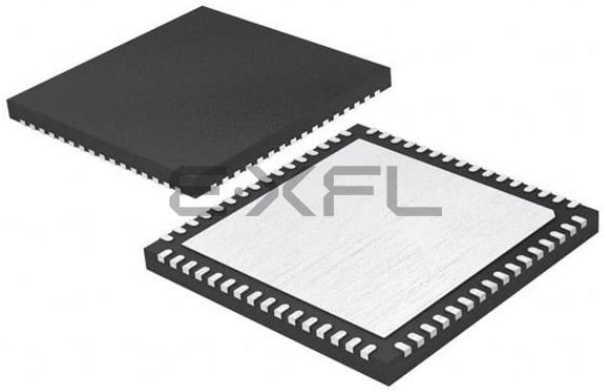


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	PIC
Core Size	16-Bit
Speed	32MHz
Connectivity	I²C, IrDA, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, LCD, POR, PWM, WDT
Number of I/O	53
Program Memory Size	64KB (22K x 24)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 3.6V
Data Converters	A/D 16x10b/12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-VFQFN Exposed Pad
Supplier Device Package	64-VQFN (9x9)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic24fj64ga306t-i-mr

TABLE 1-4: PIC24FJ128GA310 FAMILY PINOUT DESCRIPTIONS (CONTINUED)

Pin Function	Pin Number/Grid Locater				I/O	Input Buffer	Description
	64-Pin TQFP	80-Pin TQFP	100-Pin TQFP	121-Pin BGA			
PMD0	60	76	93	A4	I/O	ST/TTL	Parallel Master Port Data (Demultiplexed Master mode) or Address/Data (Multiplexed Master modes).
PMD1	61	77	94	B4	I/O	ST/TTL	
PMD2	62	78	98	B3	I/O	ST/TTL	
PMD3	63	79	99	A2	I/O	ST/TTL	
PMD4	64	80	100	A1	I/O	ST/TTL	
PMD5	1	1	3	D3	I/O	ST/TTL	
PMD6	2	2	4	C1	I/O	ST/TTL	
PMD7	3	3	5	D2	I/O	ST/TTL	
PMD8	—	75	90	A5	I/O	ST/TTL	
PMD9	—	74	89	E6	I/O	ST/TTL	
PMD10	—	73	88	A6	I/O	ST/TTL	
PMD11	—	72	87	B6	I/O	ST/TTL	
PMD12	—	64	79	A9	I/O	ST/TTL	
PMD13	—	65	80	D8	I/O	ST/TTL	
PMD14	—	68	83	D7	I/O	ST/TTL	
PMD15	—	69	84	C7	I/O	ST/TTL	
PMRD	53	67	82	B8	O	—	Parallel Master Port Read Strobe.
PMWR	52	66	81	C8	O	—	Parallel Master Port Write Strobe.
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	—	23	28	L2	I/O	ST	
RA10	—	24	29	K3	I/O	ST	
RA14	—	52	66	E11	I/O	ST	
RA15	—	53	67	E8	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

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, 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 PIC24FJ128GA310 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 PIC24FJ128GA310 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 29.0 "Special Features"**.

TABLE 4-1: FLASH CONFIGURATION WORDS FOR PIC24FJ128GA310 FAMILY DEVICES

Device	Program Memory (Words)	Configuration Word Addresses
PIC24FJ64GA3XX	22,016	00ABF8h:00ABFEh
PIC24FJ128GA3XX	44,032	0157F8h:0157FEh

