Ws,Wd	Wd = Ws + 2	1	1	C, DC, N, OV, Z
IOR	IOR	f	f = f .IOR. WREG	1	1	N, Z
	IOR	f,WREG	WREG = f .IOR. WREG	1	1	N, Z
	IOR	#lit10,Wn	Wd = lit10 .IOR. Wd	1	1	N, Z
	IOR	Wb,Ws,Wd	Wd = Wb .IOR. Ws	1	1	N, Z
	IOR	Wb,#lit5,Wd	Wd = Wb .IOR. lit5	1	1	N, Z
LNK	LNK	#lit14	Link Frame Pointer	1	1	None
LSR	LSR	f	f = Logical Right Shift f	1	1	C, N, OV, Z
	LSR	f,WREG	WREG = Logical Right Shift f	1	1	C, N, OV, Z
	LSR	Ws,Wd	Wd = Logical Right Shift Ws	1	1	C, N, OV, Z
	LSR	Wb,Wns,Wnd	Wnd = Logical Right Shift Wb by Wns	1	1	N, Z
	LSR	Wb,#lit5,Wnd	Wnd = Logical Right Shift Wb by lit5	1	1	N, Z
MOV	MOV	f,Wn	Move f to Wn	1	1	None
	MOV	[Wns+Slit10],Wnd	Move [Wns+Slit10] to Wnd	1	1	None
	MOV	f	Move f to f	1	1	N, Z
	MOV	f,WREG	Move f to WREG	1	1	N, Z
	MOV	#lit16,Wn	Move 16-bit Literal to Wn	1	1	None
	MOV.b	#lit8,Wn	Move 8-bit Literal to Wn	1	1	None
	MOV	Wn,f	Move Wn to f	1	1	None
	MOV	Wns,[Wns+Slit10]	Move Wns to [Wns+Slit10]	1	1	
	MOV	Wso,Wdo	Move Ws to Wd	1	1	None
	MOV	WREG,f	Move WREG to f	1	1	N, Z
	MOV.D	Wns,Wd	Move Double from W(ns):W(ns+1) to Wd	1	2	None
	MOV.D	Ws,Wnd	Move Double from Ws to W(nd+1):W(nd)	1	2	None
MUL	MUL.SS	Wb,Ws,Wnd	{Wnd+1, Wnd} = Signed(Wb) * Signed(Ws)	1	1	None
	MUL.SU	Wb,Ws,Wnd	{Wnd+1, Wnd} = Signed(Wb) * Unsigned(Ws)	1	1	None
	MUL.US	Wb,Ws,Wnd	{Wnd+1, Wnd} = Unsigned(Wb) * Signed(Ws)	1	1	None
	MUL.UU	Wb,Ws,Wnd	{Wnd+1, Wnd} = Unsigned(Wb) * Unsigned(Ws)	1	1	None
	MUL.SU	Wb,#lit5,Wnd	{Wnd+1, Wnd} = Signed(Wb) * Unsigned(lit5)	1	1	None
	MUL.UU	Wb,#lit5,Wnd	{Wnd+1, Wnd} = Unsigned(Wb) * Unsigned(lit5)	1	1	None
	MUL	f	W3:W2 = f * WREG	1	1	None
NEG	NEG	f	f = $\overline{f}$ + 1	1	1	C, DC, N, OV, Z
	NEG	f,WREG	WREG = $\overline{f}$ + 1	1	1	C, DC, N, OV, Z
	NEG	Ws,Wd	Wd = $\overline{Ws}$ + 1	1	1	C, DC, N, OV, Z
NOP	NOP		No Operation	1	1	None
	NOPR		No Operation	1	1	None
POP	POP	f	Pop f from Top-of-Stack (TOS)	1	1	None
	POP	Wdo	Pop from Top-of-Stack (TOS) to Wdo	1	1	None
	POP.D	Wnd	Pop from Top-of-Stack (TOS) to W(nd):W(nd+1)	1	2	None
	POP.S		Pop Shadow Registers	1	1	All
PUSH	PUSH	f	Push f to Top-of-Stack (TOS)	1	1	None
	PUSH	Wso	Push Wso to Top-of-Stack (TOS)	1	1	None
	PUSH.D	Wns	Push W(ns):W(ns+1) to Top-of-Stack (TOS)	1	2	None
	PUSH.S		Push Shadow Registers	1	1	None

TABLE 36-2: INSTRUCTION SET OVERVIEW (CONTINUED)

Assembly Mnemonic	Assembly Syntax		Description	# of Words	# of Cycles	Status Flags Affected
PWRSAV	PWRSAV	#lit1	Go into Sleep or Idle mode	1	1	WDTO, Sleep
RCALL	RCALL	Expr	Relative Call	1	2	None
	RCALL	Wn	Computed Call	1	2	None
REPEAT	REPEAT	#lit14	Repeat Next Instruction lit14 + 1 times	1	1	None
	REPEAT	Wn	Repeat Next Instruction (Wn) + 1 times	1	1	None
RESET	RESET		Software Device Reset	1	1	None
RETFIE	RETFIE		Return from Interrupt	1	3 (2)	None
RETLW	RETLW	#lit10,Wn	Return with Literal in Wn	1	3 (2)	None
RETURN	RETURN		Return from Subroutine	1	3 (2)	None
RLC	RLC	f	f = Rotate Left through Carry f	1	1	C, N, Z
	RLC	f,WREG	WREG = Rotate Left through Carry f	1	1	C, N, Z
	RLC	Ws,Wd	Wd = Rotate Left through Carry Ws	1	1	C, N, Z
RLNC	RLNC	f	f = Rotate Left (No Carry) f	1	1	N, Z
	RLNC	f,WREG	WREG = Rotate Left (No Carry) f	1	1	N, Z
	RLNC	Ws,Wd	Wd = Rotate Left (No Carry) Ws	1	1	N, Z
RRC	RRC	f	f = Rotate Right through Carry f	1	1	C, N, Z
	RRC	f,WREG	WREG = Rotate Right through Carry f	1	1	C, N, Z
	RRC	Ws,Wd	Wd = Rotate Right through Carry Ws	1	1	C, N, Z
RRNC	RRNC	f	f = Rotate Right (No Carry) f	1	1	N, Z
	RRNC	f,WREG	WREG = Rotate Right (No Carry) f	1	1	N, Z
	RRNC	Ws,Wd	Wd = Rotate Right (No Carry) Ws	1	1	N, Z
SE	SE	Ws,Wnd	Wnd = Sign-Extended Ws	1	1	C, N, Z
SETM	SETM	f	f = FFFFh	1	1	None
	SETM	WREG	WREG = FFFFh	1	1	None
	SETM	Ws	Ws = FFFFh	1	1	None
SL	SL	f	f = Left Shift f	1	1	C, N, OV, Z
	SL	f,WREG	WREG = Left Shift f	1	1	C, N, OV, Z
	SL	Ws,Wd	Wd = Left Shift Ws	1	1	C, N, OV, Z
	SL	Wb,Wns,Wnd	Wnd = Left Shift Wb by Wns	1	1	N, Z
	SL	Wb,#lit5,Wnd	Wnd = Left Shift Wb by lit5	1	1	N, Z
SUB	SUB	f	f = f – WREG	1	1	C, DC, N, OV, Z
	SUB	f,WREG	WREG = f – WREG	1	1	C, DC, N, OV, Z
	SUB	#lit10,Wn	Wn = Wn – lit10	1	1	C, DC, N, OV, Z
	SUB	Wb,Ws,Wd	Wd = Wb – Ws	1	1	C, DC, N, OV, Z
	SUB	Wb,#lit5,Wd	Wd = Wb – lit5	1	1	C, DC, N, OV, Z
SUBB	SUBB	f	f = f – WREG – ($\overline{C}$)	1	1	C, DC, N, OV, Z
	SUBB	f,WREG	WREG = f – WREG – ($\overline{C}$)	1	1	C, DC, N, OV, Z
	SUBB	#lit10,Wn	Wn = Wn – lit10 – ($\overline{C}$)	1	1	C, DC, N, OV, Z
	SUBB	Wb,Ws,Wd	Wd = Wb – Ws – ($\overline{C}$)	1	1	C, DC, N, OV, Z
	SUBB	Wb,#lit5,Wd	Wd = Wb – lit5 – ($\overline{C}$)	1	1	C, DC, N, OV, Z
SUBR	SUBR	f	f = WREG – f	1	1	C, DC, N, OV, Z
	SUBR	f,WREG	WREG = WREG – f	1	1	C, DC, N, OV, Z
	SUBR	Wb,Ws,Wd	Wd = Ws – Wb	1	1	C, DC, N, OV, Z
	SUBR	Wb,#lit5,Wd	Wd = lit5 – Wb	1	1	C, DC, N, OV, Z
SUBBR	SUBBR	f	f = WREG – f – ($\overline{C}$)	1	1	C, DC, N, OV, Z
	SUBBR	f,WREG	WREG = WREG – f – ($\overline{C}$)	1	1	C, DC, N, OV, Z
	SUBBR	Wb,Ws,Wd	Wd = Ws – Wb – ($\overline{C}$)	1	1	C, DC, N, OV, Z
	SUBBR	Wb,#lit5,Wd	Wd = lit5 – Wb – ($\overline{C}$)	1	1	C, DC, N, OV, Z
SWAP	SWAP.b	Wn	Wn = Nibble Swap Wn	1	1	None
	SWAP	Wn	Wn = Byte Swap Wn	1	1	None

TABLE 36-2: INSTRUCTION SET OVERVIEW (CONTINUED)

Assembly Mnemonic	Assembly Syntax		Description	# of Words	# of Cycles	Status Flags Affected
TBLRDH	TBLRDH	Ws,Wd	Read Prog<23:16> to Wd<7:0>	1	2	None
TBLRDL	TBLRDL	Ws,Wd	Read Prog<15:0> to Wd	1	2	None
TBLWTH	TBLWTH	Ws,Wd	Write Ws<7:0> to Prog<23:16>	1	2	None
TBLWTL	TBLWTL	Ws,Wd	Write Ws to Prog<15:0>	1	2	None
ULNK	ULNK		Unlink Frame Pointer	1	1	None
XOR	XOR	f	f = f .XOR. WREG	1	1	N, Z
	XOR	f,WREG	WREG = f .XOR. WREG	1	1	N, Z
	XOR	#lit10,Wn	Wd = lit10 .XOR. Wd	1	1	N, Z
	XOR	Wb,Ws,Wd	Wd = Wb .XOR. Ws	1	1	N, Z
	XOR	Wb,#lit5,Wd	Wd = Wb .XOR. lit5	1	1	N, Z
ZE	ZE	Ws,Wnd	Wnd = Zero-Extend Ws	1	1	C, Z, N

NOTES:

37.0 ELECTRICAL CHARACTERISTICS

This section provides an overview of the PIC24FJ128GC010 family electrical characteristics. Additional information will be provided in future revisions of this document as it becomes available.

Absolute maximum ratings for the PIC24FJ128GC010 family are listed below. Exposure to these maximum rating conditions for extended periods may affect device reliability. Functional operation of the device at these, or any other conditions above the parameters indicated in the operation listings of this specification, is not implied.

Absolute Maximum Ratings(†)

Ambient temperature under bias ..... -40°C to +100°C
Storage temperature ..... -65°C to +150°C
Voltage on V_{DD} with respect to V_{SS} ..... -0.3V to +4.0V
Voltage on any general purpose digital or analog pin (not 5.5V tolerant) with respect to V_{SS} ..... -0.3V to (V_{DD} + 0.3V)
Voltage on any general purpose digital or analog pin (5.5V tolerant, including $\overline{MCLR}$) with respect to V_{SS}:
- When V_{DD} = 0V: ..... -0.3V to + 4.0V
- When $V_{DD} \geq$ 2.0V: ..... -0.3V to +6.0V

Voltage on AV_{DD} and SV_{DD} with respect to V_{SS} ..... (V_{DD} – 0.3V) to (lesser of: 4.0V or (V_{DD} + 0.3V))
Voltage on AV_{SS} and SV_{SS} with respect to V_{SS} ..... -0.3V to +0.3V
Voltage on CH0+, CH0-, CH1+ and CH1- with respect to SV_{SS} ..... -0.3V to (SV_{DD} + 0.3V)
Voltage on V_{BAT} with respect to V_{SS} ..... -0.3V to +4.0V
Voltage on V_{USB3V3} with respect to V_{SS} ..... (V_{CAP} – 0.3V) to +4.0V
Voltage on V_{BUS} with respect to V_{SS} ..... -0.3V to +6.0V
Voltage on D+ or D- with respect to V_{SS}:
- (0Ω source impedance) **(Note 1)** ..... -0.5V to (V_{USB3V3} + 0.5V)
- (source impedance ≥ 28Ω, $V_{USB3V3} \geq$ 3.0V) ..... -1.0V to +4.6V

Maximum current out of V_{SS} pin ..... 300 mA
Maximum current into V_{DD} pin **(Note 2)** ..... 250 mA
Maximum output current sunk by any I/O pin ..... 25 mA
Maximum output current sourced by any I/O pin ..... 25 mA
Maximum current sunk by all ports ..... 200 mA
Maximum current sourced by all ports **(Note 2)** ..... 200 mA

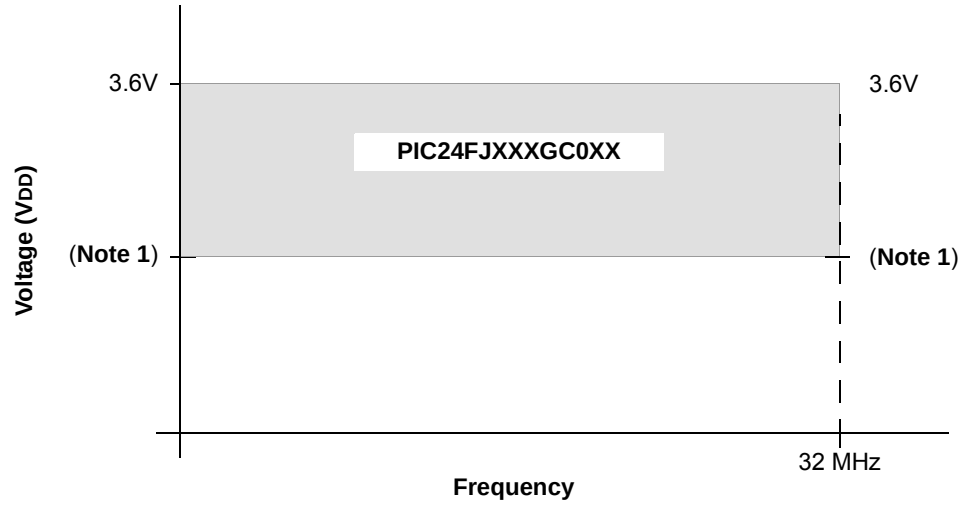
Note 1: The original *"USB 2.0 Specification"* indicated that USB devices should withstand 24-hour short circuits of D+ or D- to V_{BUS} voltages. This requirement was later removed in an Engineering Change Notice (ECN) supplement to the USB specifications, which supersedes the original specifications. PIC24FJ128GC010 family devices will typically be able to survive this short-circuit test, but it is recommended to adhere to the absolute maximum specified here to avoid damaging the device.

2: Maximum allowable current is a function of device maximum power dissipation (see Table 37-1).

† **NOTICE:** Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

37.1 DC Characteristics

FIGURE 37-1: PIC24FJ128GC010 FAMILY VOLTAGE-FREQUENCY GRAPH (INDUSTRIAL)



Note 1: Lower recommended operating boundary is 2.0V or VBOR (when BOR is enabled). For best analog performance, operation above 2.2V is suggested, but not required.

TABLE 37-1: THERMAL OPERATING CONDITIONS

Rating	Symbol	Min	Typ	Max	Unit
PIC24FJ128GC010 Family:					
Operating Junction Temperature Range	TJ	-40	—	+100	°C
Operating Ambient Temperature Range	TA	-40	—	+85	°C
Power Dissipation: Internal Chip Power Dissipation: PINT = VDD x (IDD – Σ IOH) I/O Pin Power Dissipation: PI/O = Σ ({VDD – VOH} x IOH) + Σ (VOL x IOL)	PD	PINT + PI/O			W
Maximum Allowed Power Dissipation	PDMAX	(TJMAX – TA)/θJA			W

TABLE 37-2: THERMAL PACKAGING CHARACTERISTICS

Characteristic	Symbol	Typ	Max	Unit	Note
Package Thermal Resistance, 12x12x1 mm 100-pin TQFP	θJA	45.0	—	°C/W	**(Note 1)**
Package Thermal Resistance, 10x10x1 mm 64-pin TQFP	θJA	48.3	—	°C/W	**(Note 1)**
Package Thermal Resistance, 9x9x0.9 mm 64-pin QFN	θJA	28.0	—	°C/W	**(Note 1)**
Package Thermal Resistance, 10x10x1.1 mm 121-pin BGA	θJA	40.2	—	°C/W	**(Note 1)**

Note 1: Junction to ambient thermal resistance, Theta-JA (θJA) numbers are achieved by package simulations.

