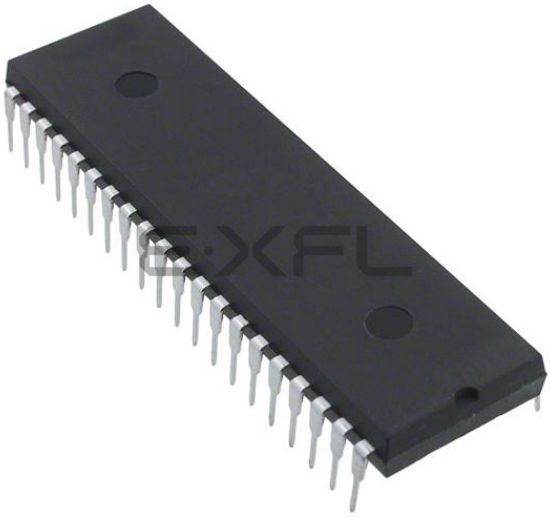


Welcome to E-XFL.COM

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	40MHz
Connectivity	CANbus, I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, HLVD, POR, PWM, WDT
Number of I/O	36
Program Memory Size	80KB (40K x 16)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	3.25K x 8
Voltage - Supply (Vcc/Vdd)	4.2V ~ 5.5V
Data Converters	A/D 11x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Through Hole
Package / Case	40-DIP (0.600", 15.24mm)
Supplier Device Package	40-PDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f4682-e-p

FIGURE 3-1: TRANSITION TIMING FOR ENTRY TO SEC_RUN MODE

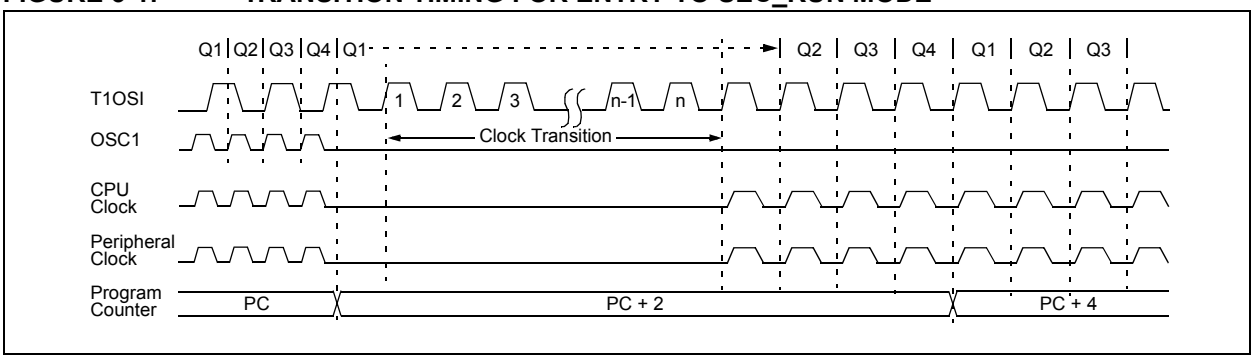
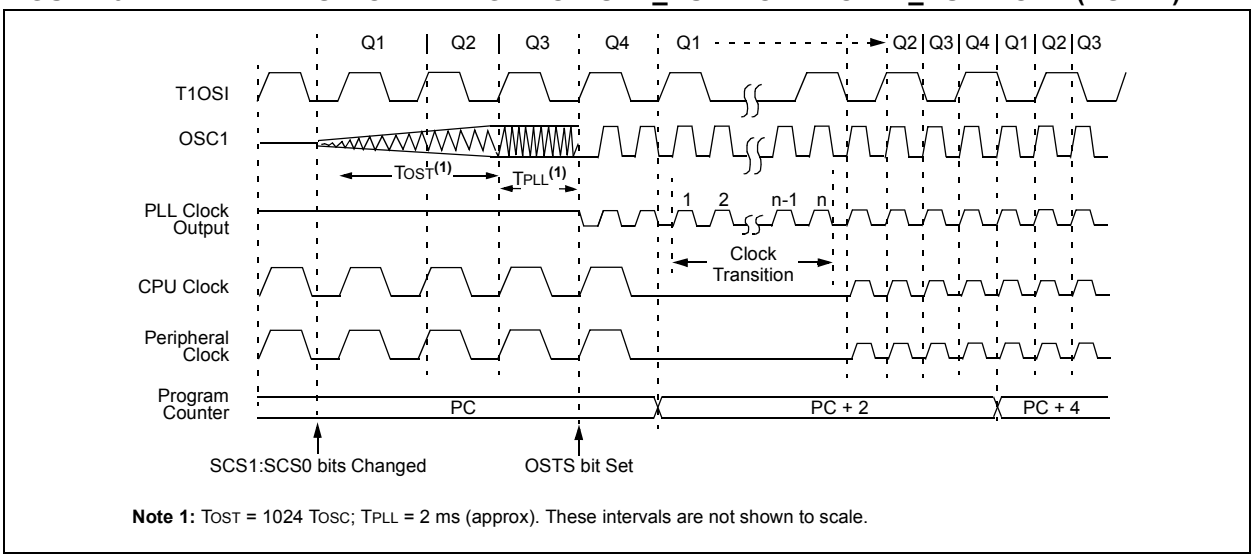