FIGURE 4-2: PROGRAM MEMORY ORGANIZATION

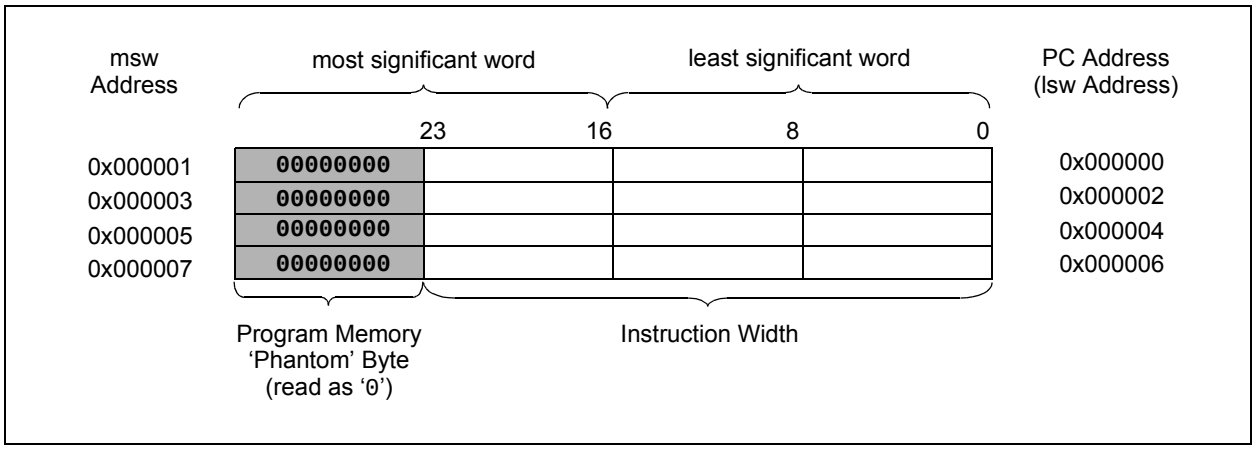


TABLE 4-29: CRC REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
CRCCON1	0640	CRCEN	—	CSIDL	VWORD4	VWORD3	VWORD2	VWORD1	VWORD0	CRCFUL	CRCMPT	CRCISEL	CRCGO	LENDIAN	—	—	—	0040
CRCCON2	0642	—	—	—	DWIDTH4	DWIDTH3	DWIDTH2	DWIDTH1	DWIDTH0	—	—	—	PLEN4	PLEN3	PLEN2	PLEN1	PLEN0	0000
CRCXORL	0644	X<15:1>															—	0000
CRCXORH	0646	X<31:16>																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 4-30: 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	—	—	—	—	—	—	—	—	3F3F
RPINR1	0682	—	—	INT3R5	INT3R4	INT3R3	INT3R2	INT3R1	INT3R0	—	—	INT2R5	INT2R4	INT2R3	INT2R2	INT2R1	INT2R0	3F3F
RPINR2	0684	—	—	—	—	—	—	—	—	—	—	INT4R5	INT4R4	INT4R3	INT4R2	INT4R1	INT4R0	3F3F
RPINR3	0686	—	—	T3CKR5	T3CKR4	T3CKR3	T3CKR2	T3CKR1	T3CKR0	—	—	T2CKR5	T2CKR4	T2CKR3	T2CKR2	T2CKR1	T2CKR0	3F3F
RPINR4	0688	—	—	T5CKR5	T5CKR4	T5CKR3	T5CKR2	T5CKR1	T5CKR0	—	—	T4CKR5	T4CKR4	T4CKR3	T4CKR2	T4CKR1	T4CKR0	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	—	—	—	—	—	—	—	—	—	—	IC7R5	IC7R4	IC7R3	IC7R2	IC7R1	IC7R0	003F
RPINR11	0696	—	—	OCFBR5	OCFBR4	OCFBR3	OCFBR2	OCFBR1	OCFBR0	—	—	OCFAR5	OCFAR4	OCFAR3	OCFAR2	OCFAR1	OCFAR0	3F3F
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	—	—	T1CKR5	T1CKR4	T1CKR3	T1CKR2	T1CKR1	T1CKR0	—	—	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'.

2: These bits are unimplemented in 64-pin and 80-pin devices, read as '0'.

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.

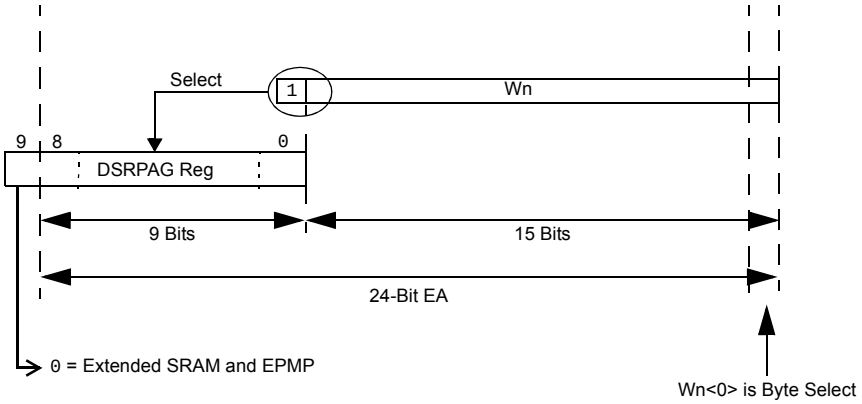
Figure 4-5 illustrates how the EDS space address is generated for read operations.