TABLE 37-3: DC CHARACTERISTICS: TEMPERATURE AND VOLTAGE SPECIFICATIONS

DC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial				
Param No.	**Symbol**	**Characteristic**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
Operating Voltage							
DC10	VDD	**Supply Voltage**	2.0	—	3.6	V	BOR disabled
			VBOR	—	3.6		BOR enabled
DC12	VDR	**RAM Data Retention Voltage**(1)	Greater of: VPORREL or VBOR	—	—	V	VBOR used only if BOR is enabled (BOREN = 1)
DC16	VPOR	**VDD Start Voltage** to Ensure Internal Power-on Reset Signal	VSS	—	—	V	**(Note 2)**
DC16A	VPORREL	**VDD Power-on Reset Release Voltage**	1.80	1.88	1.95	V	**(Note 3)**
DC17A	SRVDD	**Recommended VDD Rise Skew Rate** to Ensure Internal Power-on Reset Signal	0.05	—	—	V/ms	0-3.3V in 66 ms 0-2.5V in 50 ms **(Note 2)**
DC17B	VBOR	**Brown-out Reset Voltage** on VDD Transition, High-to-Low	2.0	2.1	2.2	V	**(Note 3)**
DC17C	VBATBOR	**VBAT BOR Threshold on VBAT High-to-Low**	—	1.68	—	V	Applies when VBTBOR = 1

Note 1: This is the limit to which VDD may be lowered and the RAM contents will always be retained.

2: If the VPOR or SRVDD parameters are not met, or the application experiences slow power-down VDD ramp rates, it is recommended to enable and use the BOR.

3: On a rising VDD power-up sequence, application firmware execution begins at the higher of the VPORREL or VBOR level (when BOREN = 1).

TABLE 37-4: DC CHARACTERISTICS: OPERATING CURRENT (IDD)

DC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial			
Parameter No.	**Typical(1)**	**Max**	**Units**	**Operating Temperature**	**VDD**	**Conditions**
Operating Current (IDD)(2)						
DC19	0.20	0.28	mA	-40°C to +85°C	2.0V	0.5 MIPS, FOSC = 1 MHz
	0.21	0.28	mA	-40°C to +85°C	3.3V	
DC20	0.38	0.52	mA	-40°C to +85°C	2.0V	1 MIPS, FOSC = 2 MHz
	0.39	0.52	mA	-40°C to +85°C	3.3V	
DC23	1.5	2.0	mA	-40°C to +85°C	2.0V	4 MIPS, FOSC = 8 MHz
	1.5	2.0	mA	-40°C to +85°C	3.3V	
DC24	5.6	7.6	mA	-40°C to +85°C	2.0V	16 MIPS, FOSC = 32 MHz
	5.7	7.6	mA	-40°C to +85°C	3.3V	
DC31	23	78	µA	-40°C to +85°C	2.0V	LPRC (15.5 KIPS), FOSC = 31 kHz
	25	80	µA	-40°C to +85°C	3.3V	

Note 1: Data in the "Typical" column is at 3.3V, +25°C unless otherwise stated. Typical parameters are for design guidance only and are not tested.

2: The supply current is mainly a function of the operating voltage and frequency. Other factors, such as I/O pin loading and switching rate, oscillator type, internal code execution pattern and temperature, also have an impact on the current consumption. No peripheral modules are operating and all of the Peripheral Module Disable (PMD) bits are set.

TABLE 37-5: DC CHARACTERISTICS: IDLE CURRENT (IIDLE)

DC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial			
Parameter No.	**Typical(1)**	**Max**	**Units**	**Operating Temperature**	**VDD**	**Conditions**
Idle Current (IIDLE)						
DC40	116	150	µA	-40°C to +85°C	2.0V	1 MIPS, FOSC = 2 MHz
	123	160	µA	-40°C to +85°C	3.3V	
DC43	0.39	0.50	mA	-40°C to +85°C	2.0V	4 MIPS, FOSC = 8 MHz
	0.41	0.54	mA	-40°C to +85°C	3.3V	
DC47	1.5	1.9	mA	-40°C to +85°C	2.0V	16 MIPS, FOSC = 32 MHz
	1.6	2.0	mA	-40°C to +85°C	3.3V	
DC50	0.54	0.61	mA	-40°C to +85°C	2.0V	4 MIPS (FRC), FOSC = 8 MHz
	0.54	0.64	mA	-40°C to +85°C	3.3V	
DC51	17	78	µA	-40°C to +85°C	2.0V	LPRC (15.5 KIPS), FOSC = 31 kHz
	18	80	µA	-40°C to +85°C	3.3V	

Note 1: Data in the "Typical" column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

TABLE 37-6: DC CHARACTERISTICS: POWER-DOWN CURRENT (IPD)

DC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial			
Parameter No.	**Typical(1)**	**Max**	**Units**	**Operating Temperature**	**VDD**	**Conditions**
Power-Down Current (IPD)						
DC60	2.9	—	µA	-40°C	2.0V	Sleep(2)
	4.3	17	µA	+25°C		
	8.3	—	µA	+60°C		
	20	27.5	µA	+85°C		
	2.9	—	µA	-40°C	3.3V	
	4.3	18	µA	+25°C		
	8.4	—	µA	+60°C		
	20.5	28	µA	+85°C		
DC61	0.07	—	µA	-40°C	2.0V	Low-Voltage Sleep(3)
	0.38	—	µA	+25°C		
	2.6	—	µA	+60°C		
	9.0	—	µA	+85°C		
	0.09	—	µA	-40°C	3.3V	
	0.42	—	µA	+25°C		
	2.75	—	µA	+60°C		
	9.0	—	µA	+85°C		
DC70	0.1	—	nA	-40°C	2.0V	Deep Sleep, capacitor on VCAP is fully discharged
	18	700	nA	+25°C		
	230	—	nA	+60°C		
	1.8	3	µA	+85°C		
	5	—	nA	-40°C	3.3V	
	75	900	nA	+25°C		
	540	—	nA	+60°C		
	1.5	6.0	µA	+85°C		
DC74	0.4	2.0	µA	-40°C to +85°C	0V	RTCC with VBAT mode (LPRC/SOSC)(4)

Note 1: Data in the Typical column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

2: The retention low-voltage regulator is disabled; RETEN (RCON<12>) = 0, $\overline{\text{LPCFG}}$ (CW1<10>) = 1.

3: The retention low-voltage regulator is enabled; RETEN (RCON<12>) = 1, $\overline{\text{LPCFG}}$ (CW1<10>) = 0.

4: The VBAT pin is connected to the battery and RTCC is running with VDD = 0.

TABLE 37-7: DC CHARACTERISTICS: Δ CURRENT (BOR, WDT, DSBOR, DSWDT, LCD)

DC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial			
Parameter No.	**Typical(1)**	**Max**	**Units**	**Operating Temperature**	**VDD**	**Conditions**
Incremental Current Brown-out Reset (ΔBOR)(2)						
DC25	3.1	5.0	μA	-40°C to +85°C	2.0V	ΔBOR(2)
	4.3	6.0	μA	-40°C to +85°C	3.3V	
Incremental Current Watchdog Timer (ΔWDT)(2)						
DC71	0.8	1.5	μA	-40°C to +85°C	2.0V	ΔWDT (with LPRC selected)(2)
	0.8	1.5	μA	-40°C to +85°C	3.3V	
Incremental Current HLVD (ΔHLVD)(2)						
DC75	4.2	15	μA	-40°C to +85°C	2.0V	ΔHLVD(2)
	4.2	15	μA	-40°C to +85°C	3.3V	
Incremental Current Real-Time Clock and Calendar (ΔRTCC)(2)						
DC77	0.30	1.0	μA	-40°C to +85°C	2.0V	ΔRTCC (with SOSC)(2)
	0.35	1.0	μA	-40°C to +85°C	3.3V	
DC77A	0.30	1.0	μA	-40°C to +85°C	2.0V	ΔRTCC (with LPRC)(2)
	0.35	1.0	μA	-40°C to +85°C	3.3V	
Incremental Current Deep Sleep BOR (ΔDSBOR)(2)						
DC81	0.11	0.40	μA	-40°C to +85°C	2.0V	ΔDeep Sleep BOR(2)
	0.12	0.40	μA	-40°C to +85°C	3.3V	
Incremental Current Deep Sleep Watchdog Timer Reset (ΔDSWDT)(2)						
DC80	0.24	0.40	μA	-40°C to +85°C	2.0V	ΔDeep Sleep WDT(2)
	0.24	0.40	μA	-40°C to +85°C	3.3V	
Incremental Current LCD (ΔLCD)(2)						
DC82	0.8	3.0	μA	-40°C to +85°C	3.3V	ΔLCD External/Internal;(2,3) 1/8 MUX, 1/3 Bias
DC90	20	—	μA	-40°C to +85°C	2.0V	ΔLCD Charge Pump;(2,4) 1/8 MUX, 1/3 Bias
	24	—	μA	-40°C to +85°C	3.3V	
VBAT A/D Monitor(5)						
DC91	1.5	—	μA	-40°C to +85°C	3.3V	VBAT = 2V
	4.0	—	μA	-40°C to +85°C	3.3V	VBAT = 3.3V

Note 1: Data in the Typical column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

2: Incremental current while the module is enabled and running.

3: LCD is enabled and running, no glass is connected; the resistor ladder current is not included.

4: LCD is enabled and running, no glass is connected.

5: The A/D channel is connected to the VBAT pin internally; this is the current during A/D VBAT operation.

TABLE 37-8: DC CHARACTERISTICS: I/O PIN INPUT SPECIFICATIONS

DC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial				
Param No.	**Sym**	**Characteristic**	**Min**	**Typ(1)**	**Max**	**Units**	**Conditions**
	VIL	**Input Low Voltage(3)**					
DI10		I/O Pins with ST Buffer	VSS	—	0.2 VDD	V	
DI11		I/O Pins with TTL Buffer	VSS	—	0.15 VDD	V	
DI15		$\overline{MCLR}$	VSS	—	0.2 VDD	V	
DI16		OSCI (XT mode)	VSS	—	0.2 VDD	V	
DI17		OSCI (HS mode)	VSS	—	0.2 VDD	V	
DI18		I/O Pins with I2C™ Buffer	VSS	—	0.3 VDD	V	
DI19		I/O Pins with SMBus Buffer	VSS	—	0.8	V	SMBus enabled
	VIH	**Input High Voltage(3)**					
DI20		I/O Pins with ST Buffer:					
		without 5V Tolerance	0.65 VDD	—	VDD	V	
		with 5V Tolerance	0.65 VDD	—	5.5	V	
DI21		I/O Pins with TTL Buffer:					
		without 5V Tolerance	0.25 VDD + 0.8	—	VDD	V	
		with 5V Tolerance	0.25 VDD + 0.8	—	5.5	V	
DI25		$\overline{MCLR}$	0.8 VDD	—	VDD	V	
DI26		OSCI (XT mode)	0.7 VDD	—	VDD	V	
DI27		OSCI (HS mode)	0.7 VDD	—	VDD	V	
DI28		I/O Pins with I2C™ Buffer	0.7 VDD	—	5.5	V	
DI29		I/O Pins with SMBus Buffer	2.1	—	5.5	V	SMBus enabled
DI30	ICNPU	**CNxx Pull-up Current**	150	350	550	µA	VDD = 3.3V, VPIN = VSS
DI30A	ICNPD	**CNxx Pull-Down Current**	15	70	150	µA	VDD = 3.3V, VPIN = VDD
	IIL	**Input Leakage Current(2)**					
DI50		I/O Ports	—	—	±1	µA	VSS ≤ VPIN ≤ VDD, pin at high-impedance
DI51		Analog Input Pins	—	—	±1	µA	VSS ≤ VPIN ≤ VDD, pin at high-impedance
DI55		$\overline{MCLR}$	—	—	±1	µA	VSS ≤ VPIN ≤ VDD
DI56		OSCI/CLKI	—	—	±1	µA	VSS ≤ VPIN ≤ VDD, EC, XT and HS modes

Note 1: Data in the "Typ" column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

2: Negative current is defined as current sourced by the pin.

3: Refer to Table 1-3 for I/O pin buffer types.

TABLE 37-9: DC CHARACTERISTICS: I/O PIN OUTPUT SPECIFICATIONS

DC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial				
Param No.	Symbol	Characteristic	Min	Typ(1)	Max	Units	Conditions
	VOL	**Output Low Voltage**					
DO10		I/O Ports	—	—	0.4	V	IOL = 6.6 mA, VDD = 3.6V
			—	—	0.4	V	IOL = 5.0 mA, VDD = 2V
DO16		OSCO/CLKO	—	—	0.4	V	IOL = 6.6 mA, VDD = 3.6V
			—	—	0.4	V	IOL = 5.0 mA, VDD = 2V
	VOH	**Output High Voltage**					
DO20		I/O Ports	3.0	—	—	V	IOH = -3.0 mA, VDD = 3.6V
			2.4	—	—	V	IOH = -6.0 mA, VDD = 3.6V
			1.65	—	—	V	IOH = -1.0 mA, VDD = 2V
			1.4	—	—	V	IOH = -3.0 mA, VDD = 2V
DO26		OSCO/CLKO	2.4	—	—	V	IOH = -6.0 mA, VDD = 3.6V
			1.4	—	—	V	IOH = -1.0 mA, VDD = 2V

Note 1: Data in the "Typ" column is at 3.3V, +25°C unless otherwise stated.

TABLE 37-10: DC CHARACTERISTICS: PROGRAM MEMORY

DC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial				
Param No.	Symbol	Characteristic	Min	Typ(1)	Max	Units	Conditions
		Program Flash Memory					
D130	EP	Cell Endurance	20000	—	—	E/W	-40°C to +85°C
D131	VPR	VDD for Read	VMIN	—	3.6	V	VMIN = Minimum operating voltage
D132B		VDD for Self-Timed Write	VMIN	—	3.6	V	VMIN = Minimum operating voltage
D133A	TIW	Self-Timed Word Write Cycle Time	—	20	—	µs	
		Self-Timed Row Write Cycle Time	—	1.5	—	ms	
D133B	TIE	Self-Timed Page Erase Time	20	—	40	ms	
D134	TRETD	Characteristic Retention	20	—	—	Year	If no other specifications are violated
D135	IDDP	Supply Current During Programming	—	5	—	mA	

Note 1: Data in the "Typ" column is at 3.3V, +25°C unless otherwise stated.

TABLE 37-11: INTERNAL VOLTAGE REGULATOR SPECIFICATIONS

Operating Conditions: -40°C < TA < +85°C (unless otherwise stated)							
Param No.	**Symbol**	**Characteristics**	**Min**	**Typ**	**Max**	**Units**	**Comments**
DVR10	VBG	Internal Band Gap Reference	—	1.2	—	V	
DVR11	TBG	Band Gap Reference Start-up Time	—	1	—	ms	
DVR20	VRGOUT	Regulator Output Voltage	—	1.8	—	V	VDD > 2.0V
DVR21	CEFC	External Filter Capacitor Value	4.7	10	—	µF	Series Resistance < 3Ω recommended; < 5Ω required.
DVR	TVREG	Start-up Time	—	10	—	µs	PMSLP = 1 with any POR or BOR
DVR30	VLVR	Low-Voltage Regulator Output Voltage	—	1.2	—	V	RETEN = 1, $\overline{\text{LPCFG}}$ = 0

TABLE 37-12: BAND GAP REFERENCE (BGBUFn) SPECIFICATIONS

Operating Conditions: -40°C < TA < +85°C, 2.0V < (A)VDD < 3.6V(1)							
Param No.	**Sym**	**Characteristics**	**Min**	**Typ**	**Max**	**Units**	**Comments**
DBG01		Recommended Output Capacitance for Optimal Transient Response	—	—	22	µF	BGBUF1 or BGBUF2
DBG02		Output Voltage	1.140	1.200	1.260	V	BUFREF<1:0> = 00, 2.0V < AVDD < 3.6V
DBG03			1.945	2.048	2.151	V	BUFREF<1:0> = 01(2)
DBG04			2.432	2.560	2.688	V	BUFREF<1:0> = 10(2)
DBG05			2.918	3.072	3.226	V	BUFREF<1:0> = 11(2)
DBG07		DC Output Resistance	20	—	—	Ω	BUFREF<1:0> = 00, 2.0V < AVDD ≤ 2.5V
DBG08			20	—	—	Ω	BUFREF<1:0> = 00, 2.5V < AVDD < 3.6V
DBG09			20	—	—	Ω	BUFREF<1:0> = 01, 10 or 11(2)
DBG10		Maximum Continuous DC Output Current Rating	—	—	1	mA	This value is not tested in production **(Note 3)**
DBG11		Module Start-up Time from Disabled State	—	5	—	ms	Time from BUFEN and BUFOE = 1 to output stable, CLOAD = 20 µF
DBG12		Module Start-up Time from Standby Mode	—	100	—	µs	Time from BUFSTBY = 0 to output stable
DBG14		AVDD Active Current	—	100	—	µA	Module enabled, BUFOE = 1

Note 1: No DC loading on module unless otherwise stated.

2: For BUFREF<1:0> ≠ 00, (Reference Output Max + 100 mV) < AVDD < 3.6V.

3: To minimize voltage error, the DC loading on the BGBUF output pins should be <100 µA.

TABLE 37-13: VBAT OPERATING VOLTAGE SPECIFICATIONS

Param No.	Symbol	Characteristic	Min	Typ	Max	Units	Comments
DVB01	VBT	Operating Voltage	1.6	—	3.6	V	Battery connected to the VBAT pin, VBTBOR = 0
DVB02			VBATBOR	—	3.6	V	Battery connected to the VBAT pin, VBTBOR = 1
DVB10	VBTADC	VBAT A/D Monitoring Voltage Specification(1)	1.6	—	3.6	V	A/D monitoring the VBAT pin using the internal A/D channel

Note 1: Measuring the A/D value using the A/D is represented by the equation: Measured Voltage = ((VBAT/2)/VDD) * 4096) for 12-bit A/D.