FIGURE 3-2: TRANSITION TIMING FROM SEC_RUN MODE TO PRI_RUN MODE (HSPLL)



Note 1: TOST = 1024 TOSC; TPLL = 2 ms (approx). These intervals are not shown to scale.

3.2.3 RC_RUN MODE

In RC_RUN mode, the CPU and peripherals are clocked from the internal oscillator block using the INTOSC multiplexer; the primary clock is shut down. When using the INTRC source, this mode provides the best power conservation of all the Run modes, while still executing code. It works well for user applications which are not highly timing sensitive or do not require high-speed clocks at all times.

If the primary clock source is the internal oscillator block (either INTRC or INTOSC), there are no distinguishable differences between PRI_RUN and RC_RUN modes during execution. However, a clock switch delay will occur during entry to and exit from RC_RUN mode. Therefore, if the primary clock source is the internal oscillator block, the use of RC_RUN mode is not recommended.

This mode is entered by setting SCS1 to '1'. Although it is ignored, it is recommended that SCS0 also be cleared; this is to maintain software compatibility with future devices. When the clock source is switched to the INTOSC multiplexer (see Figure 3-3), the primary oscillator is shut down and the OSTS bit is cleared. The IRCF bits may be modified at any time to immediately change the clock speed.

Note:	Caution should be used when modifying a single IRCF bit. If VDD is less than 3V, it is possible to select a higher clock speed than is supported by the low VDD. Improper device operation may result if the VDD/FOSC specifications are violated.

If the IRCF bits and the INTSRC bit are all clear, the INTOSC output is not enabled and the IOFS bit will remain clear; there will be no indication of the current clock source. The INTRC source is providing the device clocks.

If the IRCF bits are changed from all clear (thus, enabling the INTOSC output) or if INTSRC is set, the IOFS bit becomes set after the INTOSC output becomes stable. Clocks to the device continue while the INTOSC source stabilizes after an interval of TIOBST.

4.6 Reset State of Registers

Most registers are unaffected by a Reset. Their status is unknown on POR and unchanged by all other Resets. The other registers are forced to a "Reset state" depending on the type of Reset that occurred.

Most registers are not affected by a WDT wake-up, since this is viewed as the resumption of normal operation. Status bits from the RCON register, $\overline{RI}$, $\overline{TO}$, $\overline{PD}$, $\overline{POR}$ and $\overline{BOR}$, are set or cleared differently in different Reset situations, as indicated in Table 4-3. These bits are used in software to determine the nature of the Reset.

Table 4-4 describes the Reset states for all of the Special Function Registers. These are categorized by Power-on and Brown-out Resets, Master Clear and WDT Resets and WDT wake-ups.