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 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 are 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
```

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 = Use Alternate Interrupt Vector Table
0 = Use 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

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. 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 Semaphores and RTCC 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 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.

With VBPOR set, the user should clear it, and the next time, this bit will only set when VDD = 0 and the VBAT pin has gone below level VBTRST.

10.5.3 I/O PINS DURING VBAT MODES

All I/O pins should be maintained at VSS level; no I/O pins should be given VDD (refer to **"Absolute Maximum Ratings"** in **Section 32.0 "Electrical Characteristics"**) 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, 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, the recommendation is to connect the VBAT pin to VDD and connect a 0.1 µF capacitor close to the VBAT pin to ground. When the VBAT mode is used (connected to the battery), it is suggested to connect a 0.1 µF capacitor from the VBAT pin to ground. The capacitor should be located very close to the VBAT pin.

The BOR should be enabled for the reliable operation of the VBAT.

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 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)"**.

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 **"Serial Peripheral Interface (SPI)"** (DS39699) in the *"dsPIC33/PIC24 Family Reference Manual"*. 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, ADC Converters, etc. The SPI module is compatible with the SPI and SIOP Motorola® interfaces. All PIC24FJ128GA310 family devices include two SPI modules.

The module supports operation in two buffer modes. In Standard 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 2 SPI modules.

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 16-3: 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:** 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:** 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:** 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:** Enhanced Buffer Enable bit
1 = Enhanced buffer is enabled
0 = Enhanced buffer is disabled (Legacy mode)

REGISTER 19-2: MDSRC: 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:** Modulation Source Output Disable bit(1)