TABLE 37-14: CTMU CURRENT SOURCE SPECIFICATIONS

DC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial					
Param No.	Sym	Characteristic	Min	Typ(1)	Max	Units	Comments	Conditions
DCT10	IOUT1	CTMU Current Source, Base Range	—	550	—	nA	CTMUICON<1:0> = 00	2.5V < VDD < VDDMAX
DCT11	IOUT2	CTMU Current Source, 10x Range	—	5.5	—	µA	CTMUICON<1:0> = 01	
DCT12	IOUT3	CTMU Current Source, 100x Range	—	55	—	µA	CTMUICON<1:0> = 10	
DCT13	IOUT4	CTMU Current Source, 1000x Range	—	550	—	µA	CTMUICON<1:0> = 11(2)	
DCT21	VΔ	Temperature Diode Voltage Change per Degree Celsius	—	-3	—	mV/°C		

Note 1: Nominal value at center point of current trim range (CTMUICON<7:2> = 000000).

2: Do not use this current range with temperature sensing diode.

TABLE 37-15: USB ON-THE-GO MODULE SPECIFICATIONS

DC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial				
Param No.	Symbol	Characteristic	Min	Typ	Max	Units	Conditions
Operating Voltage							
DUS01	VUSB3V3	USB Supply Voltage	Greater of: 3.0 or (VDD – 0.3V)	3.3	3.6	V	USB module enabled
			(VDD – 0.3V)(1)	—	3.6	V	USB disabled, RG2/RG3 are unused and externally pulled low or left in a high-impedance state
			(VDD – 0.3V)	VDD	3.6	V	USB disabled, RG2/RG3 are used as general purpose I/O

Note 1: The VUSB pin may also be left in a high-impedance state under these conditions. However, if the voltage floats below (VDD – 0.3V), this may result in higher IPD currents than specified. The preferred method is to tie the VUSB pin to VDD, even if the USB module is not used.

TABLE 37-16: HIGH/LOW-VOLTAGE DETECT CHARACTERISTICS

Operating Conditions: -40°C < TA < +85°C (unless otherwise stated)								
Param No.	**Symbol**	**Characteristic**		**Min**	**Typ**	**Max**	**Units**	**Conditions**
DC18	VHLVD	HLVD Voltage on VDD Transition	HLVDL<3:0> = 0100(1)	3.45	—	3.73	V	
			HLVDL<3:0> = 0101	3.30	—	3.57	V	
			HLVDL<3:0> = 0110	3.00	—	3.25	V	
			HLVDL<3:0> = 0111	2.80	—	3.03	V	
			HLVDL<3:0> = 1000	2.67	—	2.92	V	
			HLVDL<3:0> = 1001	2.45	—	2.70	V	
			HLVDL<3:0> = 1010	2.33	—	2.60	V	
			HLVDL<3:0> = 1011	2.21	—	2.49	V	
			HLVDL<3:0> = 1100	2.11	—	2.38	V	
			HLVDL<3:0> = 1101	2.10	—	2.25	V	
			HLVDL<3:0> = 1110	2.00	—	2.15	V	
DC101	VTHL	HLVD Voltage on HLVDIN Pin Transition	HLVDL<3:0> = 1111	—	1.20	—	V	

Note 1: Trip points for values of HLVD<3:0>, from '0000' to '0011', are not implemented.

TABLE 37-17: COMPARATOR DC SPECIFICATIONS

Operating Conditions: 2.0V < VDD < 3.6V, -40°C < TA < +85°C (unless otherwise stated)							
Param No.	**Symbol**	**Characteristic**	**Min**	**Typ**	**Max**	**Units**	**Comments**
D300	VIOFF	Input Offset Voltage	—	12	±30	mV	
D301	VICM	Input Common-Mode Voltage	0	—	VDD	V	
D302	CMRR	Common-Mode Rejection Ratio	55	—	—	dB	
D306	IQCMP	AVDD Quiescent Current per Comparator	—	27	—	µA	Comparator enabled
D307	TRESP	Response Time	—	300	—	ns	**(Note 1)**
D308	TMC2OV	Comparator Mode Change to Valid Output	—	—	10	µs	

Note 1: Measured with one input at VDD/2 and the other transitioning from VSS to VDD, 40 mV step, 15 mV overdrive.

TABLE 37-18: COMPARATOR VOLTAGE REFERENCE DC SPECIFICATIONS

Operating Conditions: 2.0V < VDD < 3.6V, -40°C < TA < +85°C (unless otherwise stated)							
Param No.	**Symbol**	**Characteristic**	**Min**	**Typ**	**Max**	**Units**	**Comments**
VR310	TSET	Settling Time	—	—	10	µs	**(Note 1)**
VRD311	CVRAA	Absolute Accuracy	-100	—	100	mV	
VRD312	CVRUR	Unit Resistor Value (R)	—	4.5	—	kΩ	

Note 1: Measures the interval while CVR<4:0> transitions from '11111' to '00000'.

TABLE 37-19: OPERATIONAL AMPLIFIER SPECIFICATIONS

Operating Conditions: -40°C < TA < +85°C, 2.0V < (A)VDD < 3.6V							
Param No.	**Sym**	**Characteristics**	**Min**	**Typ**	**Max**	**Units**	**Comments**
Op Amp Mode Specifications							
CM20a	SR	Slew Rate	—	1.2	—	V/µs	SPDSEL = 1
CM20B			—	0.4	—	V/µs	SPDSEL = 0
CM23	GBW	Gain Bandwidth Product	—	2.5	—	MHz	SPDSEL = 1
			—	0.5	—	MHz	SPDSEL = 0
CM33	VGAIN	DC Open-Loop Gain	—	80	—	dB	
CM42	VOFFSET	Input Offset Voltage	—	±2	±14	mV	
CM40	VCMR	Common-Mode Input Voltage Range	AVSS	—	AVDD	V	
CM45	IB	Input Bias Current	—	—	—	nA	**(Note 1)**
CM52	VOAMAX	Maximum Output Voltage Swing	AVSS + 50	—	AVDD – 50	mV	0.5V input overdrive, no output loading
CM53	IOA	Maximum Continuous Output Current Rating (DC or RMS AC)	—	—	±6	mA	This value is not tested in production
CM54a	IQOA	AVDD Quiescent Current	—	190	—	µA	Module enabled, SPDSEL = 1, no output load
CM54b			—	40	—	µA	Module enabled, SPDSEL = 0, no output load
Comparator Mode Specifications							
CM10a	TRESPL	Large Signal Comparator Response Time	—	500	—	ns	SPDSEL = 1, 3V step with 1.5V input overdrive
			—	2.6	—	µs	SPDSEL = 0, 3V step with 1.5V input overdrive
CM10B	TRESPS	Small Signal Comparator Response Time	—	1.6	—	µs	SPDSEL = 1, 50 mV step with 15 mV input overdrive
			—	4.6	—	µs	SPDSEL = 0, 50 mV step with 15 mV input overdrive
CM15	VCMCR	Common-Mode Input Voltage Range	AVSS	—	AVDD	V	

Note 1: The op amps use CMOS input circuitry with negligible input bias current. The maximum “effective bias current” is the I/O pin leakage specified by electrical Parameter DI51.

37.2 AC Characteristics and Timing Parameters

The information contained in this section defines the PIC24FJ128GC010 family AC characteristics and timing parameters.

TABLE 37-20: TEMPERATURE AND VOLTAGE SPECIFICATIONS – AC

AC CHARACTERISTICS	**Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +85°C for Industrial Operating voltage VDD range as described in **Section 37.1 "DC Characteristics"**.

FIGURE 37-2: LOAD CONDITIONS FOR DEVICE TIMING SPECIFICATIONS

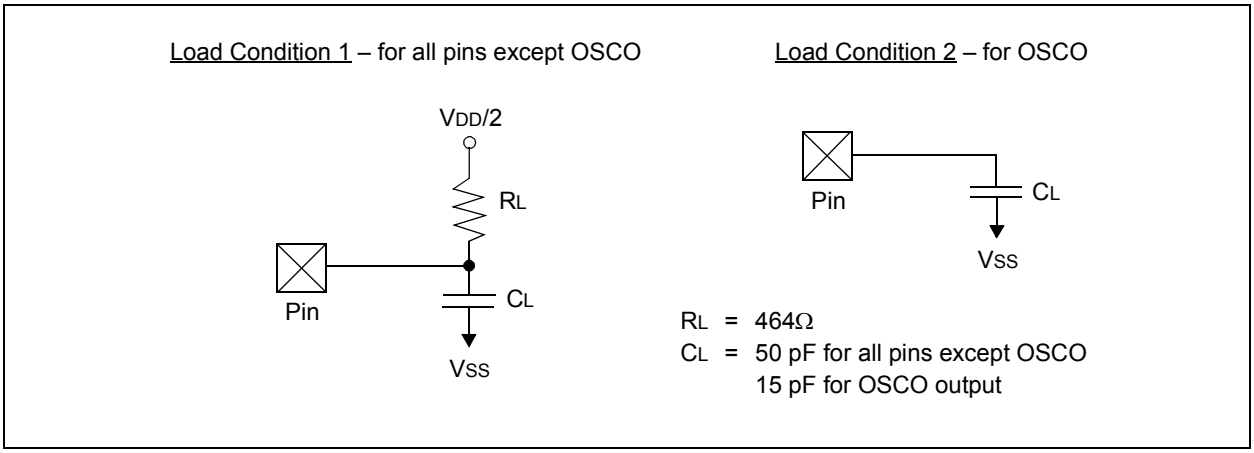


TABLE 37-21: CAPACITIVE LOADING REQUIREMENTS ON OUTPUT PINS

Param No.	Symbol	Characteristic	Min	Typ(1)	Max	Units	Conditions
DO50	COSCO	OSCO/CLKO Pin	—	—	15	pF	In XT and HS modes when external clock is used to drive OSCI
DO56	CIO	All I/O Pins and OSCO	—	—	50	pF	EC mode
DO58	CB	SCLx, SDAx	—	—	400	pF	In I^2C™ mode

Note 1: Data in the "Typ" column is at 3.3V, +25°C unless otherwise stated.

FIGURE 37-3: EXTERNAL CLOCK TIMING

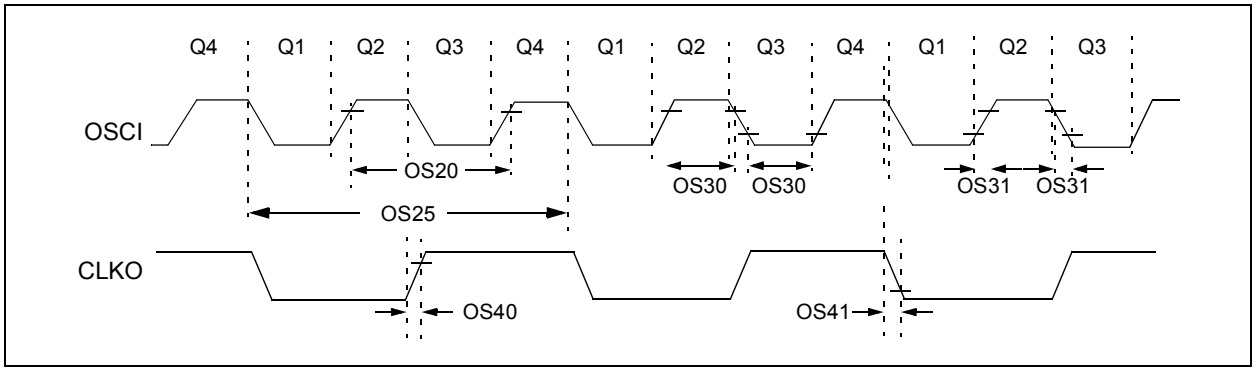


TABLE 37-22: EXTERNAL CLOCK TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial				
Param No.	**Symbol**	**Characteristic**	**Min**	**Typ(1)**	**Max**	**Units**	**Conditions**
OS10	FOSC	External CLKI Frequency (External clocks allowed only in EC mode)	DC 1.97	— —	32 48	MHz MHz	EC ECPLL (**Note 2**)
		Oscillator Frequency	3.5 4 10 12 31	— — — — —	10 8 32 32 33	MHz MHz MHz MHz kHz	XT XTPLL HS HSPLL SOSC
OS20	TOSC	TOSC = 1/FOSC	—	—	—	—	See Parameter OS10 for FOSC value
OS25	TCY	Instruction Cycle Time(3)	62.5	—	DC	ns	
OS30	TosL, TosH	External Clock in (OSCI) High or Low Time	0.45 x TOSC	—	—	ns	EC
OS31	TosR, TosF	External Clock in (OSCI) Rise or Fall Time	—	—	20	ns	EC
OS40	TckR	CLKO Rise Time(4)	—	6	10	ns	
OS41	TckF	CLKO Fall Time(4)	—	6	10	ns	

Note 1: Data in the "Typ" column is at 3.3V, +25°C unless otherwise stated.

2: Represents input to the system clock prescaler. PLL dividers and postscalers must still be configured so that the system clock frequency does not exceed the maximum frequency shown in Figure 37-1.

3: Instruction cycle period (TCY) equals two times the input oscillator time base period. All specified values are based on characterization data for that particular oscillator type, under standard operating conditions, with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at "Min." values with an external clock applied to the OSCI/CLKI pin. When an external clock input is used, the "Max." cycle time limit is "DC" (no clock) for all devices.

4: Measurements are taken in EC mode. The CLKO signal is measured on the OSCO pin. CLKO is low for the Q1-Q2 period (1/2 TCY) and high for the Q3-Q4 period (1/2 TCY).

TABLE 37-23: PLL CLOCK TIMING SPECIFICATIONS

AC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial				
Param No.	Symbol	Characteristic	Min	Typ	Max	Units	Conditions
OS50	FPLLI	PLL Input Frequency Range(1)	1.97	4	4.04	MHz	ECPLL, XTPLL, HSPLL or FRCPLL modes
OS52	TLOCK	PLL Start-up Time (Lock Time)	—	—	128	µs	
OS53	DCLK	CLKO Stability (Jitter)	-0.25	—	0.25	%	

Note 1: The PLL accepts a 1.97 MHz to 4.04 MHz input frequency. Higher input frequencies, up to 48 MHz, may be supplied to the PLL if they are prescaled down by the PLLDIVx Configuration bits into the 1.97 MHz to 4.04 MHz range.

TABLE 37-24: INTERNAL RC ACCURACY

AC CHARACTERISTICS		Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial				
Param No.	Characteristic	Min	Typ	Max	Units	Conditions
F20	FRC Accuracy @ 8 MHz(4)	-1	±0.15	1	%	2.0V ≤ VDD ≤ 3.6V, 0°C ≤ TA ≤ +85°C (**Note 1**)
		-1.5	—	1.5	%	2.0V ≤ VDD ≤ 3.6V, -40°C ≤ TA < 0°C
		-0.20	±0.05	0.20	%	2.0V ≤ VDD ≤ 3.6V, -40°C ≤ TA ≤ +85°C, self-tune is enabled and locked (**Note 2**)
F21	LPRC @ 31 kHz	-20	—	20	%	
F22	OSCTUN Step-Size	—	0.05	—	%/bit	
F23	FRC Self-Tune Lock Time	—	<5	8	ms	(**Note 3**)

Note 1: To achieve this accuracy, physical stress applied to the microcontroller package (ex., by flexing the PCB) must be kept to a minimum.

2: Accuracy measured with respect to reference source accuracy.

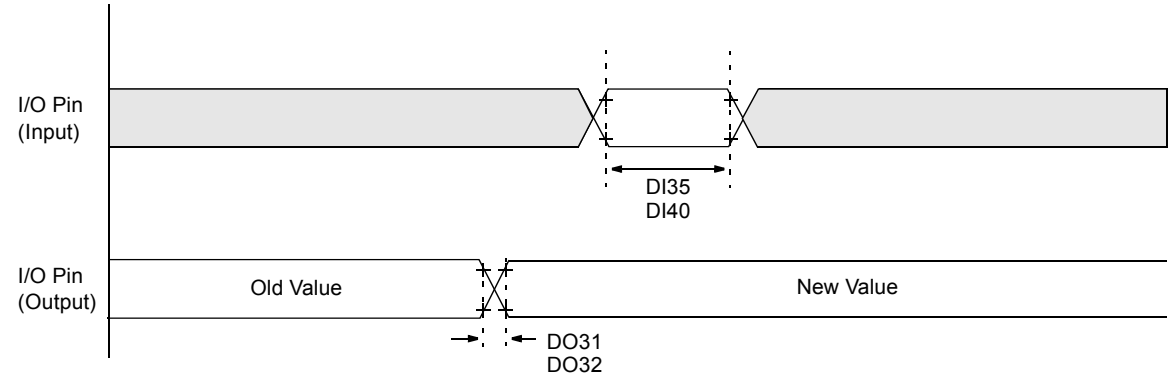
3: Time from reference clock is stable and in range until the FRC is tuned within the range specified by F20 (with self-tune).

4: Other frequencies that are derived from the FRC (either through digital division by prescalers or multiplication through a PLL) will also have the same accuracy tolerance specifications as provided here.