TABLE 4-3: STATUS BITS, THEIR SIGNIFICANCE AND THE INITIALIZATION CONDITION FOR RCON REGISTER

Condition	Program Counter	RCON Register						STKPTR Register	
		SBOREN	$\overline{RI}$	$\overline{TO}$	$\overline{PD}$	$\overline{POR}$	$\overline{BOR}$	STKFUL	STKUNF
Power-on Reset	0000h	1	1	1	1	0	0	0	0
RESET instruction	0000h	u(2)	0	u	u	u	u	u	u
Brown-out Reset	0000h	u(2)	1	1	1	u	0	u	u
$\overline{MCLR}$ during power-managed Run modes	0000h	u(2)	u	1	u	u	u	u	u
$\overline{MCLR}$ during power-managed Idle modes and Sleep mode	0000h	u(2)	u	1	0	u	u	u	u
WDT time-out during full power or power-managed Run modes	0000h	u(2)	u	0	u	u	u	u	u
$\overline{MCLR}$ during full power execution	0000h	u(2)	u	u	u	u	u	u	u
Stack Full Reset (STVREN = 1)	0000h	u(2)	u	u	u	u	u	1	u
Stack Underflow Reset (STVREN = 1)	0000h	u(2)	u	u	u	u	u	u	1
Stack Underflow Error (not an actual Reset, STVREN = 0)	0000h	u(2)	u	u	u	u	u	u	1
WDT time-out during power-managed Idle or Sleep modes	PC + 2	u(2)	u	0	0	u	u	u	u
Interrupt exit from power-managed modes	PC + 2(1)	u(2)	u	u	0	u	u	u	u

Legend: u = unchanged

Note 1: When the wake-up is due to an interrupt and the GIEH or GIEL bit is set, the PC is loaded with the interrupt vector (008h or 0018h).

2: Reset state is '1' for POR and unchanged for all other Resets when software BOR is enabled (BOREN1:BOREN0 Configuration bits = 01 and SBOREN = 1); otherwise, the Reset state is '0'.

TABLE 4-4: INITIALIZATION CONDITIONS FOR ALL REGISTERS

Register	Applicable Devices				Power-on Reset, Brown-out Reset	$\overline{\text{MCLR}}$ Resets, WDT Reset, RESET Instruction, Stack Resets	Wake-up via WDT or Interrupt
TOSU	2682	2685	4682	4685	---0 0000	---0 0000	---0 uuuu(3)
TOSH	2682	2685	4682	4685	0000 0000	0000 0000	uuuu uuuu(3)
TOSL	2682	2685	4682	4685	0000 0000	0000 0000	uuuu uuuu(3)
STKPTR	2682	2685	4682	4685	00-0 0000	uu-0 0000	uu-u uuuu(3)
PCLATU	2682	2685	4682	4685	---0 0000	---0 0000	---u uuuu
PCLATH	2682	2685	4682	4685	0000 0000	0000 0000	uuuu uuuu
PCL	2682	2685	4682	4685	0000 0000	0000 0000	PC + 2(2)
TBLPTRU	2682	2685	4682	4685	--00 0000	--00 0000	--uu uuuu
TBLPTRH	2682	2685	4682	4685	0000 0000	0000 0000	uuuu uuuu
TBLPTRL	2682	2685	4682	4685	0000 0000	0000 0000	uuuu uuuu
TABLAT	2682	2685	4682	4685	0000 0000	0000 0000	uuuu uuuu
PRODH	2682	2685	4682	4685	xxxx xxxx	uuuu uuuu	uuuu uuuu
PRODL	2682	2685	4682	4685	xxxx xxxx	uuuu uuuu	uuuu uuuu
INTCON	2682	2685	4682	4685	0000 000x	0000 000u	uuuu uuuu(1)
INTCON2	2682	2685	4682	4685	1111 -1-1	1111 -1-1	uuuu -u-u(1)
INTCON3	2682	2685	4682	4685	11-0 0-00	11-0 0-00	uu-u u-uu(1)
INDF0	2682	2685	4682	4685	N/A	N/A	N/A
POSTINC0	2682	2685	4682	4685	N/A	N/A	N/A
POSTDEC0	2682	2685	4682	4685	N/A	N/A	N/A
PREINC0	2682	2685	4682	4685	N/A	N/A	N/A
PLUSW0	2682	2685	4682	4685	N/A	N/A	N/A
FSR0H	2682	2685	4682	4685	---- 0000	---- 0000	---- uuuu
FSR0L	2682	2685	4682	4685	xxxx xxxx	uuuu uuuu	uuuu uuuu
WREG	2682	2685	4682	4685	xxxx xxxx	uuuu uuuu	uuuu uuuu
INDF1	2682	2685	4682	4685	N/A	N/A	N/A
POSTINC1	2682	2685	4682	4685	N/A	N/A	N/A
POSTDEC1	2682	2685	4682	4685	N/A	N/A	N/A
PREINC1	2682	2685	4682	4685	N/A	N/A	N/A
PLUSW1	2682	2685	4682	4685	N/A	N/A	N/A
FSR1H	2682	2685	4682	4685	---- 0000	---- 0000	---- uuuu
FSR1L	2682	2685	4682	4685	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 4-3 for Reset value for specific condition.