1 = Output signal driving the peripheral output pin (selected by MDMS<3:0>) is disabled
0 = Output signal driving the peripheral output pin (selected by MDMS<3:0>) is enabled

bit 6-4 **Unimplemented:** Read as '0'

bit 3-0 **MS<3:0>** 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.

22.3 RTCC Control Registers

REGISTER 22-1: 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.

22.4 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 form 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 22-1:

(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.

22.5 Alarm

- Configurable from half second to one year
- Enabled using the ALRMEN bit (ALCFGRPT<15>)
- One-time alarm and repeat alarm options available

22.5.1 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 22-2, the interval selection of the alarm is configured through the AMASKx 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.

22.5.2 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.

REGISTER 23-2: 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 23-3: 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
X<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0
X<7:1>							—
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 15-1 **X<15:1>:** XOR of Polynomial Term x^n Enable bits

bit 0 **Unimplemented:** Read as ‘0’

REGISTER 24-5: AD1CON5: ADC1 CONTROL REGISTER 5

R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
ASEN	LPEN	CTMREQ	BGREQ	—	—	ASINT1	ASINT0
bit 15							bit 8

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	WM1	WM0	CM1	CM0
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 **ASEN:** Auto-Scan Enable bit
1 = Auto-scan is enabled
0 = Auto-scan is disabled

bit 14 **LPEN:** Low-Power Enable bit
1 = Low power is enabled after scan
0 = Full power is enabled after scan

bit 13 **CTMREQ:** CTMU Request bit
1 = CTMU is enabled when the ADC is enabled and active
0 = CTMU is not enabled by the ADC

bit 12 **BGREQ:** Band Gap Request bit
1 = Band gap is enabled when the ADC is enabled and active
0 = Band gap is not enabled by the ADC

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

bit 9-8 **ASINT<1:0>:** Auto-Scan (Threshold Detect) Interrupt Mode bits
11 = Interrupt after Threshold Detect sequence completed and valid compare has occurred
10 = Interrupt after valid compare has occurred
01 = Interrupt after Threshold Detect sequence completed
00 = No interrupt

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

