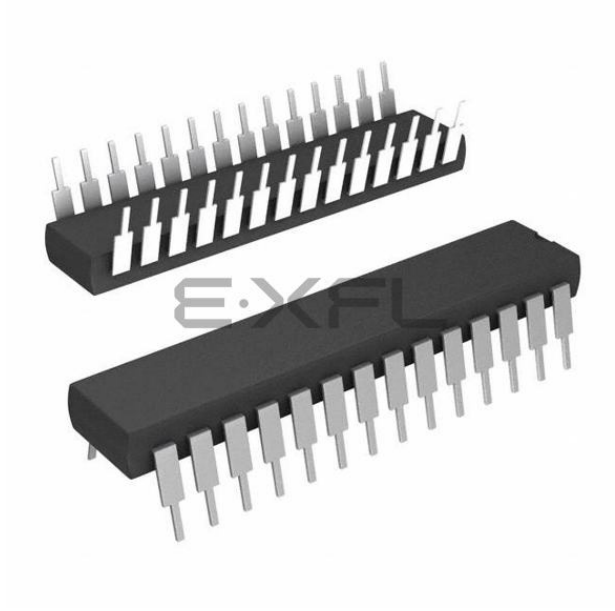


Welcome to E-XFL.COM

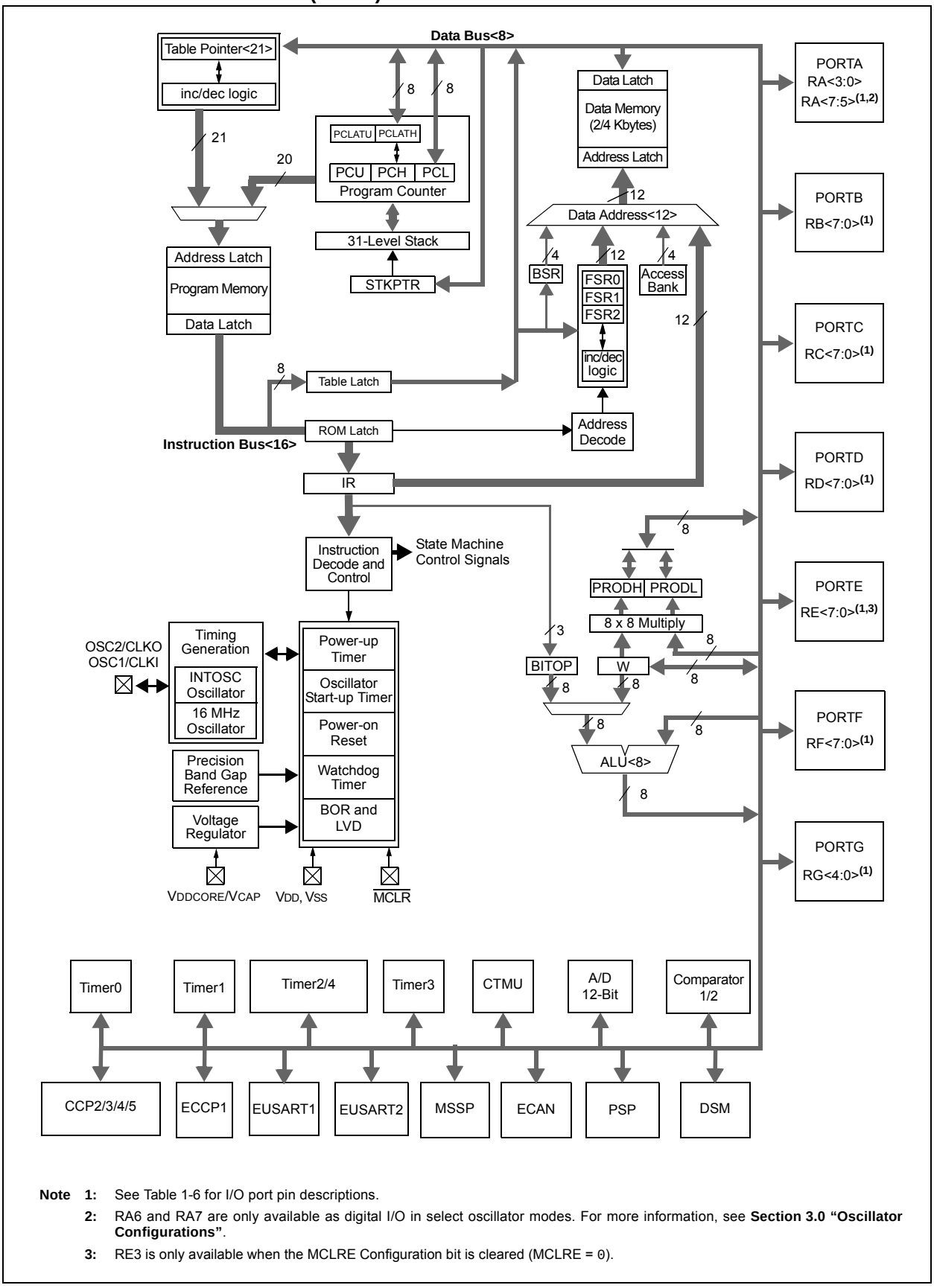
What is "Embedded - Microcontrollers"?

"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	8-Bit
Speed	64MHz
Connectivity	ECANbus, I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LVD, POR, PWM, WDT
Number of I/O	24
Program Memory Size	64KB (32K x 16)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	3.6K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 8x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Through Hole
Package / Case	28-DIP (0.300", 7.62mm)
Supplier Device Package	28-SPDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18lf26k80-i-sp

FIGURE 1-3: PIC18F6XK80 (64-PIN) BLOCK DIAGRAM



Note 1: See Table 1-6 for I/O port pin descriptions.

2: RA6 and RA7 are only available as digital I/O in select oscillator modes. For more information, see **Section 3.0 "Oscillator Configurations"**.

3: RE3 is only available when the MCLRE Configuration bit is cleared (MCLRE = 0).

5.0 RESET

The PIC18F66K80 family devices differentiate between various kinds of Reset:

a) Power-on Reset (POR)
b) $\overline{\text{MCLR}}$ Reset during Normal Operation
c) $\overline{\text{MCLR}}$ Reset during Power-Managed modes
d) Watchdog Timer (WDT) Reset (during execution)
e) Configuration Mismatch (CM) Reset
f) Programmable Brown-out Reset (BOR)
g) RESET Instruction
h) Stack Full Reset
i) Stack Underflow Reset

This section discusses Resets generated by $\overline{\text{MCLR}}$, POR and BOR, and covers the operation of the various start-up timers. Stack Reset events are covered in **Section 6.1.3.4 "Stack Full and Underflow Resets"**. WDT Resets are covered in **Section 28.2 "Watchdog Timer (WDT)"**.