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 '0'.

6: This register reads all '0's until the ECAN™ technology is set up in Mode 1 or Mode 2.

TABLE 5-1: SPECIAL FUNCTION REGISTER MAP FOR PIC18F2682/2685/4682/4685 DEVICES (CONTINUED)

Address	Name	Address	Name	Address	Name	Address	Name
DFFh	—	DDFh	—	DBFh	—	D9Fh	—
DFEh	—	DDEh	—	DBEh	—	D9Eh	—
DFDh	—	DDDh	—	DBDh	—	D9Dh	—
DFCh	TXBIE	DDCh	—	DBCh	—	D9Ch	—
DFBh	—	DDBh	—	DBBh	—	D9Bh	—
DFAh	BIE0	DDAh	—	DBAh	—	D9Ah	—
DF9h	—	DD9h	—	DB9h	—	D99h	—
DF8h	BSEL0	DD8h	SDFLC	DB8h	—	D98h	—
DF7h	—	DD7h	—	DB7h	—	D97h	—
DF6h	—	DD6h	—	DB6h	—	D96h	—
DF5h	—	DD5h	RXFCON1	DB5h	—	D95h	—
DF4h	—	DD4h	RXFCON0	DB4h	—	D94h	—
DF3h	MSEL3	DD3h	—	DB3h	—	D93h	RXF15EIDL
DF2h	MSEL2	DD2h	—	DB2h	—	D92h	RXF15EIDH
DF1h	MSEL1	DD1h	—	DB1h	—	D91h	RXF15SIDL
DF0h	MSEL0	DD0h	—	DB0h	—	D90h	RXF15SIDH
DEFh	—	DCFh	—	DAFh	—	D8Fh	—
DEEh	—	DCEh	—	DAEh	—	D8Eh	—
DEDh	—	DCDh	—	DADh	—	D8Dh	—
DECh	—	DCCh	—	DACh	—	D8Ch	—
DEBh	—	DCBh	—	DABh	—	D8Bh	RXF14EIDL
DEAh	—	DCAh	—	DAAh	—	D8Ah	RXF14EIDH
DE9h	—	DC9h	—	DA9h	—	D89h	RXF14SIDL
DE8h	—	DC8h	—	DA8h	—	D88h	RXF14SIDH
DE7h	RXFBCON7	DC7h	—	DA7h	—	D87h	RXF13EIDL
DE6h	RXFBCON6	DC6h	—	DA6h	—	D86h	RXF13EIDH
DE5h	RXFBCON5	DC5h	—	DA5h	—	D85h	RXF13SIDL
DE4h	RXFBCON4	DC4h	—	DA4h	—	D84h	RXF13SIDH
DE3h	RXFBCON3	DC3h	—	DA3h	—	D83h	RXF12EIDL
DE2h	RXFBCON2	DC2h	—	DA2h	—	D82h	RXF12EIDH
DE1h	RXFBCON1	DC1h	—	DA1h	—	D81h	RXF12SIDL
DE0h	RXFBCON0	DC0h	—	DA0h	—	D80h	RXF12SIDH

Note 1: Registers available only on PIC18F4X8X devices; otherwise, the registers read as '0'.
2: When any TX_ENn bit in RX_TX_SELn is set, then the corresponding bit in this register has transmit properties.
3: This is not a physical register.

TABLE 5-2: REGISTER FILE SUMMARY (PIC18F2682/2685/4682/4685) (CONTINUED)

File Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Details on page:
B2EIDL(8)	EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0	xxxx xxxx	60, 301
B2EIDH(8)	EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8	xxxx xxxx	60, 301
B2SIDL(8) Receive mode	SID2	SID1	SID0	SRR	EXID	—	EID17	EID16	xxxx x-xx	58, 300
B2SIDL(8) Transmit mode	SID2	SID1	SID0	—	EXIDE	—	EID17	EID16	xxx- x-xx	58, 300
B2SIDH(8)	SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3	xxxx xxxx	60, 299
B2CON(8) Receive mode	RXFUL	RXM1	RXRTRRO	FILHIT4	FILHIT3	FILHIT2	FILHIT1	FILHIT0	0000 0000	60, 298
B2CON(8) Transmit mode	TXBIF	RXM1	TXLARB	TXERR	TXREQ	RTREN	TXPRI1	TXPRI0	0000 0000	60, 298
B1D7(8)	B1D77	B1D76	B1D75	B1D74	B1D73	B1D72	B1D71	B1D70	xxxx xxxx	60, 302
B1D6(8)	B1D67	B1D66	B1D65	B1D64	B1D63	B1D62	B1D61	B1D60	xxxx xxxx	60, 302
B1D5(8)	B1D57	B1D56	B1D55	B1D54	B1D53	B1D52	B1D51	B1D50	xxxx xxxx	60, 302
B1D4(8)	B1D47	B1D46	B1D45	B1D44	B1D43	B1D42	B1D41	B1D40	xxxx xxxx	60, 302
B1D3(8)	B1D37	B1D36	B1D35	B1D34	B1D33	B1D32	B1D31	B1D30	xxxx xxxx	60, 302
B1D2(8)	B1D27	B1D26	B1D25	B1D24	B1D23	B1D22	B1D21	B1D20	xxxx xxxx	60, 302
B1D1(8)	B1D17	B1D16	B1D15	B1D14	B1D13	B1D12	B1D11	B1D10	xxxx xxxx	60, 302
B1D0(8)	B1D07	B1D06	B1D05	B1D04	B1D03	B1D02	B1D01	B1D00	xxxx xxxx	60, 302
B1DLC(8) Receive mode	—	RXRTR	RB1	RB0	DLC3	DLC2	DLC1	DLC0	-xxx xxxx	58, 303
B1DLC(8) Transmit mode	—	TXRTR	—	—	DLC3	DLC2	DLC1	DLC0	-x-- xxxx	58, 304
B1EIDL(8)	EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0	xxxx xxxx	60, 301
B1EIDH(8)	EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8	xxxx xxxx	60, 301
B1SIDL(8) Receive mode	SID2	SID1	SID0	SRR	EXID	—	EID17	EID16	xxxx x-xx	58, 300
B1SIDL(8) Transmit mode	SID2	SID1	SID0	—	EXIDE	—	EID17	EID16	xxx- x-xx	58, 300
B1SIDH(8)	SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3	xxxx xxxx	60, 299
B1CON(8) Receive mode	RXFUL	RXM1	RXRTRRO	FILHIT4	FILHIT3	FILHIT2	FILHIT1	FILHIT0	0000 0000	60, 298
B1CON(8) Transmit mode	TXBIF	TXABT	TXLARB	TXERR	TXREQ	RTREN	TXPRI1	TXPRI0	0000 0000	60, 298
B0D7(8)	B0D77	B0D76	B0D75	B0D74	B0D73	B0D72	B0D71	B0D70	xxxx xxxx	60, 302
B0D6(8)	B0D67	B0D66	B0D65	B0D64	B0D63	B0D62	B0D61	B0D60	xxxx xxxx	60, 302
B0D5(8)	B0D57	B0D56	B0D55	B0D54	B0D53	B0D52	B0D51	B0D50	xxxx xxxx	60, 302
B0D4(8)	B0D47	B0D46	B0D45	B0D44	B0D43	B0D42	B0D41	B0D40	xxxx xxxx	60, 302
B0D3(8)	B0D37	B0D36	B0D35	B0D34	B0D33	B0D32	B0D31	B0D30	xxxx xxxx	60, 302
B0D2(8)	B0D27	B0D26	B0D25	B0D24	B0D23	B0D22	B0D21	B0D20	xxxx xxxx	60, 302
B0D1(8)	B0D17	B0D16	B0D15	B0D14	B0D13	B0D12	B0D11	B0D10	xxxx xxxx	60, 302
B0D0(8)	B0D07	B0D06	B0D05	B0D04	B0D03	B0D02	B0D01	B0D00	xxxx xxxx	60, 302
B0DLC(8) Receive mode	—	RXRTR	RB1	RB0	DLC3	DLC2	DLC1	DLC0	-xxx xxxx	58, 303

Legend: x = unknown, u = unchanged, - = unimplemented, q = value depends on condition. Shaded cells are unimplemented, read as '0'.