TABLE 37-25: RC OSCILLATOR START-UP TIME

AC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial				
Param No.	Symbol	Characteristic	Min	Typ	Max	Units	Conditions
FR0	TFRC	FRC Oscillator Start-up Time	—	15	—	µs	
FR1	TLPRC	Low-Power RC Oscillator Start-up Time	—	50	—	µs	

FIGURE 37-4: CLKO AND I/O TIMING CHARACTERISTICS



Note: Refer to Figure 37-2 for load conditions.

TABLE 37-26: CLKO AND I/O TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial				
Param No.	**Symbol**	**Characteristic**	**Min**	**Typ(1)**	**Max**	**Units**	**Conditions**
DO31	TIOR	Port Output Rise Time	—	5	25	ns	
DO32	TIOF	Port Output Fall Time	—	5	25	ns	
DI35	TINP	INTx Pin High or Low Time (input)	20	—	—	ns	
DI40	TRBP	CNx High or Low Time (input)	2	—	—	TCY	

Note 1: Data in the "Typ" column is at 3.3V, +25°C unless otherwise stated.

TABLE 37-27: RESET AND BROWN-OUT RESET REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial				
Param No.	**Symbol**	**Characteristic**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
SY10	TMCL	$\overline{\text{MCLR}}$ Pulse Width (Low)	2	—	—	μs	
SY12	TPOR	Power-on Reset Delay	—	2	—	μs	
SY13	TIOZ	I/O High-Impedance from $\overline{\text{MCLR}}$ Low or Watchdog Timer Reset	Lesser of: (3 TCY + 2) or 700	—	(3 TCY + 2)	μs	
SY25	TBOR	Brown-out Reset Pulse Width	1	—	—	μs	VDD ≤ VBOR
SY45	TRST	Internal State Reset Time	—	50	—	μs	
SY70	TDSWU	Deep Sleep Wake-up Time	—	200	—	μs	VCAP fully discharged before wake-up
SY71	TPM	Program Memory Wake-up Time	—	20	—	μs	Sleep wake-up with PMSLP = 0
			—	1	—	μs	Sleep wake-up with PMSLP = 1
SY72	TLVR	Low-Voltage Regulator Wake-up Time	—	90	—	μs	Sleep wake-up with PMSLP = 0
			—	70	—	μs	Sleep wake-up with PMSLP = 1

TABLE 37-28: 12-BIT PIPELINE A/D MODULE SPECIFICATIONS

AC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial				
Param No.	**Symbol**	**Characteristic**	**Min.**	**Typ**	**Max.**	**Units**	**Conditions**
Device Supply							
AD01	AVDD	Module VDD Supply	Greater of: (VDD – 0.3) or 2.0	—	Lesser of: (VDD + 0.3) or 3.6	V	
AD02	AVSS	Module VSS Supply	VSS – 0.3	—	VSS + 0.3	V	
AD03	IQADC1	AVDD Current	—	0.4	—	mA	VDD = AVDD = 3.3V, 1 MHz A/D clock, PWRLVL = 0, A/D active and not halted
	IQADC8		—	2	—	mA	VDD = AVDD = 3.3V, 8 MHz A/D clock, PWRLVL = 1, A/D active and not halted
Reference Inputs							
AD05	VREFH	Reference Voltage High	AVSS + 1.8	—	AVDD	V	
AD06	VREFL	Reference Voltage Low	AVSS	—	VREFH – 1.2	V	
AD07	VREF	Absolute Reference Voltage	AVSS – 0.3	—	AVDD + 0.3	V	
			AVSS + 1.8	—	AVDD + 0.3	V	REFPUMP = 0
			AVSS + 1.2	—	2.0	V	REFPUMP = 1
Analog Input							
AD10	VINH-VINL	Full-Scale Input Span	VREFL	—	VREFH	V	**(Note 2)**
AD11	VIN	Absolute Input Voltage	AVSS – 0.3	—	AVDD + 0.3	V	
AD12	VINL	Absolute VINL Input Voltage	AVSS – 0.3	—	AVDD/3	V	
AD13	ILKG	Leakage Current	—	±1.0	610	nA	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V, Source Impedance = 2.5 kΩ
A/D Accuracy							
AD20B	Nr	Resolution	—	12	—	bits	
AD21B	INL	Integral Nonlinearity	-6	—	+6	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V
AD22B	DNL	Differential Nonlinearity	-3	—	+3	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V **(Note 3)**
AD23B	GERR	Gain Error	-8	—	+8	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V
AD24B	EOFF	Offset Error	-12	—	+12	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V
AD25B		Monotonicity	—	—	—	—	(**Note 1**)

Note 1: The A/D conversion result never decreases with an increase in the input voltage.

2: Measurements are taken with the external AVREF+ and AVREF- used as the A/D voltage reference.

3: Overall accuracy can be improved if 4 or more consecutive measurements are averaged. For details, see the *"PIC24F Family Reference Manual"*, **"12-Bit, High-Speed Pipeline A/D Converter"** (DS30686).

FIGURE 37-5: 12-Bit A/D DNL, 10 MS/S, AVDD = 3.0V, TYPICAL

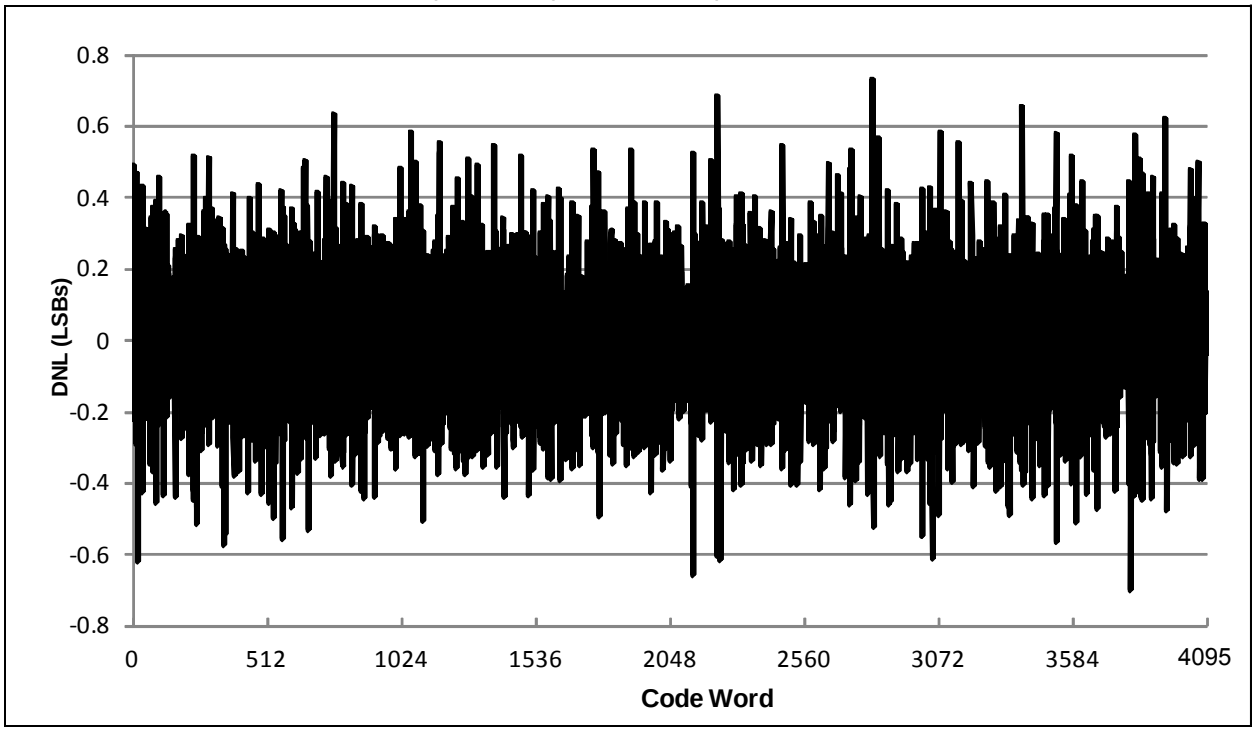


FIGURE 37-6: 12-Bit A/D INL, 10 MS/S, AVDD = 3.0V, TYPICAL

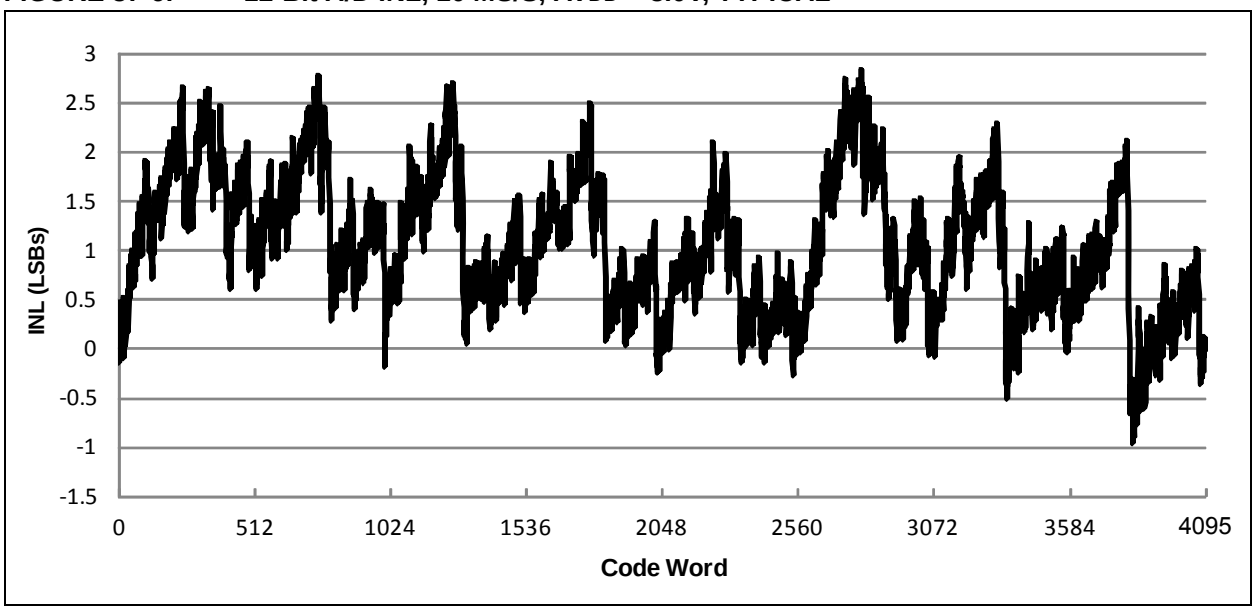


TABLE 37-29: 12-BIT PIPELINE A/D CONVERSION TIMING REQUIREMENTS(1)

AC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial				
Param No.	**Sym**	**Characteristic**	**Min.**	**Typ**	**Max.**	**Units**	**Conditions**
Clock Parameters							
AD50	TAD	A/D Clock Period	100	—	1000	ns	
Conversion Rate							
AD55	tCONV	Single Conversion Latency	—	9	—	TAD	
AD56	FCNV	Throughput Rate	—	—	10	Msps	AVDD > 2.7V
AD57	tSAMP	Sample Time	0.5	—	31	TAD	
Clock Parameters							
AD61	tPSS	Sample Start Delay from Setting Sample bit (SAMP)	—	—	1	TAD	

Note 1: The 31 TAD value is the maximum set by the SAMCx bits. Long (up to indefinite) sampling times are allowed on the channel selected by ADTBL0 when the A/D is Idle.

TABLE 37-30: 10-BIT DAC SPECIFICATIONS

AC CHARACTERISTICS			Operating Conditions: -40°C < TA < +85°C, 2.0V < (A)VDD < 3.6V(1)				
Param No.	**Sym**	**Characteristic**	**Min**	**Typ**	**Max**	**Units**	**Comments**
DAC01		Resolution	10	—	—	bits	
DAC02		DVREF+ Input Voltage Range	—	—	AVDD	V	
DAC03	DNL	Differential Linearity Error	-1	—	+1	LSb	
DAC04	INL	Integral Linearity Error	-2.5	—	+2.5	LSb	
DAC05		Offset Error	-20	—	+20	mV	Code 000h
DAC06		Gain Error	-2.5	—	2.5	LSb	Code 3FFh, not including offset error
DAC07		Monotonicity	**(Note 2)**			—	
DAC08		Maximum Output Voltage Swing	AVSS + 20	—	AVDD – 20	mV	No output load
DAC09		Slew Rate	—	3.8	—	V/µs	
DAC10		Settling Time	—	0.9	—	µs	Within ½ LSb of final value, transition from ¼ to ¾ full-scale range
DAC11		Maximum Continuous Output Current Rating (DC or AC RMS)	—	—	6	mA	This value is not tested in production
DAC12		AVDD Quiescent Current	—	700	—	µA	Module enabled, DAC Reference = AVDD, no output load
DAC13		DVREF+ Quiescent Current	—	330	—	µA	Module enabled, DAC Reference = DVREF+

Note 1: Unless otherwise stated, test conditions are with VDD = AVDD = DVREF+ = 3.3V, 3 kΩ load to VSS.

2: DAC output voltage never decreases with an increase in the data code.

TABLE 37-31: 16-BIT SIGMA-DELTA A/D CONVERTER SPECIFICATIONS

AC Characteristics			Operating Conditions: -40°C < TA < +85°C, 2.0V < SVDD < 3.6V				
Param No.	Sym	Characteristics	Min	Typ(1)	Max	Units	Comments
SDC01		SVREF+ Input Voltage Range	SVREF-	—	SVDD	V	
SDC02		SVREF- Input Voltage Range	SVSS	—	SVREF+	V	
SDC03		Analog Channel Absolute Input Voltage Range	SVSS	—	SVDD	V	Full range accepted, independent of SVREF+/SVREF-
SDC04		Analog Channel Differential Input Voltage Range	—	—	±SVDD	V	Limit differential input to ±[(SVREF+ – SVREF-)/GAIN] for non-saturated result
SDC05	INL	Integral Linearity Error	-20	6	+20	LSb	See Figure 37-8
SDC06A		Offset Error	-12	10	+12	mV	1x Gain mode without using VOSCAL, offset error is mostly independent of gain setting
SDC06B			—	0.0	—	mV	1x Gain mode, after VOSCAL based firmware correction
SDC08		Unadjusted Gain Error	-10	-6	-0.001	%	1x Gain mode uncorrected
SDC09B		SINAD	—	75	—	dB	OSR 1024, high dithering enabled, 25 Hz input
SDC10		Differential Input Impedance	—	684	—	kΩ	4 MHz A/D clock, input impedance is proportional to 1/(A/D Clock Frequency)
SDC13		A/D Clock Input Frequency	1	—	4	MHz	
SDC14		SVDD Operating Current	—	3.6	—	mA	Module enabled, 4 MHz A/D clock, SVDD/SVSS as A/D reference, PWRLVL = 1

Note 1: Unless otherwise stated, typical column test conditions are with VDD = AVDD = SVDD = 3.3V, +25°C, 1x Gain mode, OSR 1024, chopping enabled, SVDD/SVSS are used as A/D references.