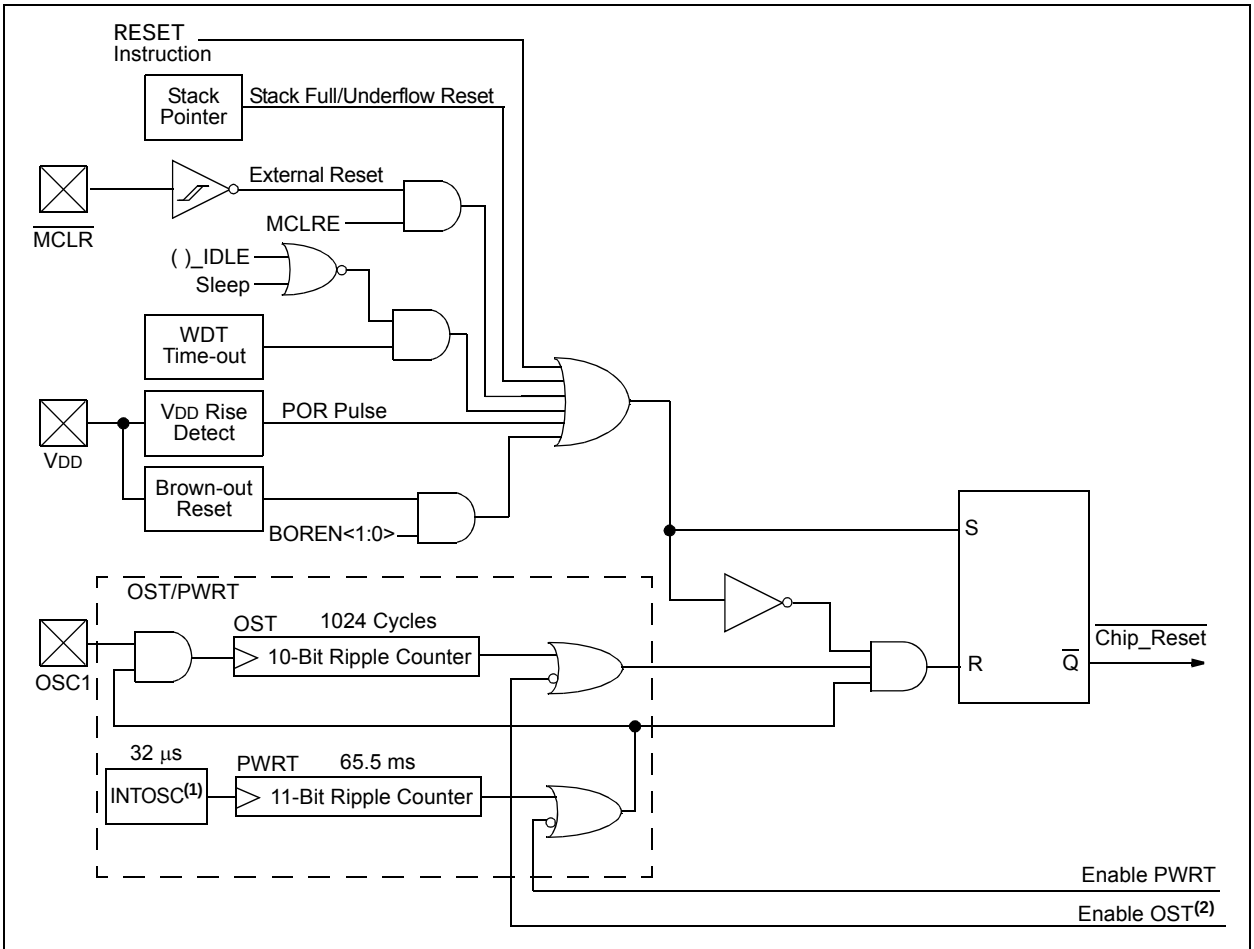
A simplified block diagram of the On-Chip Reset Circuit is shown in Figure 5-1.

5.1 RCON Register

Device Reset events are tracked through the RCON register (Register 5-1). The lower five bits of the register indicate that a specific Reset event has occurred. In most cases, these bits can only be cleared by the event and must be set by the application after the event. The state of these flag bits, taken together, can be read to indicate the type of Reset that just occurred. This is described in more detail in **Section 5.7 "Reset State of Registers"**.

The RCON register also has control bits for setting interrupt priority (IPEN) and software control of the BOR (SBOREN). Interrupt priority is discussed in **Section 10.0 "Interrupts"**. BOR is covered in **Section 5.4 "Brown-out Reset (BOR)"**.

FIGURE 5-1: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT



Note 1: This is the INTOSC source from the internal oscillator block and is separate from the RC oscillator of the CLKI pin.
2: See Table 5-2 for time-out situations.

TABLE 5-4: INITIALIZATION CONDITIONS FOR ALL REGISTERS (CONTINUED)

Register	Applicable Devices			Power-on Reset, Brown-out Reset	MCLR Resets, WDT Reset, RESET Instruction, Stack Resets	Wake-up via WDT or Interrupt
B4D5	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B4D4	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B4D3	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B4D2	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B4D1	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B4D0	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B4DLC	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	-xxx xxxx	-uuu uuuu	-uuu uuuu
B4EIDL	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B4EIDH	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B4SIDL	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx x-xx	uuuu u-uu	uuuu u-uu
B4SIDH	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B4CON	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	0000 0000	0000 0000	uuuu uuuu
CANCON_RO6	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	1000 0000	1000 0000	uuuu uuuu
CANSTAT_RO6	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	1000 0000	1000 0000	uuuu uuuu
B3D7	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B3D6	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B3D5	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B3D4	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B3D3	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B3D2	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B3D1	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B3D0	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B3DLC	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	-xxx xxxx	-uuu uuuu	-uuu uuuu
B3EIDL	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B3EIDH	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B3SIDL	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx x-xx	uuuu u-uu	uuuu u-uu
B3SIDH	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B3CON	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	0000 0000	0000 0000	uuuu uuuu
CANCON_RO7	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	1000 0000	1000 0000	uuuu uuuu
B2D7	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B2D6	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B2D5	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B2D4	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B2D3	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B2D2	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu

Legend: u = unchanged; x = unknown; - = unimplemented bit, read as '0'; q = value depends on condition.
Shaded cells indicate conditions do not apply for the designated device.

Note 1: One or more bits in the INTCONx or PIRx registers will be affected (to cause wake-up).

2: When the wake-up is due to an interrupt and the GIEL or GIEH bit is set, the PC is loaded with the interrupt vector (0008h or 0018h).

3: When the wake-up is due to an interrupt and the GIEL or GIEH bit is set, the TOSU, TOSH and TOSL are updated with the current value of the PC. The STKPTR is modified to point to the next location in the hardware stack.

4: See Table 5-3 for Reset value for specific conditions.

5: Bits 6 and 7 of PORTA, LATA and TRISA are enabled, depending on the oscillator mode selected. When not enabled as PORTA pins, they are disabled and read as '0'.

TABLE 5-4: INITIALIZATION CONDITIONS FOR ALL REGISTERS (CONTINUED)

Register	Applicable Devices			Power-on Reset, Brown-out Reset	MCLR Resets, WDT Reset, RESET Instruction, Stack Resets	Wake-up via WDT or Interrupt
B0EIDH	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B0SIDL	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx x-xx	uuuu u-uu	uuuu u-uu
B0SIDH	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B0CON	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	0000 0000	0000 0000	uuuu uuuu
TXBIE	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	---0 00--	---u uu--	---u uu--
BIE0	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	0000 0000	0000 0000	uuuu uuuu
BSEL0	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	0000 00--	0000 00--	uuuu uu--
MSEL3	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	0000 0000	0000 0000	uuuu uuuu
MSEL2	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	0000 0000	0000 0000	uuuu uuuu
MSEL1	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	0000 0101	0000 0101	uuuu uuuu
MSEL0	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	0101 0000	0101 0000	uuuu uuuu
RXFBCON7	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	0000 0000	0000 0000	uuuu uuuu
RXFBCON6	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	0000 0000	0000 0000	uuuu uuuu
RXFBCON5	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	0000 0000	0000 0000	uuuu uuuu
RXFBCON4	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	0000 0000	0000 0000	uuuu uuuu
RXFBCON3	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	0000 0000	0000 0000	uuuu uuuu
RXFBCON2	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	0001 0001	0001 0001	uuuu uuuu
RXFBCON1	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	0001 0001	0001 0001	uuuu uuuu
RXFBCON0	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	0000 0000	0000 0000	uuuu uuuu
SDFLC	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	---0 0000	---0 0000	---u uuuu
RXF15EIDL	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
RXF15EIDH	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
RXF15SIDL	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxx- x-xx	uuu- u-uu	uuu- u-uu
RXF15SIDH	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
RXF14EIDL	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
RXF14EIDH	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
RXF14SIDL	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxx- x-xx	uuu- u-uu	uuu- u-uu
RXF14SIDH	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
RXF13EIDL	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
RXF13EIDH	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
RXF13SIDL	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxx- x-xx	uuu- u-uu	uuu- u-uu
RXF13SIDH	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
RXF12EIDL	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
RXF12EIDH	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
RXF12SIDL	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxx- x-xx	uuu- u-uu	uuu- u-uu

Legend: u = unchanged; x = unknown; - = unimplemented bit, read as '0'; q = value depends on condition.
Shaded cells indicate conditions do not apply for the designated device.

Note 1: One or more bits in the INTCONx or PIRx registers will be affected (to cause wake-up).

2: When the wake-up is due to an interrupt and the GIEL or GIEH bit is set, the PC is loaded with the interrupt vector (0008h or 0018h).

3: When the wake-up is due to an interrupt and the GIEL or GIEH bit is set, the TOSU, TOSH and TOSL are updated with the current value of the PC. The STKPTR is modified to point to the next location in the hardware stack.

4: See Table 5-3 for Reset value for specific conditions.

5: Bits 6 and 7 of PORTA, LATA and TRISA are enabled, depending on the oscillator mode selected. When not enabled as PORTA pins, they are disabled and read as '0'.