bit 3-2 **WM<1:0>:** Write Mode bits
11 = Reserved
10 = Auto-compare only (conversion results are not saved, but interrupts are generated when a valid match occurs, as defined by the CMx and ASINTx bits)
01 = Convert and save (conversion results are saved to locations as determined by the register bits when a match occurs, as defined by the CMx bits)
00 = Legacy operation (conversion data is saved to a location determined by the buffer register bits)

bit 1-0 **CM<1:0>:** Compare Mode bits
11 = Outside Window mode (valid match occurs if the conversion result is outside of the window defined by the corresponding buffer pair)
10 = Inside Window mode (valid match occurs if the conversion result is inside the window defined by the corresponding buffer pair)
01 = Greater Than mode (valid match occurs if the result is greater than the value in the corresponding buffer register)
00 = Less Than mode (valid match occurs if the result is less than the value in the corresponding buffer register)

29.3 Watchdog Timer (WDT)

For PIC24FJ128GA310 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, ranging 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 NOSC 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>) bits 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.

29.3.1 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<7>) to '0'.

29.3.2 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 29-2: WDT BLOCK DIAGRAM

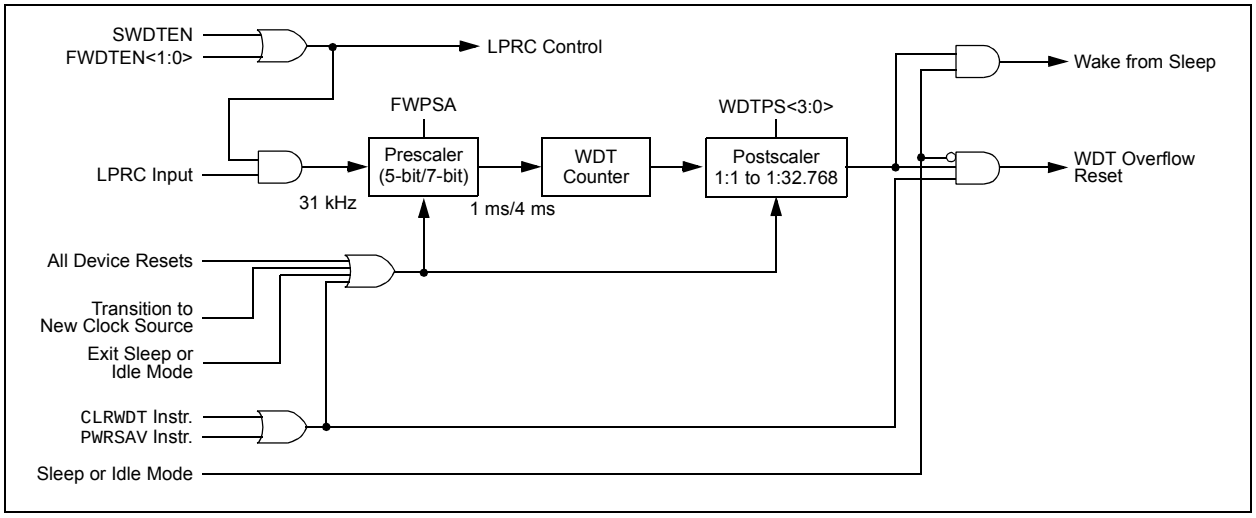


TABLE 32-37: SPIx MODULE SLAVE MODE TIMING REQUIREMENTS (CKE = 1) (CONTINUED)

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
SP52	TscH2ssH TscL2ssH	$\overline{SSx}$ ↑ after SCKx Edge	1.5 TCY + 40	—	—	ns	
SP60	TssL2doV	SDOx Data Output Valid after $\overline{SSx}$ Edge	—	—	50	ns	

Note 1: Data in "Typ" column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

2: The minimum clock period for SCKx is 100 ns. Therefore, the clock generated in Master mode must not violate this specification.

3: Assumes 50 pF load on all SPIx pins.

FIGURE 32-18: UARTx BAUD RATE GENERATOR OUTPUT TIMING

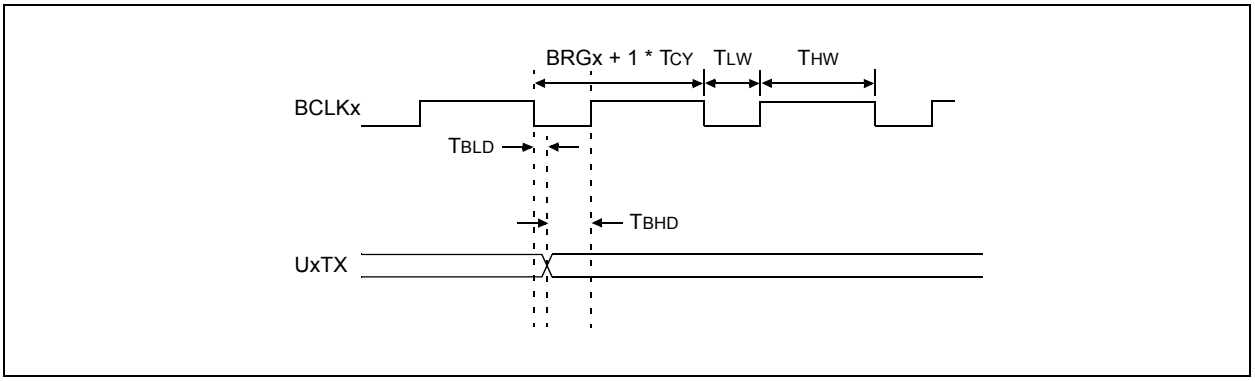
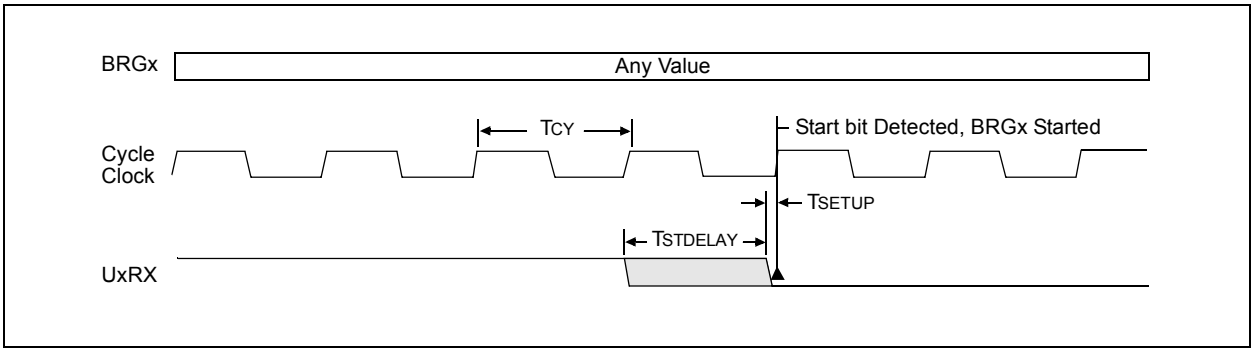
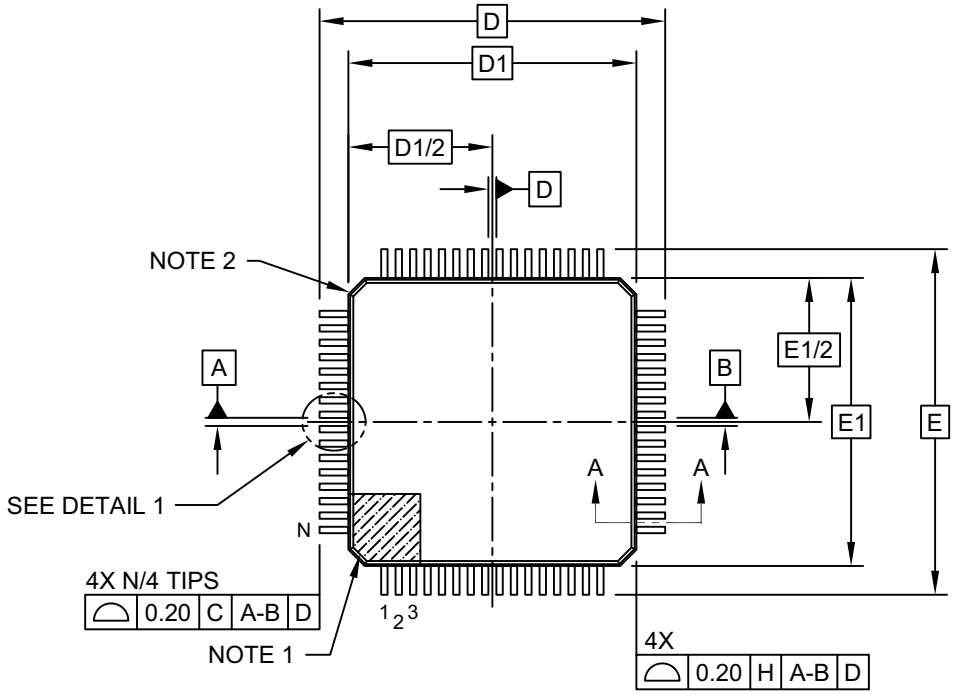


FIGURE 32-19: UARTx START BIT EDGE DETECTION

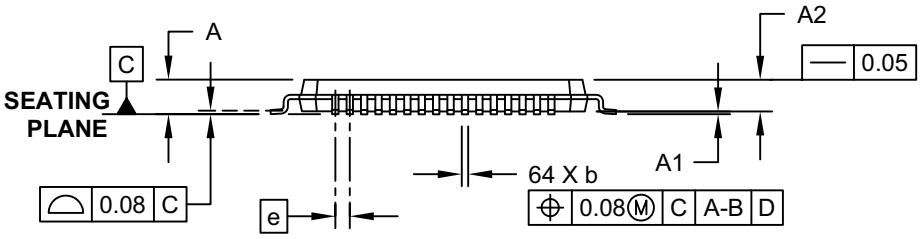


64-Lead Plastic Thin Quad Flatpack (PT)-10x10x1 mm Body, 2.00 mm Footprint [TQFP]

Note:	For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



TOP VIEW



SIDE VIEW

Microchip Technology Drawing C04-085C Sheet 1 of 2