FIGURE 37-7: 16-Bit S/D A/D DNL, SVDD = 3.0V, TYPICAL

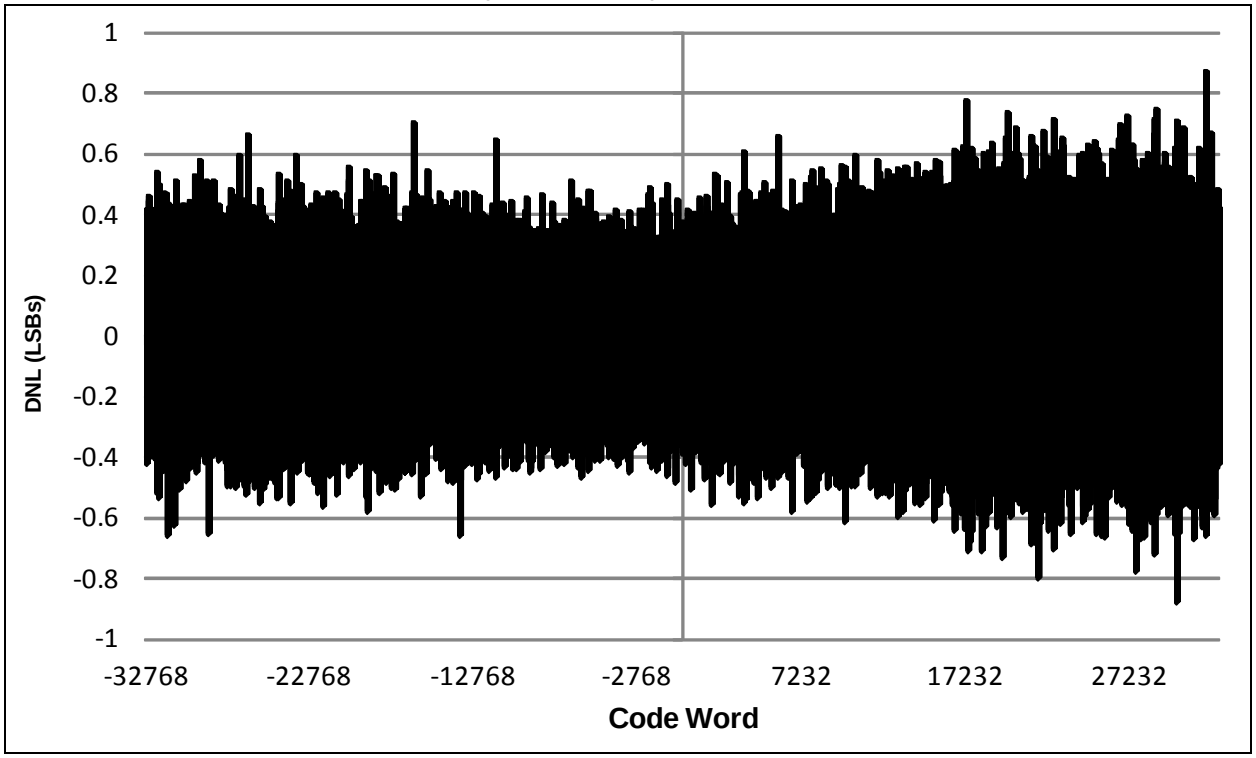
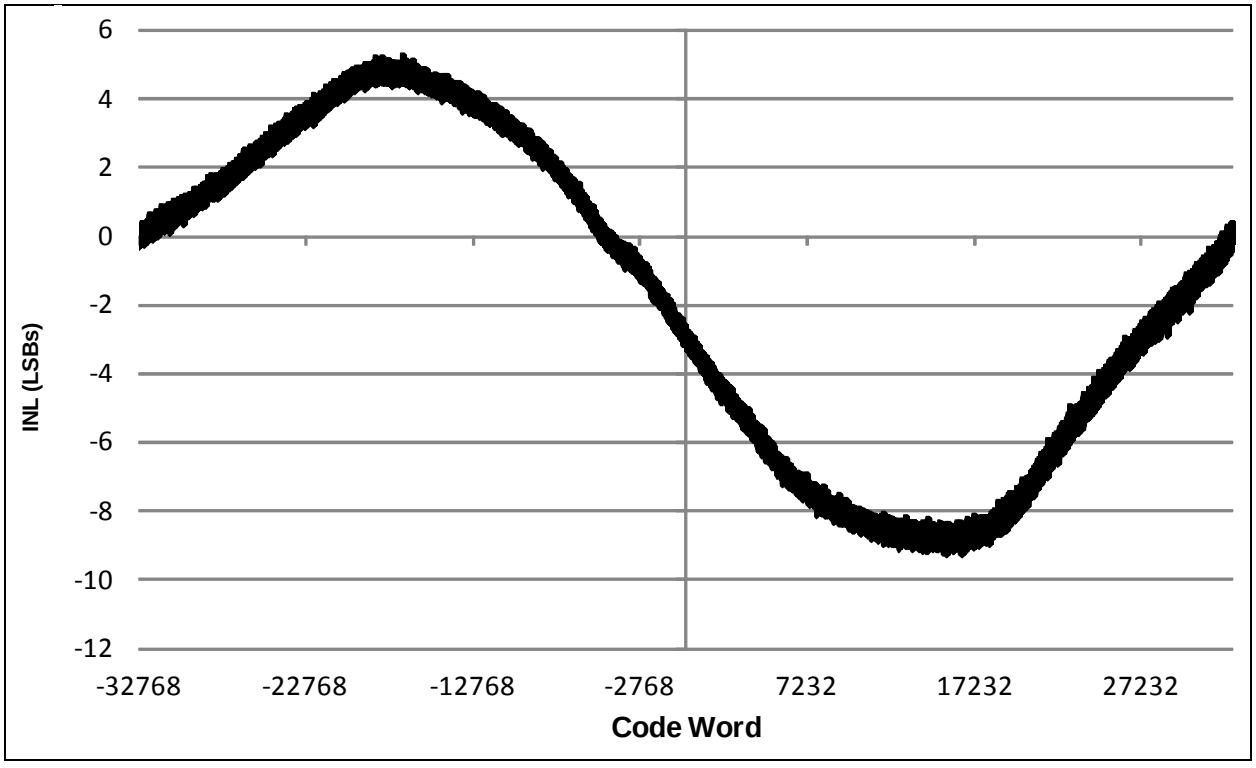


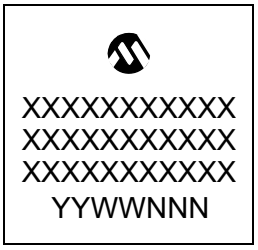
FIGURE 37-8: 16-Bit S/D A/D INL, SVDD = 3.0V, TYPICAL



38.0 PACKAGING INFORMATION

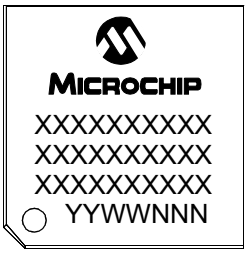
38.1 Package Marking Information

64-Lead QFN (9x9x0.9 mm)

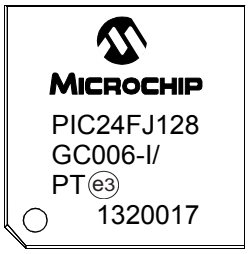


Example

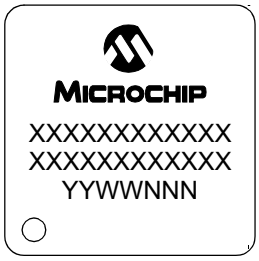
64-Lead TQFP (10x10x1 mm)



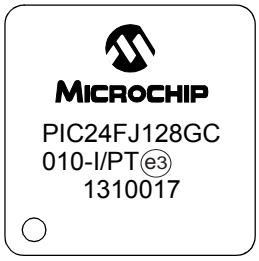
Example



100-Lead TQFP (12x12x1 mm)



Example

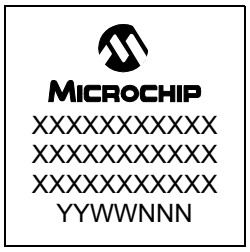


Legend:	XX...X	Customer-specific information
	Y	Year code (last digit of calendar year)
	YY	Year code (last 2 digits of calendar year)
	WW	Week code (week of January 1 is week '01')
	NNN	Alphanumeric traceability code
	(e3)	Pb-free JEDEC designator for Matte Tin (Sn)
	*	This package is Pb-free. The Pb-free JEDEC designator ((e3)) can be found on the outer packaging for this package.

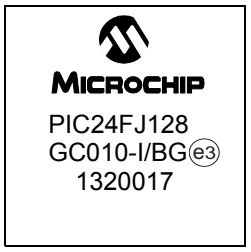
Note: In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line, thus limiting the number of available characters for customer-specific information.

38.2 Package Marking Information (Continued)

121-BGA (10x10x1.1 mm)



Example

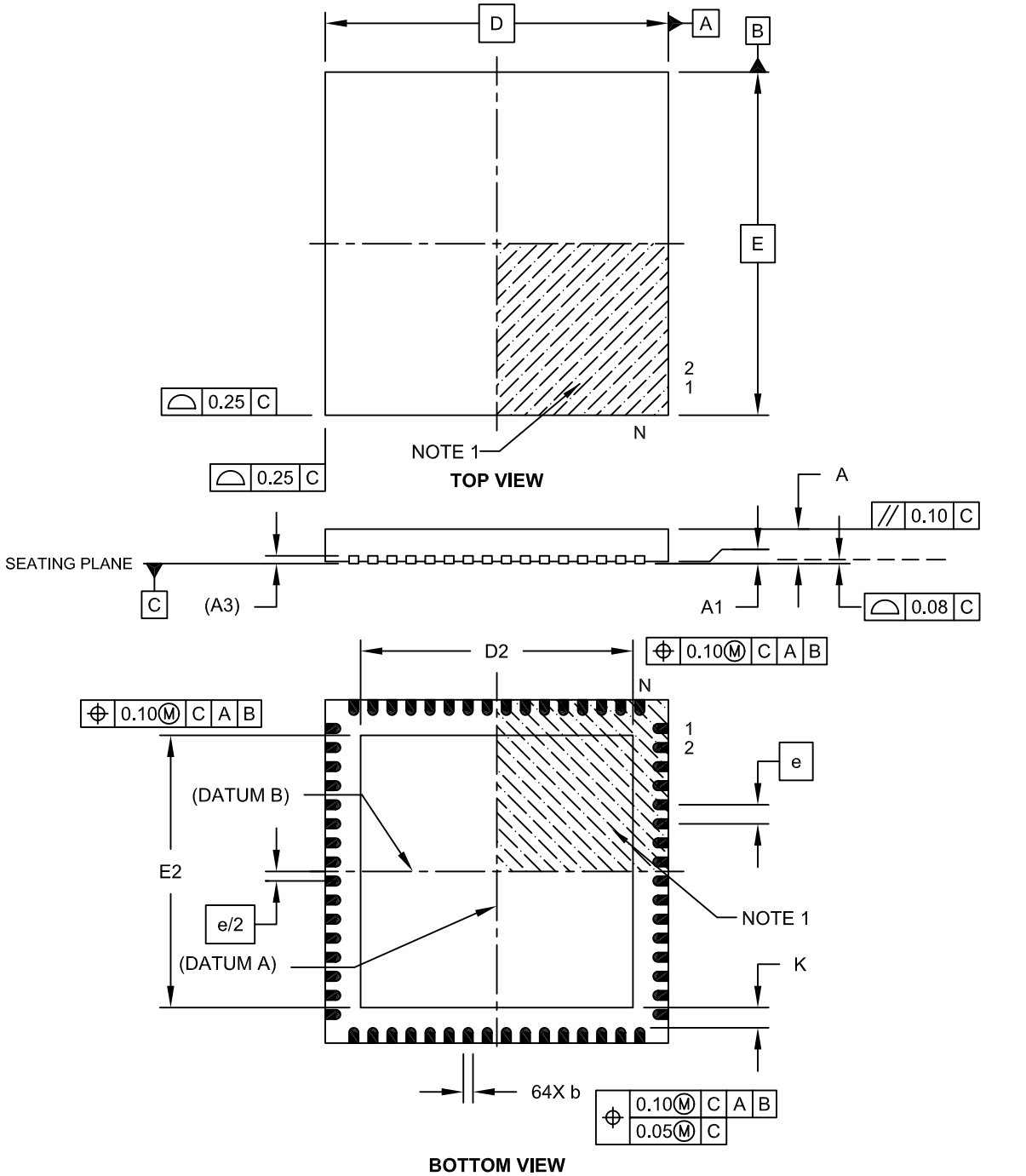


38.3 Package Details

The following sections give the technical details of the packages.

64-Lead Plastic Quad Flat, No Lead Package (MR) – 9x9x0.9 mm Body [QFN] With 7.15 x 7.15 Exposed Pad [QFN]

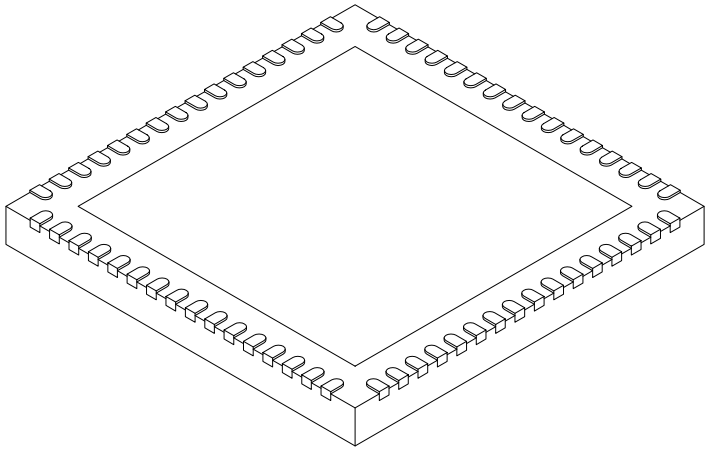
Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Microchip Technology Drawing C04-149C Sheet 1 of 2

64-Lead Plastic Quad Flat, No Lead Package (MR) – 9x9x0.9 mm Body [QFN]
With 7.15 x 7.15 Exposed Pad [QFN]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		64	
Pitch	e		0.50 BSC	
Overall Height	A	0.80	0.90	1.00
Standoff	A1	0.00	0.02	0.05
Contact Thickness	A3		0.20 REF	
Overall Width	E		9.00 BSC	
Exposed Pad Width	E2	7.05	7.15	7.50
Overall Length	D		9.00 BSC	
Exposed Pad Length	D2	7.05	7.15	7.50
Contact Width	b	0.18	0.25	0.30
Contact Length	L	0.30	0.40	0.50
Contact-to-Exposed Pad	K	0.20	-	-

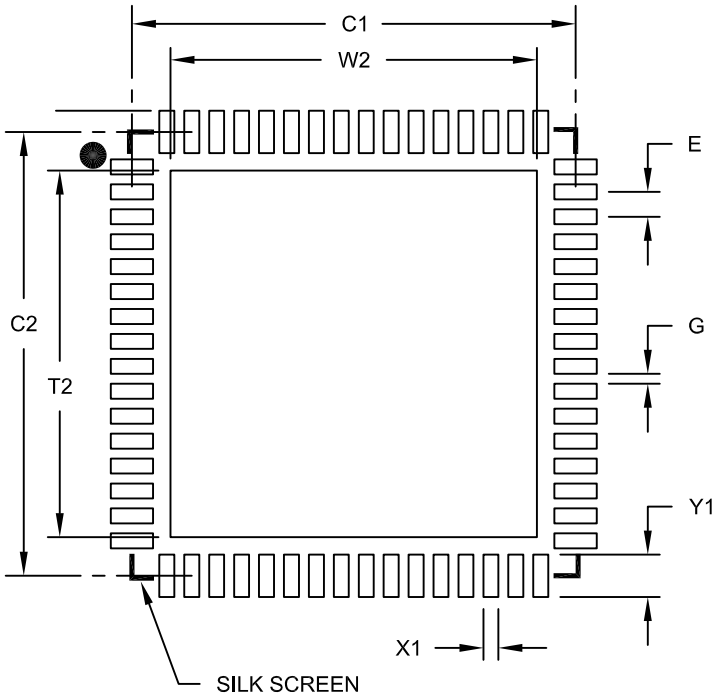
Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated.
3. Dimensioning and tolerancing per ASME Y14.5M.
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-149C Sheet 2 of 2

64-Lead Plastic Quad Flat, No Lead Package (MR) – 9x9x0.9 mm Body [QFN]
With 0.40 mm Contact Length

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E		0.50 BSC	
Optional Center Pad Width	W2			7.35
Optional Center Pad Length	T2			7.35
Contact Pad Spacing	C1		8.90	
Contact Pad Spacing	C2		8.90	
Contact Pad Width (X64)	X1			0.30
Contact Pad Length (X64)	Y1			0.85
Distance Between Pads	G	0.20		

Notes:

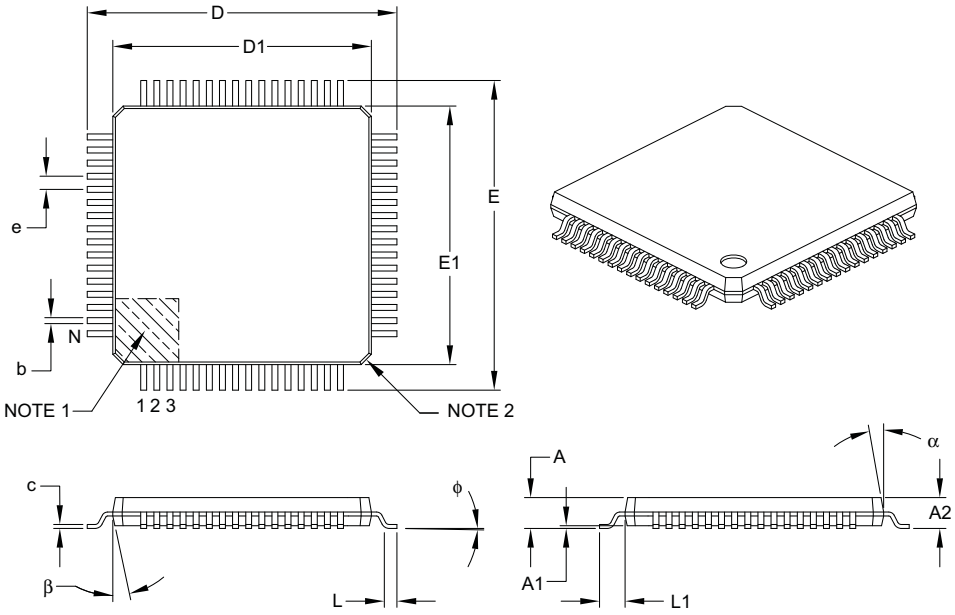
1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2149A

64-Lead Plastic Thin Quad Flatpack (PT) – 10x10x1 mm Body, 2.00 mm [TQFP]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Leads	N		64	
Lead Pitch	e		0.50 BSC	
Overall Height	A	–	–	1.20
Molded Package Thickness	A2	0.95	1.00	1.05
Standoff	A1	0.05	–	0.15
Foot Length	L	0.45	0.60	0.75
Footprint	L1		1.00 REF	
Foot Angle	ϕ	0°	3.5°	7°
Overall Width	E		12.00 BSC	
Overall Length	D		12.00 BSC	
Molded Package Width	E1		10.00 BSC	
Molded Package Length	D1		10.00 BSC	
Lead Thickness	c	0.09	–	0.20
Lead Width	b	0.17	0.22	0.27
Mold Draft Angle Top	α	11°	12°	13°
Mold Draft Angle Bottom	β	11°	12°	13°