TABLE 6-2: PIC18F66K80 FAMILY REGISTER FILE SUMMARY

Addr.	File Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR on page
FFFh	TOSU	—	—	—	Top-of-Stack Upper Byte (TOS<20:16>)					88
FFEh	TOSH	Top-of-Stack High Byte (TOS<15:8>)								88
FFDh	TOSL	Top-of-Stack Low Byte (TOS<7:0>)								88
FFCh	STKPTR	STKFUL	STKUNF	—	SP4	SP3	SP2	SP1	SP0	88
FFBh	PCLATU	—	—	Bit 21	Holding Register for PC<20:16>					88
FFAh	PCLATH	Holding Register for PC<15:8>								88
FF9h	PCL	PC Low Byte (PC<7:0>)								88
FF8h	TBLPTRU	—	—	Bit 21	Program Memory Table Pointer Upper Byte (TBLPTR<20:16>)					88
FF7h	TBLPTRH	Program Memory Table Pointer High Byte (TBLPTR<15:8>)								88
FF6h	TBLPTRL	Program Memory Table Pointer Low Byte (TBLPTR<7:0>)								88
FF5h	TABLAT	Program Memory Table Latch								88
FF4h	PRODH	Product Register High Byte								88
FF3h	PRODL	Product Register Low Byte								88
FF2h	INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	88
FF1h	INTCON2	$\overline{\text{RBPU}}$	INTEDG0	INTEDG1	INTEDG2	INTEDG3	TMR0IP	INT3IP	RBIP	88
FF0h	INTCON3	INT2IP	INT1IP	INT3IE	INT2IE	INT1IE	INT3IF	INT2IF	INT1IF	88
FEFh	INDF0	Uses contents of FSR0 to address data memory – value of FSR0 not changed (not a physical register)								88
FEEh	POSTINC0	Uses contents of FSR0 to address data memory – value of FSR0 post-incremented (not a physical register)								88
FEDh	POSTDEC0	Uses contents of FSR0 to address data memory – value of FSR0 post-decremented (not a physical register)								88
FECh	PREINC0	Uses contents of FSR0 to address data memory – value of FSR0 pre-incremented (not a physical register)								88
FEBh	PLUSW0	Uses contents of FSR0 to address data memory – value of FSR0 pre-incremented (not a physical register) – value of FSR0 offset by W								88
FEAh	FSR0H	—	—	—	—	Indirect Data Memory Address Pointer 0 High Byte				88
FE9h	FSR0L	Indirect Data Memory Address Pointer 0 Low Byte								88
FE8h	WREG	Working Register								88
FE7h	INDF1	Uses contents of FSR1 to address data memory – value of FSR1 not changed (not a physical register)								88
FE6h	POSTINC1	Uses contents of FSR1 to address data memory – value of FSR1 post-incremented (not a physical register)								88
FE5h	POSTDEC1	Uses contents of FSR1 to address data memory – value of FSR1 post-decremented (not a physical register)								88
FE4h	PREINC1	Uses contents of FSR1 to address data memory – value of FSR1 pre-incremented (not a physical register)								88
FE3h	PLUSW1	Uses contents of FSR1 to address data memory – value of FSR1 pre-incremented (not a physical register) – value of FSR1 offset by W								88
FE2h	FSR1H	—	—	—	—	Indirect Data Memory Address Pointer 1 High Byte				88
FE1h	FSR1L	Indirect Data Memory Address Pointer 1 Low Byte								88
FE0h	BSR	—	—	—	—	Bank Select Register				88
FDFh	INDF2	Uses contents of FSR2 to address data memory – value of FSR2 not changed (not a physical register)								88
FDEh	POSTINC2	Uses contents of FSR2 to address data memory – value of FSR2 post-incremented (not a physical register)								89
FDDh	POSTDEC2	Uses contents of FSR2 to address data memory – value of FSR2 post-decremented (not a physical register)								89
FDCh	PREINC2	Uses contents of FSR2 to address data memory – value of FSR2 pre-incremented (not a physical register)								89
FDBh	PLUSW2	Uses contents of FSR2 to address data memory – value of FSR2 pre-incremented (not a physical register) – value of FSR2 offset by W								89
FDAh	FSR2H	—	—	—	—	Indirect Data Memory Address Pointer 2 High Byte				89
FD9h	FSR2L	Indirect Data Memory Address Pointer 2 Low Byte								89
FD8h	STATUS	—	—	—	N	OV	Z	DC	C	89
FD7h	TMR0H	Timer0 Register High Byte								89
FD6h	TMR0L	Timer0 Register Low Byte								89
FD5h	T0CON	TMR0ON	T08BIT	T0CS	T0SE	PSA	T0PS2	T0PS1	T0PS0	89
FD4h	Unimplemented									—
FD3h	OSCCON	IDLEN	IRCF2	IRCF1	IRCF0	OSTS	HFIOFS	SCS1	SCS0	89
FD2h	OSCCON2	—	SOSCRUN	—	SOSCDRV	SOSCGO	—	MFIOFS	MFIOSEL	89
FD1h	WDTCON	REGSLP	—	ULPLVL	SRETEN	—	ULPEN	ULPSINK	SWDTEN	89
FD0h	RCON	IPEN	SBOREN	$\overline{\text{CM}}$	$\overline{\text{RI}}$	$\overline{\text{TO}}$	$\overline{\text{PD}}$	$\overline{\text{POR}}$	$\overline{\text{BOR}}$	89

TABLE 6-2: PIC18F66K80 FAMILY REGISTER FILE SUMMARY (CONTINUED)

Addr.	File Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR on page
ED6h	B5D0	B5D07	B5D06	B5D05	B5D04	B5D03	B5D02	B5D01	B5D00	95
ED5h	B5DLC	—	TXRTR	—	—	DLC3	DLC2	DLC1	DLC0	95
ED4h	B5EIDL	EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0	95
ED3h	B5EIDH	EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8	95
ED2h	B5SIDL	SID2	SID1	SID0	SRR	EXID	—	EID17	EID16	95
ED1h	B5SIDH	SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3	95
ED0h	B5CON	TXBIF	TXABT	TXLARB	TXERR	TXREQ	—	TXPRI1	TXPRI0	95
ECFh	CANCON_RO5	CANCON_RO5								95
ECEh	CANSTAT_RO5	CANSTAT_RO5								96
ECDh	B4D7	B4D77	B4D76	B4D75	B4D74	B4D73	B4D72	B4D71	B4D70	96
ECCh	B4D6	B4D67	B4D66	B4D65	B4D64	B4D63	B4D62	B4D61	B4D60	96
ECBh	B4D5	B4D57	B4D56	B4D55	B4D54	B4D53	B4D52	B4D51	B4D50	96
ECAh	B4D4	B4D47	B4D46	B4D45	B4D44	B4D43	B4D42	B4D41	B4D40	96
EC9h	B4D3	B4D37	B4D36	B4D35	B4D34	B4D33	B4D32	B4D31	B4D30	96
EC8h	B4D2	B4D27	B4D26	B4D25	B4D24	B4D23	B4D22	B4D21	B4D20	96
EC7h	B4D1	B4D17	B4D16	B4D15	B4D14	B4D13	B4D12	B4D11	B4D10	96
EC6h	B4D0	B4D07	B4D06	B4D05	B4D04	B4D03	B4D02	B4D01	B4D00	96
EC5h	B4DLC	—	TXRTR	—	—	DLC3	DLC2	DLC1	DLC0	96
EC4h	B4EIDL	EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0	96
EC3h	B4EIDH	EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8	96
EC2h	B4SIDL	SID2	SID1	SID0	SRR	EXID	—	EID17	EID16	96
EC1h	B4SIDH	SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3	96
EC0h	B4CON	TXBIF	TXABT	TXLARB	TXERR	TXREQ	—	TXPRI1	TXPRI0	96
EBFh	CANCON_RO6	CANCON_RO6								96
EBEh	CANSTAT_RO6	CANSTAT_RO6								96
EBDh	B3D7	B3D77	B3D76	B3D75	B3D73	B3D73	B3D72	B3D71	B3D70	96
EBCh	B3D6	B3D67	B3D66	B3D65	B3D63	B3D63	B3D62	B3D61	B3D60	96
EBBh	B3D5	B3D57	B3D56	B3D55	B3D53	B3D53	B3D52	B3D51	B3D50	96
EBAh	B3D4	B3D47	B3D46	B3D45	B3D43	B3D43	B3D42	B3D41	B3D40	96
EB9h	B3D3	B3D37	B3D36	B3D35	B3D33	B3D33	B3D32	B3D31	B3D30	96
EB8h	B3D2	B3D27	B3D26	B3D25	B3D23	B3D23	B3D22	B3D21	B3D20	96
EB7h	B3D1	B3D17	B3D16	B3D15	B3D13	B3D13	B3D12	B3D11	B3D10	96
EB6h	B3D0	B3D07	B3D06	B3D05	B3D03	B3D03	B3D02	B3D01	B3D00	96
EB5h	B3DLC	—	TXRTR	—	—	DLC3	DLC2	DLC1	DLC0	96
EB4h	B3EIDL	EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0	96
EB3h	B3EIDH	EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8	96
EB2h	B3SIDL	SID2	SID1	SID0	SRR	EXID	—	EID17	EID16	96
EB1h	B3SIDH	SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3	96
EB0h	B3CON	TXBIF	TXABT	TXLARB	TXERR	TXREQ	—	TXPRI1	TXPRI0	96
EAFh	CANCON_RO7	CANCON_RO7								96
EAEh	CANSTAT_RO7	CANSTAT_RO7								96
EADh	B2D7	B2D77	B2D76	B2D75	B2D72	B2D73	B2D72	B2D71	B2D70	96
EACh	B2D6	B2D67	B2D66	B2D65	B2D62	B2D63	B2D62	B2D61	B2D60	96
EABh	B2D5	B2D57	B2D56	B2D55	B2D52	B2D53	B2D52	B2D51	B2D50	97
EAAh	B2D4	B2D47	B2D46	B2D45	B2D42	B2D43	B2D42	B2D41	B2D40	97
EA9h	B2D3	B2D37	B2D36	B2D35	B2D32	B2D33	B2D32	B2D31	B2D30	97
EA8h	B2D2	B2D27	B2D26	B2D25	B2D22	B2D23	B2D22	B2D21	B2D20	97
EA7h	B2D1	B2D17	B2D16	B2D15	B2D12	B2D13	B2D12	B2D11	B2D10	97
EA6h	B2D0	B2D07	B2D06	B2D05	B2D02	B2D03	B2D02	B2D01	B2D00	97
EA5h	B2DLC	—	TXRTR	—	—	DLC3	DLC2	DLC1	DLC0	97
EA4h	B2EIDL	EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0	97