Note 1: Bit 21 of the PC is only available in Test mode and Serial Programming modes.

2: The SBOREN bit is only available when CONFIG2L<1:0> = 01; otherwise, it is disabled and reads as '0'. See **Section 4.4 "Brown-out Reset (BOR)"**.

3: These registers and/or bits are not implemented on PIC18F2682/2685 devices and are read as '0'. Reset values are shown for PIC18F4682/4685 devices; individual unimplemented bits should be interpreted as '—'.

4: The PLLEN bit is only available in specific oscillator configurations; otherwise, it is disabled and reads as '0'. See **Section 2.6.4 "PLL in INTOSC Modes"**.

5: The RE3 bit is only available when Master Clear Reset is disabled (CONFIG3H<7> = 0); otherwise, RE3 reads as '0'. This bit is read-only.

6: RA6/RA7 and their associated latch and direction bits are individually configured as port pins based on various primary oscillator modes. When disabled, these bits read as '0'.

7: CAN bits have multiple functions depending on the selected mode of the CAN module.

8: This register reads all '0's until the ECAN™ technology is set up in Mode 1 or Mode 2.

9: These registers and/or bits are available on PIC18F4682/4685 devices only.

TABLE 5-2: REGISTER FILE SUMMARY (PIC18F2682/2685/4682/4685) (CONTINUED)

File Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Details on page:
B0DLC(8) Transmit mode	—	TXRTR	—	—	DLC3	DLC2	DLC1	DLC0	-x-- xxxx	58, 304
B0EIDL(8)	EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0	xxxx xxxx	61, 301
B0EIDH(8)	EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8	xxxx xxxx	61, 301
B0SIDL(8) Receive mode	SID2	SID1	SID0	SRR	EXID	—	EID17	EID16	xxxx x-xx	58, 300
B0SIDL(8) Transmit mode	SID2	SID1	SID0	—	EXIDE	—	EID17	EID16	xxx- x-xx	58, 300
B0SIDH(8)	SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3	xxxx xxxx	61, 299
B0CON(8) Receive mode	RXFUL	RXM1	RXRTRRO	FILHIT4	FILHIT3	FILHIT2	FILHIT1	FILHIT0	0000 0000	60, 298
B0CON(8) Transmit mode	TXBIF	TXABT	TXLARB	TXERR	TXREQ	RTREN	TXPRI1	TXPRI0	0000 0000	60, 298
TXBIE	—	—	—	TXB2IE	TXB1IE	TXB0IE	—	—	---0 00--	61, 321
BIE0	B5IE	B4IE	B3IE	B2IE	B1IE	B0IE	RXB1IE	RXB0IE	0000 0000	61, 321
BSEL0	B5TXEN	B4TXEN	B3TXEN	B2TXEN	B1TXEN	B0TXEN	—	—	0000 00--	61, 304
MSEL3	FIL15_1	FIL15_0	FIL14_1	FIL14_0	FIL13_1	FIL13_0	FIL12_1	FIL12_0	0000 0000	61, 313
MSEL2	FIL11_1	FIL11_0	FIL10_1	FIL10_0	FIL9_1	FIL9_0	FIL8_1	FIL8_0	0000 0000	61, 312
MSEL1	FIL7_1	FIL7_0	FIL6_1	FIL6_0	FIL5_1	FIL5_0	FIL4_1	FIL4_0	0000 0101	61, 311
MSEL0	FIL3_1	FIL3_0	FIL2_1	FIL2_0	FIL1_1	FIL1_0	FIL0_1	FIL0_0	0101 0000	61, 310
RXFBCON7	F15BP_3	F15BP_2	F15BP_1	F15BP_0	F14BP_3	F14BP_2	F14BP_1	F14BP_0	0000 0000	61, 309
RXFBCON6	F13BP_3	F13BP_2	F13BP_1	F13BP_0	F12BP_3	F12BP_2	F12BP_1	F12BP_0	0000 0000	61, 309
RXFBCON5	F11BP_3	F11BP_2	F11BP_1	F11BP_0	F10BP_3	F10BP_2	F10BP_1	F10BP_0	0000 0000	61, 309
RXFBCON4	F9BP_3	F9BP_2	F9BP_1	F9BP_0	F8BP_3	F8BP_2	F8BP_1	F8BP_0	0000 0000	61, 309
RXFBCON3	F7BP_3	F7BP_2	F7BP_1	F7BP_0	F6BP_3	F6BP_2	F6BP_1	F6BP_0	0000 0000	61, 309
RXFBCON2	F5BP_3	F5BP_2	F5BP_1	F5BP_0	F4BP_3	F4BP_2	F4BP_1	F4BP_0	0001 0001	61, 309
RXFBCON1	F3BP_3	F3BP_2	F3BP_1	F3BP_0	F2BP_3	F2BP_2	F2BP_1	F2BP_0	0001 0001	61, 309
RXFBCON0	F1BP_3	F1BP_2	F1BP_1	F1BP_0	F0BP_3	F0BP_2	F0BP_1	F0BP_0	0000 0000	61, 309
SDFLC	—	—	—	FLC4	FLC3	FLC2	FLC1	FLC0	---0 0000	61, 308
RXFCON1	RXF15EN	RXF14EN	RXF13EN	RXF12EN	RXF11EN	RXF10EN	RXF9EN	RXF8EN	0000 0000	61, 308
RXFCON0	RXF7EN	RXF6EN	RXF5EN	RXF4EN	RXF3EN	RXF2EN	RXF1EN	RXF0EN	0000 0000	61, 308
RXF15EIDL	EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0	xxxx xxxx	61, 306
RXF15EIDH	EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8	xxxx xxxx	61, 306
RXF15SIDL	SID2	SID1	SID0	—	EXIDEN	—	EID17	EID16	xxx- x-xx	61, 305
RXF15SIDH	SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3	xxxx xxxx	61, 305
RXF14EIDL	EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0	xxxx xxxx	61, 306
RXF14EIDH	EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8	xxxx xxxx	61, 306
RXF14SIDL	SID2	SID1	SID0	—	EXIDEN	—	EID17	EID16	xxx- x-xx	61, 305
RXF14SIDH	SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3	xxxx xxxx	61, 305
RXF13EIDL	EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0	xxxx xxxx	62, 306
RXF13EIDH	EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8	xxxx xxxx	62, 306
RXF13SIDL	SID2	SID1	SID0	—	EXIDEN	—	EID17	EID16	xxx- x-xx	62, 305
RXF13SIDH	SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3	xxxx xxxx	62, 305