Notes:

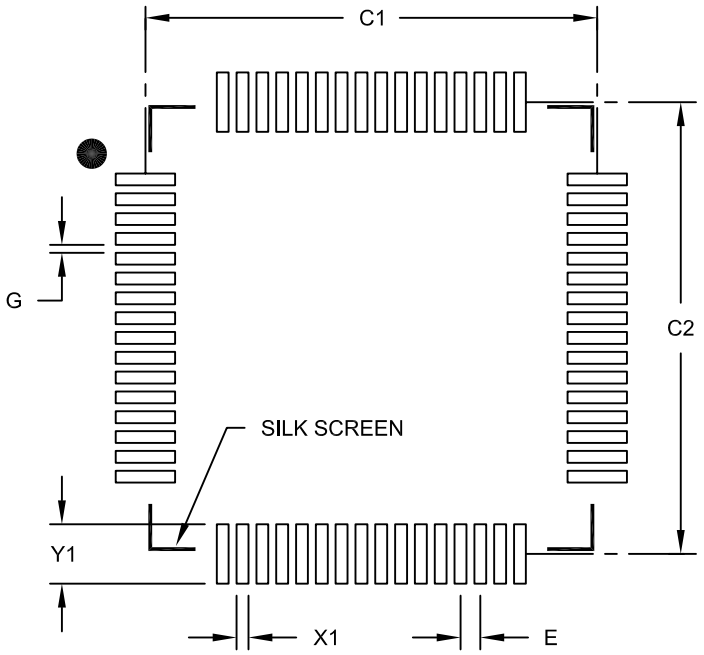
1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Chamfers at corners are optional; size may vary.
3. Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.25 mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-085B

64-Lead Plastic Thin Quad Flatpack (PT) 10x10x1 mm Body, 2.00 mm Footprint [TQFP]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E		0.50 BSC	
Contact Pad Spacing	C1		11.40	
Contact Pad Spacing	C2		11.40	
Contact Pad Width (X64)	X1			0.30
Contact Pad Length (X64)	Y1			1.50
Distance Between Pads	G	0.20		

Notes:

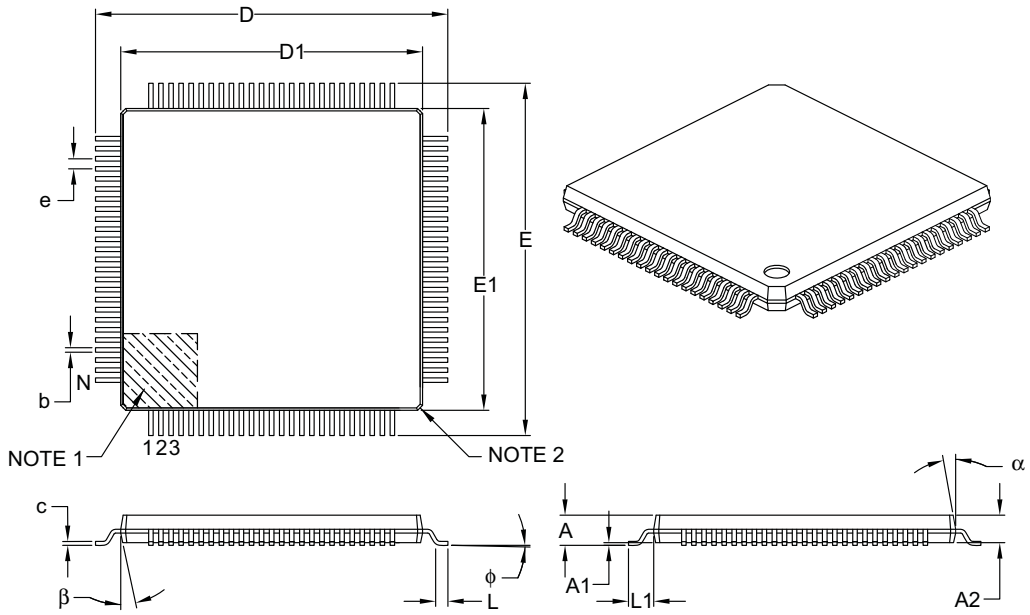
1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2085B

100-Lead Plastic Thin Quad Flatpack (PT) – 12x12x1 mm Body, 2.00 mm [TQFP]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Leads	N		100	
Lead Pitch	e		0.40 BSC	
Overall Height	A	–	–	1.20
Molded Package Thickness	A2	0.95	1.00	1.05
Standoff	A1	0.05	–	0.15
Foot Length	L	0.45	0.60	0.75
Footprint	L1		1.00 REF	
Foot Angle	ϕ	0°	3.5°	7°
Overall Width	E		14.00 BSC	
Overall Length	D		14.00 BSC	
Molded Package Width	E1		12.00 BSC	
Molded Package Length	D1		12.00 BSC	
Lead Thickness	c	0.09	–	0.20
Lead Width	b	0.13	0.18	0.23
Mold Draft Angle Top	α	11°	12°	13°
Mold Draft Angle Bottom	β	11°	12°	13°

Notes:

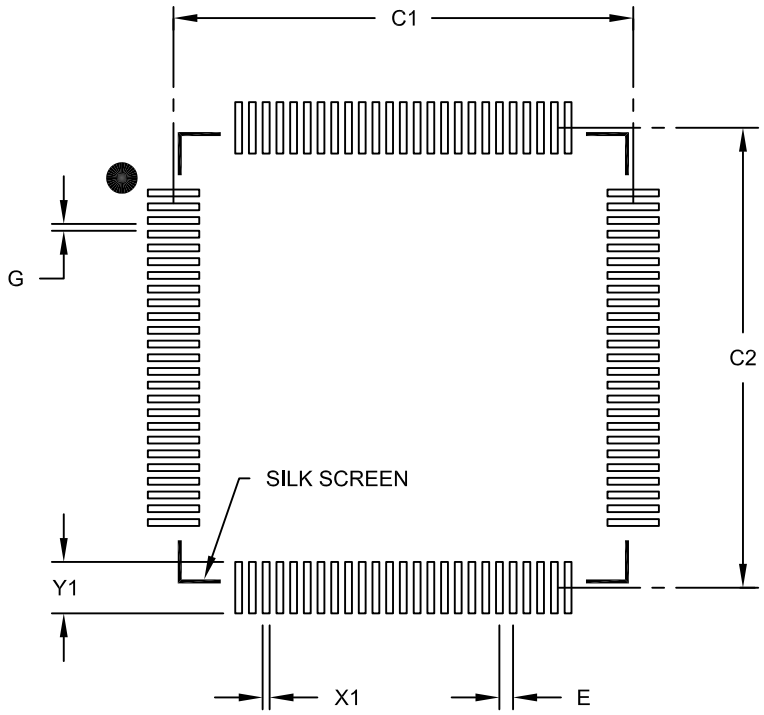
1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Chamfers at corners are optional; size may vary.
3. Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.25 mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-100B

100-Lead Plastic Thin Quad Flatpack (PT)-12x12x1mm Body, 2.00 mm Footprint [TQFP]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E		0.40 BSC	
Contact Pad Spacing	C1		13.40	
Contact Pad Spacing	C2		13.40	
Contact Pad Width (X100)	X1			0.20
Contact Pad Length (X100)	Y1			1.50
Distance Between Pads	G	0.20		

Notes:

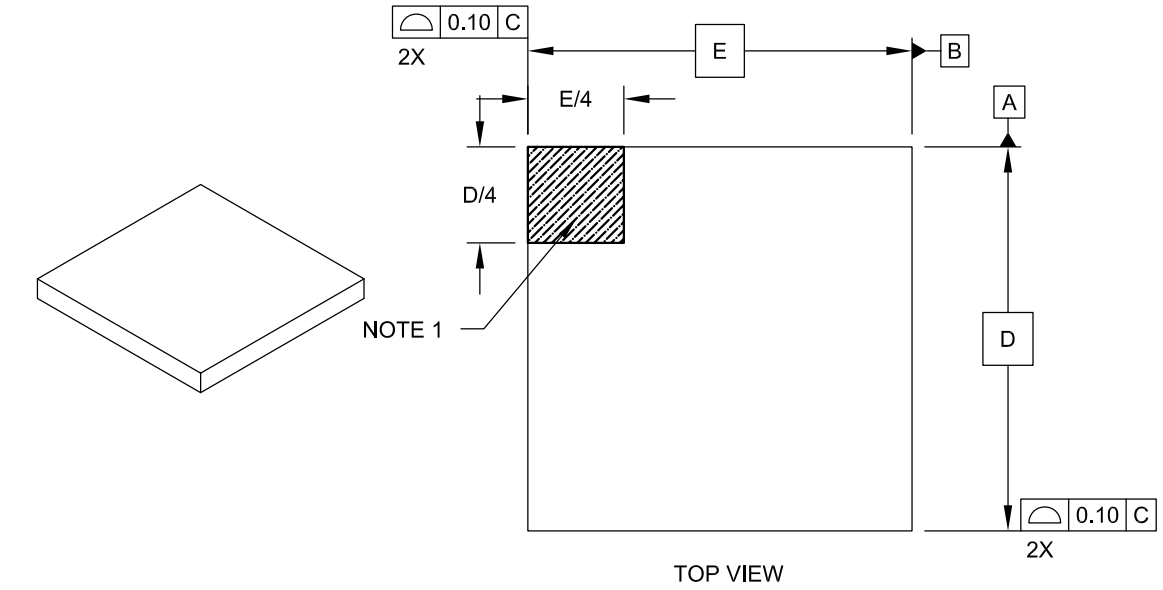
1. Dimensioning and tolerancing per ASME Y14.5M

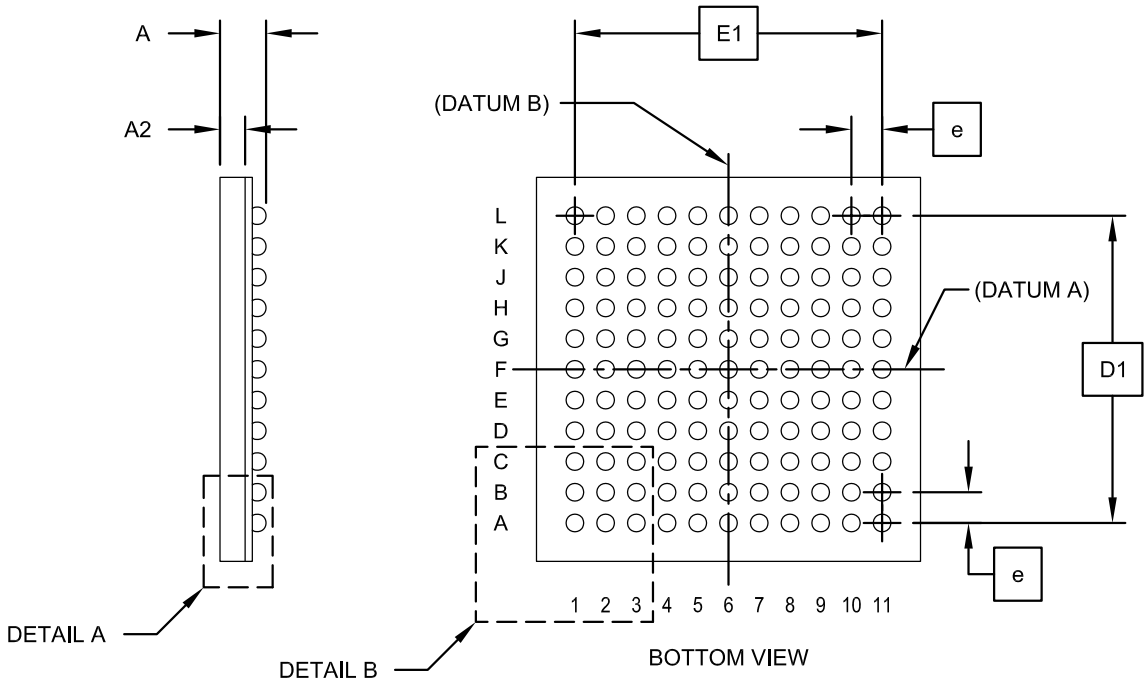
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2100B

121-Lead Plastic Thin Profile Ball Grid Array (BG) - 10x10x1.10 mm Body [TFBGA--Formerly XBGA]

Note:	For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging

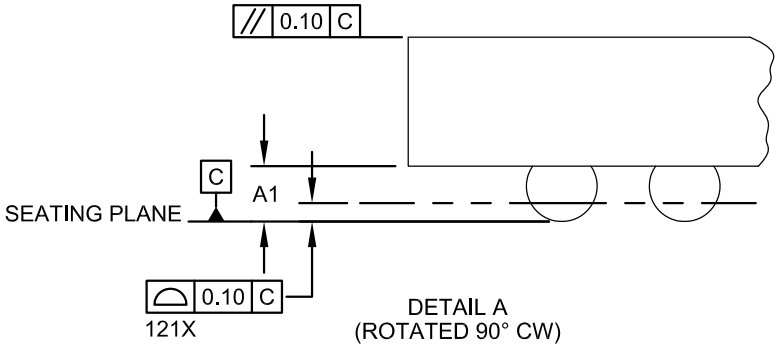


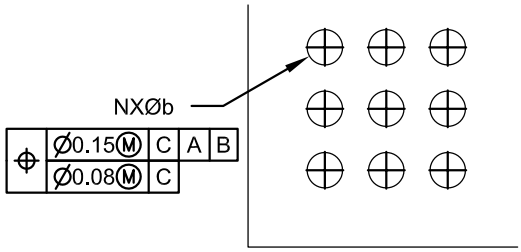


Microchip Technology Drawing C04-148 Rev D Sheet 1 of 2

121-Lead Plastic Thin Profile Ball Grid Array (BG) - 10x10x1.10 mm Body [TFBGA–Formerly XBGA]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging





DETAIL B

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Contacts	N		121	
Contact Pitch	e		0.80 BSC	
Overall Height	A	1.00	1.10	1.20
Standoff	A1	0.25	0.30	0.35
Molded Package Thickness	A2	0.55	0.60	0.65
Overall Width	E		10.00 BSC	
Array Width	E1		8.00 BSC	
Overall Length	D		10.00 BSC	
Array Length	D1		8.00 BSC	
Contact Diameter	b		0.40 TYP	

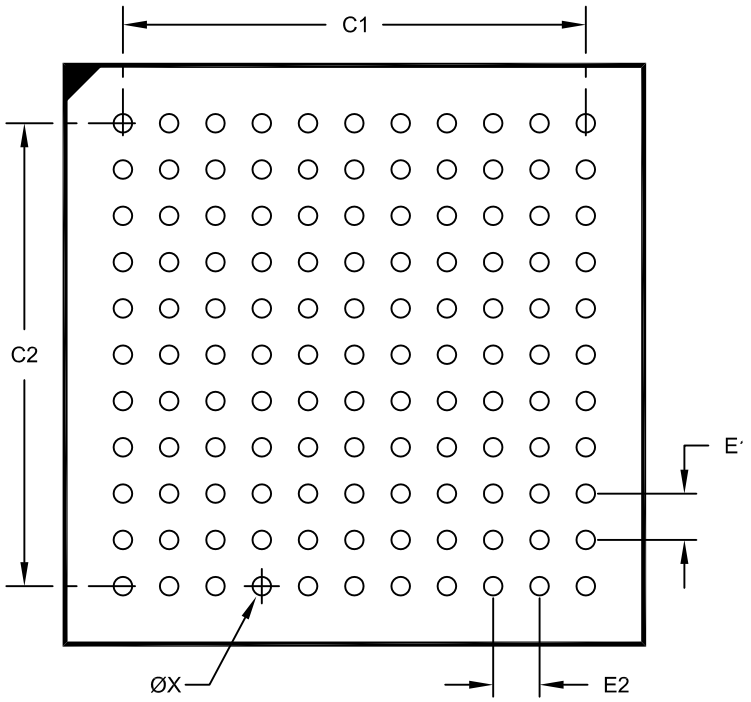
Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Dimensioning and tolerancing per ASME Y14.5M.
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
   REF: Reference Dimension, usually without tolerance, for information purposes only.
3. The outer rows and colums of balls are located with respect to datums A and B.

Microchip Technology Drawing C04-148 Rev D Sheet 2 of 2

121-Lead Plastic Thin Profile Ball Grid Array (BG) - 10x10x1.10 mm Body [TFBGA--Formerly XBGA]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E1		0.80 BSC	
Contact Pitch	E2		0.80 BSC	
Contact Pad Spacing	C1		8.00	
Contact Pad Spacing	C2		8.00	
Contact Pad Diameter (X121)	X			0.32

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2148 Rev D

APPENDIX A: REVISION HISTORY

Revision A (July 2012)

Original data sheet for the PIC24FJ128GC010 family of devices.

Revision B (May 2013)

Changes descriptive title on Page 1 to “16-Bit Flash Microcontrollers with 12-Bit Pipeline A/D, Sigma-Delta A/D, USB On-The-Go and XLP Technology”.

Adds CoreMark® rating to the **“High-Performance CPU”** section on Page 2.

Removes all references to JTAG device programming throughout the document.

Corrects the default Doze mode ratio as 1:8 (previously described as 1:1) throughout the document.

Corrects the default FRC postscaler setting to 1:2.

Corrects references in **Section 10.4.6 “Deep Sleep WDT”** regarding the Configuration register for the DSWDTOSC and DSWDPS<4:0> bits.