EXAMPLE 7-3: WRITING TO FLASH PROGRAM MEMORY

```
                MOVLW   SIZE_OF_BLOCK          ; number of bytes in erase block
                MOVWF   COUNTER
                MOVLW   BUFFER_ADDR_HIGH       ; point to buffer
                MOVWF   FSR0H
                MOVLW   BUFFER_ADDR_LOW
                MOVWF   FSR0L
                MOVLW   CODE_ADDR_UPPER        ; Load TBLPTR with the base
                MOVWF   TBLPTRU                ; address of the memory block
                MOVLW   CODE_ADDR_HIGH
                MOVWF   TBLPTRH
                MOVLW   CODE_ADDR_LOW
                MOVWF   TBLPTRL
READ_BLOCK
                TBLRD*+                        ; read into TABLAT, and inc
                MOVF    TABLAT, W              ; get data
                MOVWF   POSTINC0               ; store data
                DECFSZ  COUNTER                ; done?
                BRA     READ_BLOCK             ; repeat
MODIFY_WORD
                MOVLW   DATA_ADDR_HIGH         ; point to buffer
                MOVWF   FSR0H
                MOVLW   DATA_ADDR_LOW
                MOVWF   FSR0L
                MOVLW   NEW_DATA_LOW           ; update buffer word
                MOVWF   POSTINC0
                MOVLW   NEW_DATA_HIGH
                MOVWF   INDF0
ERASE_BLOCK
                MOVLW   CODE_ADDR_UPPER        ; load TBLPTR with the base
                MOVWF   TBLPTRU                ; address of the memory block
                MOVLW   CODE_ADDR_HIGH
                MOVWF   TBLPTRH
                MOVLW   CODE_ADDR_LOW
                MOVWF   TBLPTRL
                BSF     EECON1, EEPGD          ; point to Flash program memory
                BCF     EECON1, CFGS           ; access Flash program memory
                BSF     EECON1, WREN           ; enable write to memory
                BSF     EECON1, FREE           ; enable Row Erase operation
                BCF     INTCON, GIE            ; disable interrupts
                MOVLW   55h
Required        MOVWF   EECON2                 ; write 55h
Sequence        MOVLW   0AAh
                MOVWF   EECON2                 ; write 0AAh
                BSF     EECON1, WR             ; start erase (CPU stall)
                BSF     INTCON, GIE            ; re-enable interrupts
                TBLRD*-                        ; dummy read decrement
                MOVLW   BUFFER_ADDR_HIGH       ; point to buffer
                MOVWF   FSR0H
                MOVLW   BUFFER_ADDR_LOW
                MOVWF   FSR0L
WRITE_BUFFER_BACK
                MOVLW   SIZE_OF_BLOCK          ; number of bytes in holding register
                MOVWF   COUNTER
WRITE_BYTE_TO_HREGS
                MOVFF   POSTINC0, WREG         ; get low byte of buffer data
                MOVWF   TABLAT                 ; present data to table latch
                TBLWT+*                        ; write data, perform a short write
                                               ; to internal TBLWT holding register.
                DECFSZ  COUNTER                ; loop until buffers are full
                BRA     WRITE_BYTE_TO_HREGS
```

TABLE 11-3: PORTB FUNCTIONS

Pin Name	Function	TRIS Setting	I/O	I/O Type	Description
RB0/AN10/C1INA FLT0/INT0	RB0	0	O	DIG	LATB<0> data output.
		1	I	ST	PORTB<0> data input; weak pull-up when RBPU bit is cleared.
	AN10	1	I	ANA	A/D Input Channel 10 and Comparator C1+ input. Default input configuration on POR.
	C1INA(1)	1	I	ANA	Comparator 1 Input A.
	FLT0	x	I	ST	Enhanced PWM Fault input for ECCPx.
	INT0	1	I	ST	External Interrupt 0 input.
RB1/AN8/C1INB/ P1B/CTDIN/INT1	RB1	0	O	DIG	LATB<1> data output.
		1	I	ST	PORTB<1> data input; weak pull-up when RBPU bit is cleared.
	AN8	1	I	ANA	A/D Input Channel 8 and Comparator C2+ input. Default input configuration on POR; not affected by analog output.
	C1INB(1)	1	I	ANA	Comparator 1 Input B.
	P1B(1)	0	O	DIG	ECCP1 PWM Output B. May be configured for tri-state during Enhanced PWM shutdown events.
	CTDIN	1	I	ST	CTMU pulse delay input.
	INT1	1	I	ST	External Interrupt 1 input.
RB2/CANTX/C1OUT/ P1C/CTED1/INT2	RB2	0	O	DIG	LATB<2> data output.
		1	I	ST	PORTB<2> data input; weak pull-up when RBPU bit is cleared.
	CANTX(2)	0	O	DIG	CAN bus TX.
	C1OUT(1)	0	O	DIG	Comparator 1 output; takes priority over port data.
	P1C(1)	0	O	DIG	ECCP1 PWM Output C. May be configured for tri-state during Enhanced PWM.
	CTED1	x	I	ST	CTMU Edge 1 input.
	INT2	1	I	ST	External Interrupt 2.
RB3/CANRX/ C2OUT/P1D/ CTED2/INT3	RB3	0	O	DIG	LATB<3> data output.
		1	I	ST	PORTB<3> data input; weak pull-up when RBPU bit is cleared.
	CANRX(2)	1	I	ST	CAN bus RX.
	C2OUT(1)	x	I	ST	CTMU Edge 2 input.
	P1D(1)	0	O	DIG	ECCP1 PWM Output D. May be configured for tri-state during Enhanced PWM.
	CTED2	x	I	ST	CTMU Edge 2 input.
	INT3	1	I	ST	External Interrupt 3 input.

Legend: O = Output; I = Input; ANA = Analog Signal; DIG = CMOS Output; ST = Schmitt Trigger Buffer Input; x = Don't care (TRIS bit does not affect port direction or is overridden for this option)

Note 1: This pin assignment is only available for 28-pin devices (PIC18F2XK80).

2: This is the default pin assignment for CANRX and CANTX when the CANMX Configuration bit is set.

3: This is the default pin assignment for T0CKI when the T0CKMX Configuration bit is set.

4: This is the default pin assignment for T3CKI for 28, 40 and 44-pin devices. This is the alternate pin assignment for T3CKI for 64-pin devices when T3CKMX is cleared.

TABLE 11-7: PORTD FUNCTIONS

Pin Name	Function	TRIS Setting	I/O	I/O Type	Description
RD0/C1INA/ PSP0	RD0	0	O	DIG	LATD<0> data output.
		1	I	ST	PORTD<0> data input.
	C1INA	1	I	ANA	Comparator 1 Input A.
	PSP0	x	I/O	ST	Parallel Slave Port data.
RD1/C1INB/ PSP1	RD1(1)	0	O	DIG	LATD<1> data output.
		1	I	ST	PORTD<1> data input.
	C1INB(1)	1	I	ANA	Comparator 1 Input B.
	PSP1(1)	x	I/O	ST	Parallel Slave Port data.
RD2/C2INA/ PSP2	RD2	0	O	DIG	LATD<2> data output.
		1	I	ST	PORTD<2> data input.
	C2INA	1	I	ANA	Comparator 2 Input A.
	PSP2	x	I/O	ST	Parallel Slave Port data.
RD3/C2INB/ CTMUI/PSP3	RD3	0	O	DIG	LATD<3> data output.
		1	I	ST	PORTD<3> data input.
	C2INB	1	I	ANA	Comparator 2 Input B.
	CTMUI	x	I	—	CTMU pulse generator charger for the C2INB comparator input.
	PSP3	x	I/O	ST	Parallel Slave Port data.
RD4/ECCP1/ P1A/PSP4	RD4	0	O	DIG	LATD<4> data output.
		1	I	ST	PORTD<4> data input.
	ECCP1	0	O	DIG	ECCP1 compare output and ECCP1 PWM output; takes priority over port data.
		1	I	ST	ECCP1 capture input.
	P1A	0	O	DIG	ECCP1 Enhanced PWM output, Channel A. May be configured for tri-state during Enhanced PWM shutdown events; takes priority over port data.
	PSP4	x	I/O	ST	Parallel Slave Port data.
RD5/P1B/PSP5	RD5	0	O	DIG	LATD<5> data output.
		1	I	ST	PORTD<5> data input.
	P1B	0	O	DIG	ECCP1 Enhanced PWM output, Channel B. May be configured for tri-state during Enhanced PWM shutdown events; takes priority over port data.
	PSP5	x	I/O	ST	Parallel Slave Port data.
RD6/TX2/CK2 P1C/PSP6	RD6	0	O	DIG	LATD<6> data output.
		1	I	ST	PORTD<6> data input.
	TX2(1)	0	O	DIG	Asynchronous serial data output (EUSARTx module); takes priority over port data.
	CK2(1)	0	O	DIG	Synchronous serial clock output (EUSARTx module); user must configure as an input.
		1	I	ST	Synchronous serial clock input (EUSARTx module); user must configure as an input.
	P1C	0	O	DIG	ECCP1 Enhanced PWM output, Channel C. May be configured for tri-state during Enhanced PWM.
	PSP6	x	I/O	ST	Parallel Slave Port data.

Legend: O = Output; I = Input; ANA = Analog Signal; DIG = CMOS Output; ST = Schmitt Trigger Buffer Input; x = Don't care (TRIS bit does not affect port direction or is overridden for this option)

Note 1: This is the pin assignment for 40 and 44-pin devices (PIC18F4XK80).

TABLE 11-9: PORTE FUNCTIONS (CONTINUED)

Pin Name	Function	TRIS Setting	I/O	I/O Type	Description
RE5/CANTX	RE5(1)	0	O	DIG	LATE<5> data output.
		1	I	ST	PORTE<5> data input.
	CANTX(1,2)	0	O	DIG	CAN bus TX.
RE6/RX2/DT2	RE6(1)	0	O	DIG	LATE<6> data output.
		1	I	ST	PORTE<6> data input.
	RX2(1)	1	I	ST	Asynchronous serial receive data input (EUSARTx module).
	DT2(1)	1	O	DIG	Synchronous serial data output (EUSARTx module); takes priority over port data.
		1	I	ST	Synchronous serial data input (EUSARTx module); user must configure as an input.
RE7/TX2/CK2	RE7(1)	0	O	DIG	LATE<7> data output.
		1	I	ST	PORTE<7> data input.
	TX2(1)	0	O	DIG	Asynchronous serial data output (EUSARTx module); takes priority over port data.
	CK2(1)	0	O	DIG	Synchronous serial clock output (EUSARTx module); user must configure as an input.
		1	I	ST	Synchronous serial clock input (EUSARTx module); user must configure as an input.

Legend: O = Output, I = Input, ANA = Analog Signal, DIG = CMOS Output, ST = Schmitt Trigger Buffer Input, x = Don't care (TRIS bit does not affect port direction or is overridden for this option)

Note 1: These bits are unavailable for 40 and 44-pin devices (PIC18F4XK0).

2: This is the alternate pin assignment for CANRX and CANTX on 64-pin devices (PIC18F6XK80) when the CANMX Configuration bit is cleared.

TABLE 11-10: SUMMARY OF REGISTERS ASSOCIATED WITH PORTE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PORTE	RE7(1)	RE6(1)	RE5(1)	RE4(1)	RE3	RE2	RE1	RE0
LATE	LATE7	LATE6	LATE5	LATE4	—	LATE2	LATE1	LATE0
TRISE	TRISE7	TRISE6	TRISE5	TRISE4	—	TRISE2	TRISE1	TRISE0
PADCFG1	RDPU	REPU	RFPU(1)	RGPU(1)	—	—	—	CTMUDS
ANCON0	ANSEL7	ANSEL6	ANSEL5	ANSEL4	ANSEL3	ANSEL2	ANSEL1	ANSEL0

Legend: Shaded cells are not used by PORTE.

Note 1: These bits are unimplemented on 44-pin devices, read as '0'.

14.8.4 TIMER1 GATE SINGLE PULSE MODE

When Timer1 Gate Single Pulse mode is enabled, it is possible to capture a single pulse gate event. Timer1 Gate Single Pulse mode is enabled by setting the T1GSPM bit (T1GCON<4>) and the T1GGO/$\overline{\text{T1DONE}}$ bit (T1GCON<3>). The Timer1 will be fully enabled on the next incrementing edge.

On the next trailing edge of the pulse, the T1GGO/$\overline{\text{T1DONE}}$ bit will automatically be cleared. No other gate events will be allowed to increment Timer1 until the T1GGO/$\overline{\text{T1DONE}}$ bit is once again set in software.

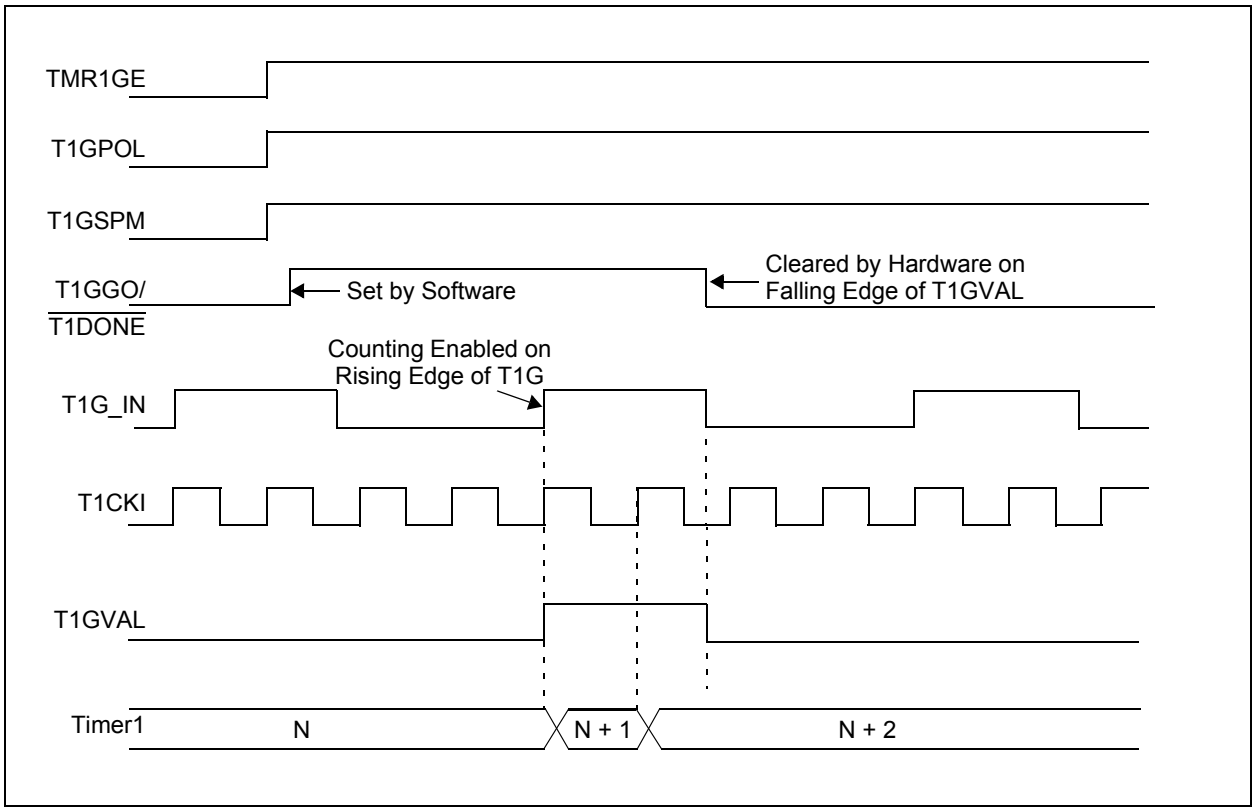
Clearing the T1GSPM bit of the T1GCON register will also clear the T1GGO/$\overline{\text{T1DONE}}$ bit. (For timing details, see Figure 14-6.)