Changes the description of the behavior of the UERRIF bit in the U1IR register, from “Read-Only” to “Read, Write 1 to Clear”, in both contexts of the register.

Corrects the low end of the operating range of the voltage regulator, described in **Section 34.2 “On-Chip Voltage Regulator”**, to 2.0V.

Updates **Section 37.0 “Electrical Characteristics”**:

- Adds maximum specifications to most DC Specifications
- Adds systematic parameter numbers to existing DC and AC Specifications that were previously not numbered
- Moves DC Specification for USB module from Table 37-4 to a new Table 37-15; all subsequent tables are renumbered accordingly
- Updates most typical and maximum specifications in the following tables:
  - Table 37-12 (Band Gap Reference (BGBUFn) Specifications)
  - Table 37-19 (Operational Amplifier Specifications)
  - Table 37-28 (12-bit Pipeline A/D Module Specifications)
  - Table 37-30 (10-Bit DAC Specifications)
  - Table 37-31 (16-Bit Sigma-Delta A/D Converter Specifications)

Other minor typographic changes and updates throughout.

NOTES:

INDEX

A

A/D
- Operation ... 347
- Registers ... 348

AC Characteristics
- 10-Bit DAC Converter ... 444
- 12-Bit Pipeline A/D Conversion Timing Requirements ... 444
- 12-Bit Pipeline A/D Module ... 442
- 16-Bit Sigma-Delta A/D Converter ... 445
- CLKO and I/O Timing Requirements ... 440
- External Clock Timing ... 438
- Internal RC Accuracy ... 439
- Load Conditions and Requirements for Specifications ... 437
- PLL Clock Timing ... 439
- RC Oscillator Start-up Time ... 439
- Reset and Brown-out Reset Requirements ... 441

Advanced Analog Features ... 14, 341
- Shared Analog Pins ... 341

Alternate Interrupt Vector Table (AIVT) ... 103

Assembler
- MPASM Assembler ... 414

B

Block Diagrams
- 12-Bit A/D Converter Single-Ended Analog Input Model ... 367
- 12-Bit Pipeline A/D Converter ... 349
- 16-Bit Asynchronous Timer3 and Timer5 ... 219
- 16-Bit Synchronous Timer2 and Timer4 ... 219
- 16-Bit Timer1 Module ... 215
- 32-Bit Timer2/3 and Timer4/5 ... 218
- Accessing Program Memory Using Table Instructions ... 80
- Addressing for Table Registers ... 91
- Analog Block (overview) ... 342
- BDT Mapping for Endpoint Buffering Modes ... 270
- CALL Stack Frame ... 77
- Comparator Voltage Reference ... 387
- CPU Programmer's Model ... 41
- CRC Module ... 335
- CRC Shift Engine Detail ... 335
- CTMU Connections and Internal Configuration for Capacitance Measurement ... 390
- CTMU Typical Connections and Internal Configuration for Pulse Delay Generation ... 391
- CTMU Typical Connections and Internal Configuration for Time Measurement ... 391
- Data Access from Program Space Address Generation ... 79
- Data Signal Modulator ... 299
- DMA Controller ... 83
- EDS Address Generation for Read ... 75
- EDS Address Generation for Write ... 76
- High/Low-Voltage Detect (HLVD) ... 397
- I^2C Module ... 250
- Individual Comparator Configurations, CREF = 0 ... 382
- Individual Comparator Configurations, CREF = 1, CVREFP = 0 ... 383
- Individual Comparator Configurations, CREF = 1, CVREFP = 1 ... 383
- Input Capture x ... 223
- LCD Controller ... 315
- MCLR Pin Connections Example ... 34
- On-Chip Regulator Connections ... 409
- Output Compare x (16-Bit Mode) ... 228
- Output Compare x (Double-Buffered, 16-Bit PWM Mode) ... 230
- PCI24FJ128GC010 Family (General) ... 18
- PIC24F CPU Core ... 40
- PSV Operation (Lower Word) ... 82
- PSV Operation (Upper Word) ... 82
- Recommended Minimum Connections ... 33
- Reset System ... 97
- RTCC ... 323
- Shared I/O Port Structure ... 183
- Sigma-Delta A/D Converter ... 369
- Simplified Single DAC ... 375
- Single Operational Amplifier ... 377
- SPIx Master, Frame Master Connection ... 246
- SPIx Master, Frame Slave Connection ... 246
- SPIx Master/Slave Connection (Enhanced Buffer Modes) ... 245
- SPIx Master/Slave Connection (Standard Mode) ... 245
- SPIx Module (Enhanced Mode) ... 239
- SPIx Module (Standard Mode) ... 238
- SPIx Slave, Frame Master Connection ... 246
- SPIx Slave, Frame Slave Connection ... 246
- System Clock ... 159
- Triple Comparator Module ... 381
- UARTx (Simplified) ... 257
- USB OTG Bus-Powered Interface Example ... 267
- USB OTG Dual Power Mode Example ... 267
- USB OTG Host Interface Example ... 268
- USB OTG Interface Example ... 268
- USB OTG Interrupt Funnel ... 274
- USB OTG Module ... 266
- USB OTG Self-Power Only Mode ... 267
- USB PLL ... 167
- Watchdog Timer (WDT) ... 410

C

C Compilers
- MPLAB C18 ... 414

Charge Time Measurement Unit. *See* CTMU.

Code Examples
- Basic Clock Switching ... 166
- Configuring UART1 Input/Output Functions (PPS) ... 196
- EDS Read From Program Memory in Assembly Code ... 81
- EDS Read in Assembly Code ... 75
- EDS Write in Assembly Code ... 76
- Erasing a Program Memory Block (Assembly) ... 94
- Erasing a Program Memory Block (C Language) ... 95
- Initiating a Programming Sequence ... 95
- Loading the Write Buffers ... 95
- Port Write/Read in Assembly ... 191
- Port Write/Read in C ... 191
- PWRSAV Instruction Syntax ... 172
- Setting the RTCWREN Bit ... 324
- Single-Word Flash Programming ... 96
- Single-Word Flash Programming (C Language) ... 96
- The Repeat Sequence ... 175

Code Protection 411
Code Segment Protection 411
Configuration Options 411
Configuration Protection 412
General Segment Protection 411
Comparator Voltage Reference 387
Configuring 387
Configuration Bits 399
Core Features 13
CPU
Arithmetic Logic Unit (ALU) 44
Clocking Scheme 160
Control Registers 42
Core Registers 40
Programmer's Model 39
CRC
Polynomials 336
Setup Examples for 16 and 32-Bit Polynomials 336
User Interface 336
CTMU
Measuring Capacitance 389
Measuring Time 390
Pulse Delay and Generation 390
Customer Change Notification Service 467
Customer Notification Service 467
Customer Support 467

D

Data Memory
Address Space 47
Extended Data Space (EDS) 74
Memory Map 47
Near Data Space 48
SFR Space 48
Software Stack 77
Space Organization, Alignment 48
Data Signal Modulator 299
Data Signal Modulator (DSM) 299
DC Characteristics
Comparator 435
Comparator Voltage Reference 435
CTMU Current Source 434
Delta Current (BOR, WDT, DSBOR, DSWDT, LCD) 430
I/O Pin Input Specifications 431
I/O Pin Output Specifications 432
Idle Current (IIDLE) 428
Operating Current (IDD) 428
Power-Down Current (IPD) 429
Program Memory 432
Temperature and Voltage Specifications 427
Deep Sleep WDT 176
Development Support 413
Device Features
100/121-Pin 17
64-Pin 16
Direct Memory Access (DMA) Controller 14
Direct Memory Access Controller. *See* DMA.
DMA 83
Channel Trigger Sources 90
Peripheral Module Disable (PMD) 86
Summary of Operations 84
Types of Data Transfers 85
Typical Setup 86
Dual Operational Amplifier 377

E

Electrical Characteristics
Absolute Maximum Ratings 425
Band Gap (BGBUFn) Reference 433
Capacitive Loading on Output Pins 437
High/Low-Voltage Detect 435
Internal Voltage Regulator 433
Operational Amplifier 436
Thermal Operating Conditions 426
Thermal Packaging Characteristics 426
V/F Graph 426
VBAT Operating Voltage 434
Enhanced Parallel Master Port (EPMP) 303
Enhanced Parallel Master Port. *See* EPMP.
EPMP
Key Features 303
Package Variations 303
Equations
16-Bit, 32-Bit CRC Polynomials 336
Baud Rate Reload Calculation 251
Calculating the PWM Period 230
Calculation for Maximum PWM Resolution 231
Estimating USB Transceiver Current Consumption 269
Relationship Between Device and SPIx Clock Speed 247
UARTx Baud Rate with BRGH = 0 258
UARTx Baud Rate with BRGH = 1 258
Errata 12
Extended Data Space (EDS) 303

F

Flash Configuration Word Locations 399
Flash Configuration Words 46
Flash Program Memory 91
and Table Instructions 91
Enhanced ICSP Operation 92
Programming Algorithm 94
Programming Operations 94
RTSP Operation 92
Single-Word Programming 96

G

Getting Started
Basic Connection Requirements 33
External Oscillator Pins 37
ICSP Operation
Analog and Digital Pin Configuration 38
Pins 36
Master Clear Reset ($\overline{MCLR}$) Pin 34
Power Supply Pins 34
Voltage Regulator (VCAP) 35

H

High/Low-Voltage Detect (HLVD) 397

I

I/O Ports
- Analog Port Pins Configuration (ANSx) 184
- Analog/Digital Function of an I/O Pin 184
- Input Change Notification (ICN) 191
- Input Voltage Levels for Port/Pin Tolerated Description Input 185
- Open-Drain Configuration 184
- Parallel (PIO) 183
- Peripheral Pin Select 192
- Pull-ups and Pull-Downs 191
- Selectable Input Sources 193
- Selectable Output Sources 194

I^2C
- Clock Rates 251
- Communicating as Master in Single Master Environment 249
- Reserved Addresses 251
- Setting Baud Rate as Bus Master 251
- Slave Address Masking 251

Input Capture
- 32-Bit Cascaded Mode 224
- Operations 224
- Synchronous and Trigger Modes 223

Input Capture with Dedicated Timers 223
Instruction Set
- Overview 419
- Summary 417
- Symbols Used in Opcode Descriptions 418

Interfacing Program and Data Spaces 78
Inter-Integrated Circuit. *See* I^2C.
Internal Band Gap References 341
Internet Address 467
Interrupt Controller 103
Interrupt Vector Table (IVT) 103
Interrupts
- Control and Status Registers 107
- Implemented Vectors 105
- Reset Sequence 103
- Setup and Service Procedures 158
- Trap Vectors 104
- Vector Table 104

J

JTAG Interface 412

K

Key Features 399

L

LCD Controller 14
Liquid Crystal Display (LCD) Controller 315
Low-Voltage/Retention Regulator 173

M

Memory Organization 45
Microchip Internet Web Site 467
Modulator. *See* Data Signal Modulator.
MPLAB ASM30 Assembler, Linker, Librarian 414
MPLAB Integrated Development Environment Software 413
MPLAB PM3 Device Programmer 415
MPLAB REAL ICE In-Circuit Emulator System 415
MPLINK Object Linker/MPLIB Object Librarian 414

N

Near Data Space 48

O

On-Chip Voltage Regulator 409
- POR 409
- Standby Mode 409

Oscillator Configuration
- Clock Switching 165
  - Sequence 165
- Configuration Bit Values for Clock Selection 160
- FRC Self-Tuning 166
- Initial Configuration on POR 160
- USB Operation 167
  - Special Considerations 168

Output Compare
- 32-Bit Cascaded Mode 227
- Operations 228
- Synchronous and Trigger Modes 227

Output Compare with Dedicated Timers 227

P

Packaging 447
- Details 449
- Marking 447

Peripheral Pin Select (PPS) 192
- Available Peripherals and Pins 192
- Configuration Control 195
- Considerations for Use 196
- Input Mapping 193
- Mapping Exceptions 195
- Output Mapping 194
- Peripheral Priority 192
- Registers 197

Pin Descriptions
- 100-Pin Devices 6
- 121-Pin Devices (BGA) 9
- 64-Pin Devices 4

Pinout Descriptions 19–31
Power-Saving Features 171
- Clock Frequency and Clock Switching 181
- Doze Mode 181
- Hardware-Based Modes 173
- Instruction-Based Modes 172
  - Deep Sleep 174
    - I/O Pins 175
    - Retention Mode 174
  - Idle 173
  - Sleep 173
    - Low-Voltage/Retention Sleep 173
- Power-on Resets (PORs) 176
- Selective Peripheral Control 181
- Vbat Mode 177
  - I/O Pins 177
  - With no RTCC 177

Product Identification System 469
Program Memory
- Access Using Table Instructions 80
- Address Construction 78
- Address Space 45
- Flash Configuration Words 46
- Hard Memory Vectors 46
- Memory Maps 45
- Organization 46
- Reading From Program Memory Using EDS 81

Program Verification 411
Pulse-Width Modulation (PWM) Mode 229
Pulse-Width Modulation. *See* PWM.
PWM
    Duty Cycle and Period 230

R

Reader Response 468
Real-Time Clock and Calendar (RTCC) 323
Reference Clock Output 168
Register Maps
    12-Bit Pipeline A/D Converter 61
    Analog Configuration 60
    Band Gap Buffer Interface 70
    Comparators 70
    CPU Core 49
    CTMU 59
    Cyclic Redundancy Check (CRC) 70
    DAC 60
    Data Signal Modulator (DSM) 69
    Deep Sleep 73
    DMA 65
    I^2C 55
    ICN 50
    Input Capture 53
    Interrupt Controller 51
    LCD Controller 67
    NVM 73
    Op Amp 57
    Output Compare 54
    PADCFG1 (Pad Configuration) 59
    Parallel Master/Slave Port 69
    Peripheral Module Disable (PMD) 73
    Peripheral Pin Select (PPS) 71
    PORTA 57
    PORTB 57
    PORTC 58
    PORTD 58
    PORTE 58
    PORTF 59
    PORTG 59
    Real-Time Clock and Calendar (RTCC) 69
    Sigma-Delta A/D 60
    SPI 56
    System Control (Clock and Reset) 72
    Timers 52
    UART 56
    USB OTG 66
Registers
    ACCONH (A/D Accumulator Control High) 362
    ACCONL (A/D Accumulator Control Low) 362
    ADCHITH (A/D Match Hit High) 363
    ADCHITL (A/D Match Hit Low) 363
    ADCON1 (A/D Control 1) 350
    ADCON2 (A/D Control 2) 351
    ADCON3 (A/D Control 3) 352
    ADLnCONH (A/D Sample List n Control High) 355
    ADLnCONL (A/D Sample List n Control Low) 357
    ADLnMSEL0 (A/D Sample List n Multi-Channel Select 0) 366
    ADLnMSEL1 (A/D Sample List n Multi-Channel Select 1) 366
    ADLnMSEL2 (A/D Sample List n Multi-Channel Select 2) 365
    ADLnMSEL3 (A/D Sample List n Multi-Channel Select 3) 365
    ADLnPTR (A/D Sample List n Pointer) 360
    ADLnSTAT (A/D Sample List n Status) 359
    ADSTATH (A/D Status High) 353
    ADSTATL (A/D Status Low) 354
    ADTBLn (A/D Sample Table Entry n) 360
    ADTHnH (A/D Sample Table n Threshold Value High) 364
    ADTHnL (A/D Sample Table n Threshold Value Low) 364
    ALCFGRPT (Alarm Configuration) 328
    ALMINSEC (Alarm Minutes and Seconds Value) 332
    ALMTHDY (Alarm Month and Day Value) 331
    ALWDHR (Alarm Weekday and Hours Value) 331
    AMPxCON (Op Amp x Control) 378
    ANSA (PORTA Analog Function Selection) 186
    ANSB (PORTB Analog Function Selection) 187
    ANSC (PORTC Analog Function Selection) 187
    ANSD (PORTD Analog Function Selection) 188
    ANSE (PORTE Analog Function Selection) 188
    ANSF (PORTF Analog Function Selection) 189
    ANSG (PORTG Analog Function Selection) 190
    BDnSTAT Prototype (Buffer Descriptor n Status, CPU Mode) 273
    BDnSTAT Prototype (Buffer Descriptor n Status, USB Mode) 272
    BUFCON0 (Internal Voltage Reference Control) 344
    BUFCONx (Band Gap Buffers 1, 2 Control) 345
    CLKDIV (Clock Divider) 163
    CMSTAT (Comparator Status) 385
    CMxCON (Comparator x Control, Comparators 1-3) 384
    CORCON (CPU Control) 109
    CORCON (CPU Core Control) 43
    CRCCON1 (CRC Control 1) 338
    CRCCON2 (CRC Control 2) 339
    CRCXORH (CRC XOR Polynomial, High Byte) 340
    CRCXORL (CRC XOR Polynomial, Low Byte) 339
    CTMUCON1 (CTMU Control 1) 392
    CTMUCON2 (CTMU Control 2) 393
    CTMUICON (CTMU Current Control) 395
    CVRCON (Comparator Voltage Reference Control) 388
    CW1 (Flash Configuration Word 1) 400
    CW2 (Flash Configuration Word 2) 402
    CW3 (Flash Configuration Word 3) 404
    CW4 (Flash Configuration Word 4) 406
    DACxCON (DACx Control) 376
    DEVID (Device ID) 408
    DEVREV (Device Revision) 408
    DMACHn (DMA Channel n Control) 88
    DMACON (DMA Engine Control) 87
    DMAINTn (DMA Channel n Interrupt) 89
    DSCON (Deep Sleep Control) 178
    DSWAKE (Deep Sleep Wake-up Source) 179
    HLVDCON (High/Low-Voltage Detect Control) 398
    I2CxCON (I2Cx Control) 252
    I2CxMSK (I2Cx Slave Mode Address Mask) 255
    I2CxSTAT (I2Cx Status) 254
    ICxCON1 (Input Capture x Control 1) 225
    ICxCON2 (Input Capture x Control 2) 226
    IEC0 (Interrupt Enable Control 0) 122
    IEC1 (Interrupt Enable Control 1) 124
    IEC2 (Interrupt Enable Control 2) 126
    IEC3 (Interrupt Enable Control 3) 127

IEC4 (Interrupt Enable Control 4) 128
IEC5 (Interrupt Enable Control 5) 129
IEC6 (Interrupt Enable Control 6) 130
IEC7 (Interrupt Enable Control 7) 131
IFS0 (Interrupt Flag Status 0) 112
IFS1 (Interrupt Flag Status 1) 114
IFS2 (Interrupt Flag Status 2) 116
IFS3 (Interrupt Flag Status 3) 117
IFS4 (Interrupt Flag Status 4) 118
IFS5 (Interrupt Flag Status 5) 119
IFS6 (Interrupt Flag Status 6) 120
IFS7 (Interrupt Flag Status 7) 121
INTCON1 (Interrupt Control 1)..... 110
INTCON2 (Interrupt Control 2)..... 111
INTTREG (Interrupt Controller Test)..... 157
IPC0 (Interrupt Priority Control 0) 132
IPC1 (Interrupt Priority Control 1) 133
IPC10 (Interrupt Priority Control 10) 142
IPC11 (Interrupt Priority Control 11) 143
IPC12 (Interrupt Priority Control 12) 144
IPC13 (Interrupt Priority Control 13) 145
IPC15 (Interrupt Priority Control 15) 146
IPC16 (Interrupt Priority Control 16) 147
IPC18 (Interrupt Priority Control 18) 148
IPC19 (Interrupt Priority Control 19) 149
IPC2 (Interrupt Priority Control 2) 134
IPC20 (Interrupt Priority Control 20) 150
IPC21 (Interrupt Priority Control 21) 151
IPC22 (Interrupt Priority Control 22) 152
IPC23 (Interrupt Priority Control 23) 153
IPC25 (Interrupt Priority Control 25) 154
IPC26 (Interrupt Priority Control 26) 155
IPC29 (Interrupt Priority Control 29) 156
IPC3 (Interrupt Priority Control 3) 135
IPC4 (Interrupt Priority Control 4) 136
IPC5 (Interrupt Priority Control 5) 137
IPC6 (Interrupt Priority Control 6) 138
IPC7 (Interrupt Priority Control 7) 139
IPC8 (Interrupt Priority Control 8) 140
IPC9 (Interrupt Priority Control 9) 141
LCDCON (LCD Control)..... 316
LCDDATAx (LCD Data x) 319
LCDPS (LCD Phase) 318
LCDREF (LCD Reference Ladder Control)..... 321
LCDREG (LCD Charge Pump Control)..... 317
LCDSEx (LCD Segment x Enable) 319
MDCAR (Data Signal Modulator Carrier Control) 302
MDCON (Data Signal Modulator Control)..... 300
MDSRC (Data Signal Modulator Source Control)..... 301
MINSEC (RTCC Minutes and Seconds Value)..... 330
MTHDY (RTCC Month and Day Value) 329
NVMCON (Flash Memory Control) 93
OCxCON1 (Output Compare x Control 1) 232
OCxCON2 (Output Compare x Control 2) 234
OSCCON (Oscillator Control) 161
OSCTUN (FRC Oscillator Tune)..... 164
PADCFG1 (Pad Configuration Control) 313
PMCON1 (EPMP Control 1) 305
PMCON2 (EPMP Control 2) 306
PMCON3 (EPMP Control 3) 307
PMCON4 (EPMP Control 4) 308
PMCSxBS (EPMP Chip Select x Base Address)..... 310
PMCSxCF (EPMP Chip Select x Configuration)..... 309
PMCSxMD (EPMP Chip Select x Mode) 311
PMSTAT (EPMP Status, Slave Mode) 312
RCFGCAL (RTCC Calibration and Configuration) 325
RCON (Reset Control)..... 98
RCON2 (Reset and System Control 2)..... 180
RCON2 (Reset Control 2)..... 100
REFOCON (Reference Oscillator Control)..... 169
RPINR0 (PPS Input 0)..... 197
RPINR1 (PPS Input 1)..... 197
RPINR10 (PPS Input 10)..... 200
RPINR11 (PPS Input 11)..... 200
RPINR15 (PPS Input 15)..... 201
RPINR17 (PPS Input 17)..... 201
RPINR18 (PPS Input 18)..... 202
RPINR19 (PPS Input 19)..... 202
RPINR2 (PPS Input 2)..... 198
RPINR20 (PPS Input 20)..... 203
RPINR21 (PPS Input 21)..... 203
RPINR22 (PPS Input 22)..... 204
RPINR23 (PPS Input 23)..... 204
RPINR27 (PPS Input 27)..... 205
RPINR30 (PPS Input 30)..... 205
RPINR31 (PPS Input 31)..... 206
RPINR7 (PPS Input 7)..... 198
RPINR8 (PPS Input 8)..... 199
RPINR9 (PPS Input 9)..... 199
RPOR0 (PPS Output 0)..... 207
RPOR1 (PPS Output 1)..... 207
RPOR10 (PPS Output 10)..... 212
RPOR11 (PPS Output 11)..... 212
RPOR12 (PPS Output 12)..... 213
RPOR13 (PPS Output 13)..... 213
RPOR14 (PPS Output 14)..... 214
RPOR15 (PPS Output 15)..... 214
RPOR2 (PPS Output 2)..... 208
RPOR3 (PPS Output 3)..... 208
RPOR4 (PPS Output 4)..... 209
RPOR5 (PPS Output 5)..... 209
RPOR6 (PPS Output 6)..... 210
RPOR7 (PPS Output 7)..... 210
RPOR8 (PPS Output 8)..... 211
RPOR9 (PPS Output 9)..... 211
RTCPWC (RTCC Power Control)..... 327
SD1CON1 (S/D Control 1)..... 371
SD1CON2 (S/D Control 2)..... 372
SD1CON3 (S/D Control 3)..... 373
SPIxCON1 (SPIx Control 1) 242
SPIxCON2 (SPIx Control 2) 244
SPIxSTAT (SPIx Status and Control) 240
SR (ALU STATUS) 42, 108
T1CON (Timer1 Control) 216
TxCON (Timer2 and Timer4 Control) 220
TyCON (Timer3 and Timer5 Control) 222
U1ADDR (USB Address)..... 287
U1CNFG1 (USB Configuration 1)..... 289
U1CNFG2 (USB Configuration 2)..... 290
U1CON (USB Control, Device Mode)..... 285
U1CON (USB Control, Host Mode) 286
U1EIE (USB Error Interrupt Enable)..... 297
U1EIR (USB Error Interrupt Status)..... 296
U1EPn (USB Endpoint n Control)..... 298
U1IE (USB Interrupt Enable, All Modes) 295
U1IR (USB Interrupt Status, Device Mode)..... 293
U1IR (USB Interrupt Status, Host Mode)..... 294
U1OTGCON (USB OTG Control)..... 282

U1OTGIE (USB OTG Interrupt Enable, Host Mode) ... 292
U1OTGIR (USB OTG Interrupt Status, Host Mode) ... 291
U1OTGSTAT (USB OTG Status, Host Mode) ... 281
U1PWRC (USB Power Control) ... 283
U1SOF (USB OTG Start-of-Token Threshold, Host Mode) ... 288
U1STAT (USB Status) ... 284
U1TOK (USB Token, Host Mode) ... 287
UxMODE (UARTx Mode) ... 260
UxSTA (UARTx Status and Control) ... 262
WKDYHR (RTCC Weekday and Hours Value) ... 330
YEAR (RTCC Year Value) ... 329
Resets
BOR (Brown-out Reset) ... 97
Brown-out Reset (BOR) ... 101
Clock Source Selection ... 101
CM (Configuration Mismatch Reset) ... 97
Delay Times ... 102
Device Times ... 101
IOPUWR (Illegal Opcode Reset) ... 97
MCLR (Pin Reset) ... 97
POR (Power-on Reset) ... 97
RCON Flags, Operation ... 100
SFR States ... 101
SWR (RESET Instruction) ... 97
TRAPR (Trap Conflict Reset) ... 97
UWR (Uninitialized W Register Reset) ... 97
WDT (Watchdog Timer Reset) ... 97
Revision History ... 459
RTCC
Alarm Configuration ... 333
Alarm Mask Settings (figure) ... 334
Calibration ... 333
Clock Source Selection ... 324
Power Control ... 334
Register Mapping ... 324
Source Clock ... 323
VBAT OPERATION ... 334
Write Lock ... 324

S

Secondary Oscillator Operation ... 168
Serial Peripheral Interface (SPI) ... 237
Serial Peripheral Interface. *See* SPI.
SFR Space ... 48
Sigma-Delta A/D
Differences Compared to Conventional A/Ds ... 370
Sigma-Delta A/D Converter
Required Connections ... 38
Software Simulator (MPLAB SIM) ... 415
Software Stack ... 77
Special Features ... 15
SPI ... 237

T

Timer1 ... 215
Timer2/3 and Timer4/5 ... 217
Timing Diagrams
12-Bit A/D DNL ... 443
12-Bit A/D INL ... 443
16-Bit S/D A/D DNL ... 446
16-Bit S/D A/D INL ... 446
CLKO and I/O Timing ... 440
External Clock ... 438
Triple Comparator ... 381
Triple Comparator Module ... 381

U

UART ... 257
UARTx
Baud Rate Generator (BRG) ... 258
Infrared Support ... 259
Operation of UxCTS and UxRTS Pins ... 259
Receiving
8-Bit or 9-Bit Data Mode ... 259
Transmitting
8-Bit Data Mode ... 259
9-Bit Data Mode ... 259
Break and Sync Sequence ... 259
Universal Asynchronous Receiver Transmitter. *See* UART.
Universal Serial Bus
Interrupts
and USB Transactions ... 275
Universal Serial Bus. *See* USB OTG.
Unused I/Os ... 38
USB On-The-Go (OTG) ... 14
USB OTG ... 265
Buffer Descriptors
Assignment in Different Buffering Modes ... 271
Buffer Descriptors and BDT ... 270
Device Mode Operation ... 275
DMA Interface ... 271
Hardware
Calculating
Transceiver Power Requirements ... 269
Hardware Configuration ... 267
Device Mode ... 267
External Interface ... 269
Host and OTG Modes ... 268
VBUS Voltage Generation ... 269
Host Mode Operation ... 276
Interrupts ... 274
Operation ... 278
HNP ... 279
SRP ... 278
Registers ... 280
VBUS Voltage Generation ... 269

W

Watchdog Timer (WDT) ... 410
Control Register ... 410
Windowed Operation ... 410
WWW Address ... 467
WWW, On-Line Support ... 12

THE MICROCHIP WEB SITE

Microchip provides online support via our WWW site at www.microchip.com. This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at www.microchip.com. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support
- Development Systems Information Line

Customers should contact their distributor, representative or field application engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the web site at: http://microchip.com/support

READER RESPONSE

It is our intention to provide you with the best documentation possible to ensure successful use of your Microchip product. If you wish to provide your comments on organization, clarity, subject matter, and ways in which our documentation can better serve you, please FAX your comments to the Technical Publications Manager at (480) 792-4150.

Please list the following information, and use this outline to provide us with your comments about this document.

TO: Technical Publications Manager
RE: Reader Response

Total Pages Sent _____

From: Name _____

Company _____

Address _____

City / State / ZIP / Country _____

Telephone: (_____) _____ - _____ FAX: (_____) _____ - _____

Application (optional):

Would you like a reply? ___Y ___N

Device: PIC24FJ128GC010 Family Literature Number: DS30009312B

Questions:

1. What are the best features of this document?

2. How does this document meet your hardware and software development needs?

3. Do you find the organization of this document easy to follow? If not, why?

4. What additions to the document do you think would enhance the structure and subject?

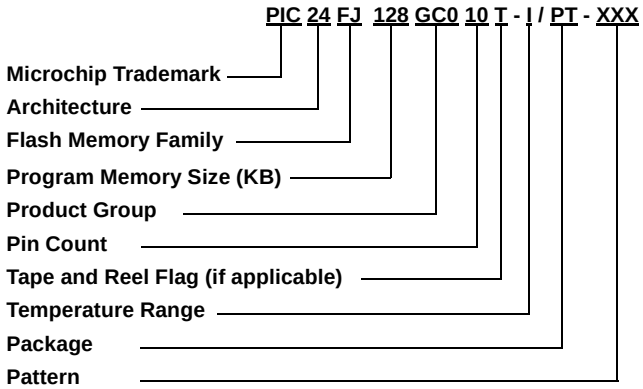
5. What deletions from the document could be made without affecting the overall usefulness?

6. Is there any incorrect or misleading information (what and where)?

7. How would you improve this document?

PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.



Architecture	24	= 16-bit modified Harvard without DSP
Flash Memory Family	FJ	= Flash program memory
Product Group	GC0	= Advanced Analog Microcontrollers with USB OTG, LCD Controller and XLP Technology
Pin Count	06	= 64-pin
	10	= 100-pin (TQFP) and 121-pin (BGA)
Temperature Range	I	= -40°C to +85°C (Industrial)
Package	BG	= 121-pin (10x10x1.10 mm) BGA package
	PT	= 100-lead (12x12x1 mm) TQFP (Thin Quad Flatpack)
	PT	= 64-lead (10x10x1 mm) TQFP (Thin Quad Flatpack)
	MR	= 64-lead (9x9x0.9 mm) QFN (Quad Flatpack, No Lead)
Pattern		Three-digit QTP, SQTP, Code or Special Requirements (blank otherwise)
	ES	= Engineering Sample

Examples:

a) PIC24FJ64GC006-I/MR:
PIC24F device with Advanced Analog, LCD Controller and XLP Technology, 64-Kbyte Program Memory, 64-pin, Industrial temp., QFN package.

b) PIC24FJ128GC010-I/PT:
PIC24F device with Advanced Analog, LCD Controller and XLP Technology, 128-Kbyte Program Memory, 100-pin, Industrial temp., TQFP package.

c) PIC24FJ128GC010-I/BG:
PIC24F device with Advanced Analog, LCD Controller and XLP Technology, 128-Kbyte Program Memory, 121-pin, Industrial temp., BGA package.

NOTES:

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Trademarks

Printed on recycled paper.

ISBN: 978-1-62077-224-9

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
═ ISO/TS 16949 ═

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://www.microchip.com/support
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Cleveland
Independence, OH
Tel: 216-447-0464
Fax: 216-447-0643

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Farmington Hills, MI
Tel: 248-538-2250
Fax: 248-538-2260

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608

Santa Clara
Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

Toronto
Mississauga, Ontario,
Canada
Tel: 905-673-0699
Fax: 905-673-6509

ASIA/PACIFIC

Asia Pacific Office
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

China - Beijing
Tel: 86-10-8569-7000
Fax: 86-10-8528-2104

China - Chengdu
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

China - Chongqing
Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

China - Hangzhou
Tel: 86-571-2819-3187
Fax: 86-571-2819-3189

China - Hong Kong SAR
Tel: 852-2943-5100
Fax: 852-2401-3431

China - Nanjing
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

China - Qingdao
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

China - Shanghai
Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

China - Shenyang
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

China - Shenzhen
Tel: 86-755-8864-2200
Fax: 86-755-8203-1760

China - Wuhan
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

China - Xian
Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

China - Xiamen
Tel: 86-592-2388138
Fax: 86-592-2388130

China - Zhuhai
Tel: 86-756-3210040
Fax: 86-756-3210049

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-3090-4444
Fax: 91-80-3090-4123

India - New Delhi
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

India - Pune
Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

Japan - Osaka
Tel: 81-6-6152-7160
Fax: 81-6-6152-9310

Japan - Tokyo
Tel: 81-3-6880- 3770
Fax: 81-3-6880-3771

Korea - Daegu
Tel: 82-53-744-4301
Fax: 82-53-744-4302

Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 or
82-2-558-5934

Malaysia - Kuala Lumpur
Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

Malaysia - Penang
Tel: 60-4-227-8870
Fax: 60-4-227-4068

Philippines - Manila
Tel: 63-2-634-9065
Fax: 63-2-634-9069

Singapore
Tel: 65-6334-8870
Fax: 65-6334-8850

Taiwan - Hsin Chu
Tel: 886-3-5778-366
Fax: 886-3-5770-955

Taiwan - Kaohsiung
Tel: 886-7-213-7828
Fax: 886-7-330-9305

Taiwan - Taipei
Tel: 886-2-2508-8600
Fax: 886-2-2508-0102

Thailand - Bangkok
Tel: 66-2-694-1351
Fax: 66-2-694-1350

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

UK - Wokingham
Tel: 44-118-921-5869
Fax: 44-118-921-5820

11/29/12