Simultaneously enabling the Toggle and Single Pulse modes will permit both sections to work together. This allows the cycle times on the Timer1 gate source to be measured. (For timing details, see Figure 14-7.)

14.8.5 TIMER1 GATE VALUE STATUS

When the Timer1 gate value status is utilized, it is possible to read the most current level of the gate control value. The value is stored in the T1GVAL bit (T1GCON<2>). This bit is valid even when the Timer1 gate is not enabled (TMR1GE bit is cleared).

FIGURE 14-6: TIMER1 GATE SINGLE PULSE MODE



18.9 Measuring Temperature with the CTMU Module

The CTMU, along with an internal diode, can be used to measure the temperature. The A/D can be connected to the internal diode and the CTMU module can source the current to the diode. The A/D reading will reflect the temperature. With the increase, the A/D readings will go low. This can be used for low-cost temperature measurement applications.

EXAMPLE 18-6: ROUTINE FOR TEMPERATURE MEASUREMENT USING INTERNAL DIODE

```
// Initialize CTMU
CTMUICON = 0x03;
CTMUCONHbits.CTMUEN = 1;
CTMUCONLbits.EDG1STAT = 1;

// Initialize ADC
ADCON0 = 0xE5;                          // Enable ADC and connect to Internal diode
ADCON1 = 0x00;
ADCON2 = 0xBE;                          //Right Justified

ADCON0bits.GO = 1;                      // Start conversion
while(ADCON0bits.G0);
Temp = ADRES;                           // Read ADC results (inversely proportional to temperature)
```

Note: The temperature diode is not calibrated or standardized; the user must calibrate the diode to their application.

21.3.6 MASTER MODE

The master can initiate the data transfer at any time because it controls the SCK. The master determines when the slave (Processor 1, Figure 21-2) is to broadcast data by the software protocol.

In Master mode, the data is transmitted/received as soon as the SSPBUF register is written to. If the SPI is only going to receive, the SDO output could be disabled (programmed as an input). The SSPSR register will continue to shift in the signal present on the SDI pin at the programmed clock rate. As each byte is received, it will be loaded into the SSPBUF register as if a normal received byte (interrupts and status bits appropriately set). This could be useful in receiver applications as a “Line Activity Monitor” mode.

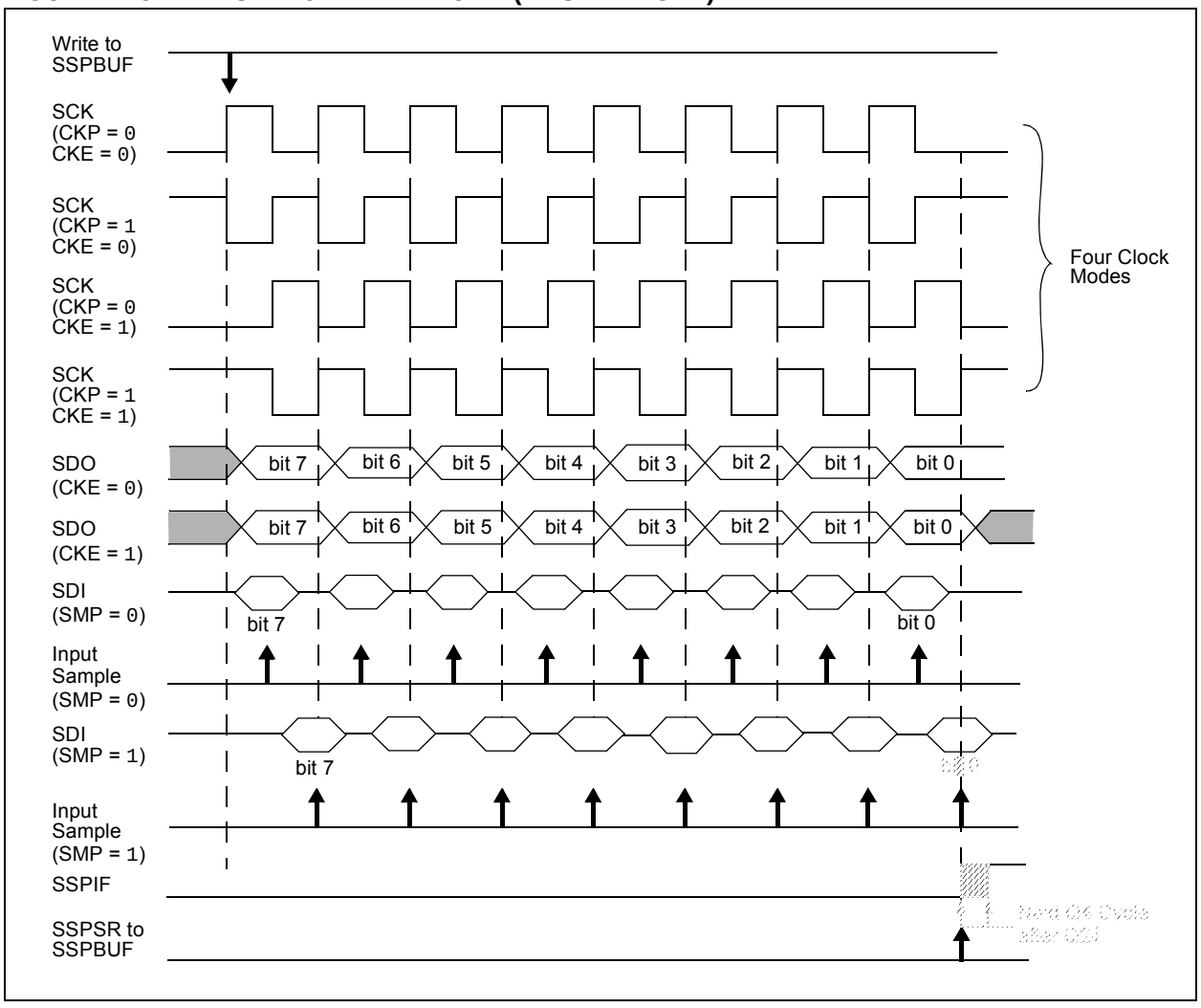
The clock polarity is selected by appropriately programming the CKP bit (SSPCON1<4>). This then, would give waveforms for SPI communication as shown in Figure 21-3, Figure 21-5 and Figure 21-6, where the MSB is transmitted first. In Master mode, the SPI clock rate (bit rate) is user-programmable to be one of the following:

- FOSC/4 (or TCY)
- FOSC/16 (or 4 • TCY)
- FOSC/64 (or 16 • TCY)
- Timer2 output/2

This allows a maximum data rate (at 64 MHz) of 16 Mbps.

Figure 21-3 shows the waveforms for Master mode. When the CKE bit is set, the SDO data is valid before there is a clock edge on SCK. The change of the input sample is shown based on the state of the SMP bit. The time when the SSPBUF is loaded with the received data is shown.

FIGURE 21-3: SPI MODE WAVEFORM (MASTER MODE)



23.3 A/D Acquisition Requirements

For the A/D Converter to meet its specified accuracy, the Charge Holding (C_{HOLD}) capacitor must be allowed to fully charge to the input channel voltage level. The analog input model is shown in Figure 23-5. The source impedance (R_S) and the internal sampling switch (R_{SS}) impedance directly affect the time required to charge the capacitor C_{HOLD}. The sampling switch (R_{SS}) impedance varies over the device voltage (V_{DD}).

The source impedance affects the offset voltage at the analog input (due to pin leakage current). **The maximum recommended impedance for analog sources is 2.5 kΩ.** After the analog input channel is selected or changed, the channel must be sampled for at least the minimum acquisition time before starting a conversion.

Note:	When the conversion is started, the holding capacitor is disconnected from the input pin.

To calculate the minimum acquisition time, Equation 23-1 can be used. This equation assumes that 1/2 LSb error is used (1,024 steps for the A/D). The 1/2 LSb error is the maximum error allowed for the A/D to meet its specified resolution.

Equation 23-3 shows the calculation of the minimum required acquisition time, T_{ACQ}. This calculation is based on the following application system assumptions:

C_{HOLD}	=	25 pF
Rs	=	2.5 kΩ
Conversion Error	≤	1/2 LSb
V_{DD}	=	3V → Rss = 2 kΩ
Temperature	=	85°C (system max.)

EQUATION 23-1: ACQUISITION TIME

$$T_{ACQ} = \text{Amplifier Settling Time} + \text{Holding Capacitor Charging Time} + \text{Temperature Coefficient}$$
$$= T_{AMP} + T_C + T_{COFF}$$

EQUATION 23-2: A/D MINIMUM CHARGING TIME

$$V_{HOLD} = (V_{REF} - (V_{REF}/2048)) \bullet (1 - e^{(-T_C/C_{HOLD}(R_{IC} + R_{SS} + R_S))})$$
or
$$T_C = -(C_{HOLD})(R_{IC} + R_{SS} + R_S)\ln(1/2048)$$

EQUATION 23-3: CALCULATING THE MINIMUM REQUIRED ACQUISITION TIME

$T_{ACQ} = T_{AMP} + T_C + T_{COFF}$

$T_{AMP} = 0.2\ \mu s$

$T_{COFF} = (\text{Temp} - 25°C)(0.02\ \mu s/°C)$
$(85°C - 25°C)(0.02\ \mu s/°C)$
$1.2\ \mu s$

Temperature coefficient is only required for temperatures > 25°C. Below 25°C, T_{COFF} = 0 ms.

$T_C = -(C_{HOLD})(R_{IC} + R_{SS} + R_S)\ln(1/2048)\ \mu s$
$-(25\ pF)(1\ k\Omega + 2\ k\Omega + 2.5\ k\Omega)\ln(0.0004883)\ \mu s$
$1.05\ \mu s$

$T_{ACQ} = 0.2\ \mu s + 1.05\ \mu s + 1.2\ \mu s$
$2.45\ \mu s$

REGISTER 24-2: CMSTAT: COMPARATOR STATUS REGISTER

R-x	R-x	U-0	U-0	U-0	U-0	U-0	U-0
CMP2OUT	CMP1OUT	—	—	—	—	—	—
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 7-6 **CMP2OUT:CMP1OUT:** Comparator x Status bits

If CPOL (CMxCON<5>)= 0 (non-inverted polarity):

1 = Comparator x's VIN+ > VIN-

0 = Comparator x's VIN+ < VIN-

If CPOL = 1 (inverted polarity):

1 = Comparator x's VIN+ < VIN-

0 = Comparator x's VIN+ > VIN-

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

24.5 Comparator Control and Configuration

Each comparator has up to eight possible combinations of inputs: up to four external analog inputs and one of two internal voltage references.

All of the comparators allow a selection of the signal from pin, CxINA, or the voltage from the comparator reference (CV_{REF}) on the non-inverting channel. This is compared to either C1INB, CxINC, C2INB or the microcontroller's fixed internal reference voltage (V_{BG}, 1.024V nominal) on the inverting channel. The comparator inputs and outputs are tied to fixed I/O pins, defined in Table 24-1. The available comparator configurations and their corresponding bit settings are shown in Figure 24-4.

TABLE 24-1: COMPARATOR INPUTS AND OUTPUTS

Comparator	Input or Output	I/O Pin(†)
1	C1INA (V_{IN}+)	RB0/RD0
	C1INB (V_{IN}-)	RB1/RD1
	C1INC (V_{IN}-)	RA1
	C2INB(V_{IN}-)	RA5/RD3
	C1OUT	RB2/RE1
2	C2INA(V_{IN}+)	RB4/RD2
	C2INB(V_{IN}-)	RA5/RD3
	C2INC(V_{IN}-)	RA2
	C2OUT	RB3/RE2

† The I/O pin is dependent on package type.

24.5.1 COMPARATOR ENABLE AND INPUT SELECTION

Setting the CON bit of the CMxCON register (CMxCON<7>) enables the comparator for operation. Clearing the CON bit disables the comparator, resulting in minimum current consumption.

The CCH<1:0> bits in the CMxCON register (CMxCON<1:0>) direct either one of three analog input pins, or the Internal Reference Voltage (V_{BG}), to the comparator, V_{IN}-. Depending on the comparator operating mode, either an external or internal voltage reference may be used.

The analog signal present at V_{IN}- is compared to the signal at V_{IN}+ and the digital output of the comparator is adjusted accordingly.

The external reference is used when CREF = 0 (CMxCON<2>) and V_{IN}+ is connected to the CxINA pin. When external voltage references are used, the comparator module can be configured to have the reference sources externally. The reference signal must be between V_{SS} and V_{DD} and can be applied to either pin of the comparator.

The comparator module also allows the selection of an internally generated voltage reference (CV_{REF}) from the comparator voltage reference module. This module is described in more detail in **Section 25.0 "Comparator Voltage Reference Module"**. The reference from the comparator voltage reference module is only available when CREF = 1. In this mode, the internal voltage reference is applied to the comparator's V_{IN}+ pin.

Note:	The comparator input pin selected by CCH<1:0> must be configured as an input by setting both the corresponding TRIS bit and the corresponding ANSELx bit in the ANCONx register.

24.5.2 COMPARATOR ENABLE AND OUTPUT SELECTION

The comparator outputs are read through the CMSTAT register. The CMSTAT<6> bit reads the Comparator 1 output, CMSTAT<7> reads the Comparator 2 output. These bits are read-only.

The comparator outputs may also be directly output to the RE2 and RE1 pins by setting the COE bit (CMxCON<6>). When enabled, multiplexers in the output path of the pins switch to the output of the comparator. While in this mode, the TRISE<2:1> bits still function as the digital output enable bits for the RE2, and RE1 pins.

By default, the comparator's output is at logic high whenever the voltage on V_{IN}+ is greater than on V_{IN}-. The polarity of the comparator outputs can be inverted using the CPOL bit (CMxCON<5>).

The uncertainty of each of the comparators is related to the input offset voltage and the response time given in the specifications, as discussed in **Section 24.2 "Comparator Operation"**.

BTG **Bit Toggle f**

Syntax: BTG f, b {,a}

Operands: $0 \le f \le 255$
$0 \le b < 7$
$a \in [0,1]$

Operation: $(\overline{f<b>}) \rightarrow f<b>$

Status Affected: None

Encoding:

0111	bbba	ffff	ffff

Description: Bit ‘b’ in data memory location ‘f’ is inverted.

If ‘a’ is ‘0’, the Access Bank is selected. If ‘a’ is ‘1’, the BSR is used to select the GPR bank.

If ‘a’ is ‘0’ and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever $f \le 95$ (5Fh). See **Section 29.2.3 “Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode”** for details.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register ‘f’	Process Data	Write register ‘f’

Example: `BTG PORTC, 4, 0`

Before Instruction:
PORTC = 0111 0101 [75h]
After Instruction:
PORTC = 0110 0101 [65h]

BOV **Branch if Overflow**

Syntax: BOV n

Operands: $-128 \le n \le 127$

Operation: if Overflow bit is ‘1’,
$(PC) + 2 + 2n \rightarrow PC$

Status Affected: None

Encoding:

1110	0100	nnnn	nnnn

Description: If the Overflow bit is ‘1’, then the program will branch.

The 2’s complement number, ‘2n’, is added to the PC. Since the PC will have incremented to fetch the next instruction, the new address will be PC + 2 + 2n. This instruction is then a two-cycle instruction.

Words: 1

Cycles: 1(2)

Q Cycle Activity:
If Jump:

Q1	Q2	Q3	Q4
Decode	Read literal ‘n’	Process Data	Write to PC
No operation	No operation	No operation	No operation

If No Jump:

Q1	Q2	Q3	Q4
Decode	Read literal ‘n’	Process Data	No operation

Example:
```
HERE    BOV   Jump
```
Before Instruction
PC = address (HERE)
After Instruction
If Overflow = 1;
PC = address (Jump)
If Overflow = 0;
PC = address (HERE + 2)

SLEEP **Enter Sleep Mode**

Syntax: SLEEP

Operands: None

Operation: 00h → WDT,
0 → WDT postscaler,
1 → $\overline{TO}$,
0 → $\overline{PD}$

Status Affected: $\overline{TO}$, $\overline{PD}$

Encoding:

0000	0000	0000	0011

Description: The Power-Down status bit ($\overline{PD}$) is cleared. The Time-out status bit ($\overline{TO}$) is set. The Watchdog Timer and its postscaler are cleared.

The processor is put into Sleep mode with the oscillator stopped.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	No operation	Process Data	Go to Sleep

Example: SLEEP

Before Instruction
$\overline{TO}$ = ?
$\overline{PD}$ = ?
After Instruction
$\overline{TO}$ = 1 †
$\overline{PD}$ = 0

† If WDT causes wake-up, this bit is cleared.

SUBFWB **Subtract f from W with Borrow**

Syntax: SUBFWB f {,d {,a}}

Operands: 0 ≤ f ≤ 255
d ∈ [0,1]
a ∈ [0,1]

Operation: (W) – (f) – ($\overline{C}$) → dest

Status Affected: N, OV, C, DC, Z

Encoding:

0101	01da	ffff	ffff

Description: Subtract register 'f' and Carry flag (borrow) from W (2's complement method). If 'd' is '0', the result is stored in W. If 'd' is '1', the result is stored in register 'f' (default).

If 'a' is '0', the Access Bank is selected. If 'a' is '1', the BSR is used to select the GPR bank.

If 'a' is '0' and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever f ≤ 95 (5Fh). See **Section 29.2.3 "Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode"** for details.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	Write to destination

Example 1: SUBFWB REG, 1, 0

Before Instruction
REG = 3
W = 2
C = 1
After Instruction
REG = FF
W = 2
C = 0
Z = 0
N = 1 ; result is negative

Example 2: SUBFWB REG, 0, 0

Before Instruction
REG = 2
W = 5
C = 1
After Instruction
REG = 2
W = 3
C = 1
Z = 0
N = 0 ; result is positive

Example 3: SUBFWB REG, 1, 0

Before Instruction
REG = 1
W = 2
C = 0
After Instruction
REG = 0
W = 2
C = 1
Z = 1 ; result is zero
N = 0

31.2 DC Characteristics: Power-Down and Supply Current PIC18F66K80 Family (Industrial/Extended) (Continued)

PIC18F66K80 Family (Industrial/Extended)		Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended					
Param No.	**Device**	**Typ**	**Max**	**Units**	**Conditions**		
	Supply Current (IDD) Cont.(2,3)						
	PIC18**LF**XXK80	880	1600	nA	-40°C	VDD = 1.8V(4) Regulator Disabled	FOSC = 31 kHz (**RC_IDLE** mode, LF-INTOSC)
		880	1600	nA	+25°C		
		880	1600	nA	+60°C		
		1	2	µA	+85°C		
		5	10	µA	+125°C		
	PIC18**LF**XXK80	1.6	5	µA	-40°C	VDD = 3.3V(4) Regulator Disabled	
		1.6	5	µA	+25°C		
		1.6	5	µA	+60°C		
		2	6	µA	+85°C		
		7	12	µA	+125°C		
	PIC18**F**XXK80	41	130	µA	-40°C	VDD = 3.3V(5) Regulator Enabled	
		59	130	µA	+25°C		
		64	130	µA	+60°C		
		70	150	µA	+85°C		
		80	175	µA	+125°C		
	PIC18**F**XXK80	53	160	µA	-40°C	VDD = 5V(5) Regulator Enabled	
		62	160	µA	+25°C		
		70	160	µA	+60°C		
		85	170	µA	+85°C		
		100	180	µA	+125°C		

Legend: Shading of rows is to assist in readability of the table.

Note 1: The power-down current in Sleep mode does not depend on the oscillator type. Power-down current is measured with the part in Sleep mode, with all I/O pins in a high-impedance state and tied to VDD or VSS, and all features that add delta current are disabled (such as WDT, SOSC oscillator, BOR, etc.).

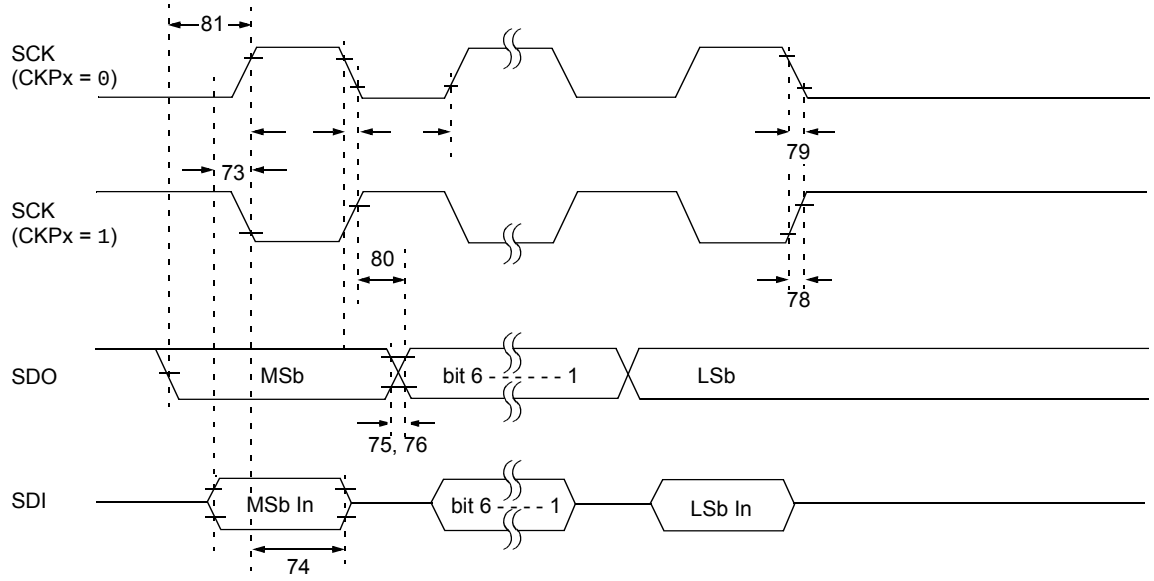
2: The supply current is mainly a function of operating voltage, frequency and mode. Other factors, such as I/O pin loading and switching rate, oscillator type and circuit, internal code execution pattern and temperature, also have an impact on the current consumption.
The test conditions for all IDD measurements in active operation mode are:
OSC1 = external square wave, from rail-to-rail; all I/O pins tri-stated, pulled to VDD;
$\overline{MCLR}$ = VDD; WDT enabled/disabled as specified.

3: Standard, low-cost 32 kHz crystals have an operating temperature range of -10°C to +70°C. Extended temperature crystals are available at a much higher cost.

4: For LF devices, $\overline{RETEN}$ (CONFIG1L<0>) = 1.

5: For F devices, SRETEN (WDTCON<4>) = 1 and $\overline{RETEN}$ (CONFIG1L<0>) = 0.

FIGURE 31-12: EXAMPLE SPI MASTER MODE TIMING (CKE = 1)



Note: Refer to Figure 31-3 for load conditions.

TABLE 31-16: EXAMPLE SPI MODE REQUIREMENTS (MASTER MODE, CKE = 1)

Param. No.	Symbol	Characteristic	Min	Max	Units	Conditions
73	TDIV2SCH, TDIV2SCL	Setup Time of SDI Data Input to SCK Edge	20	—	ns	
73A	TB2B	Last Clock Edge of Byte 1 to the 1st Clock Edge of Byte 2	1.5 TCY + 40	—	ns	
74	TSCH2DIL, TSCL2DIL	Hold Time of SDI Data Input to SCK Edge	40	—	ns	
75	TDOR	SDO Data Output Rise Time	—	25	ns	
76	TDOF	SDO Data Output Fall Time	—	25	ns	
78	TSCR	SCK Output Rise Time (Master mode)	—	25	ns	
79	TSCF	SCK Output Fall Time (Master mode)	—	25	ns	
80	TSCH2DOV, TSCL2DOV	SDO Data Output Valid after SCK Edge	—	50	ns	
81	TDOV2SCH, TDOV2SCL	SDO Data Output Setup to SCK Edge	TCY	—	ns	