Legend: x = unknown, u = unchanged, - = unimplemented, q = value depends on condition. Shaded cells are unimplemented, read as '0'.

Note 1: Bit 21 of the PC is only available in Test mode and Serial Programming modes.

2: The SBOREN bit is only available when CONFIG2L<1:0> = 01; otherwise, it is disabled and reads as '0'. See **Section 4.4 "Brown-out Reset (BOR)"**.

3: These registers and/or bits are not implemented on PIC18F2682/2685 devices and are read as '0'. Reset values are shown for PIC18F4682/4685 devices; individual unimplemented bits should be interpreted as '—'.

4: The PLLEN bit is only available in specific oscillator configurations; otherwise, it is disabled and reads as '0'. See **Section 2.6.4 "PLL in INTOSC Modes"**.

5: The RE3 bit is only available when Master Clear Reset is disabled (CONFIG3H<7> = 0); otherwise, RE3 reads as '0'. This bit is read-only.

6: RA6/RA7 and their associated latch and direction bits are individually configured as port pins based on various primary oscillator modes. When disabled, these bits read as '0'.

7: CAN bits have multiple functions depending on the selected mode of the CAN module.

8: This register reads all '0's until the ECAN™ technology is set up in Mode 1 or Mode 2.

9: These registers and/or bits are available on PIC18F4682/4685 devices only.

EXAMPLE 6-3: WRITING TO FLASH PROGRAM MEMORY

```
                MOVLW   D'64                    ; 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   D’64                    ; number of bytes in holding register
                MOVWF   COUNTER
WRITE_BYTE_TO_HREGS
                MOVF    POSTINC0, W             ; 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
```

8.0 8 x 8 HARDWARE MULTIPLIER

8.1 Introduction

All PIC18 devices include an 8 x 8 hardware multiplier as part of the ALU. The multiplier performs an unsigned operation and yields a 16-bit result that is stored in the product register pair, PRODH:PRODL. The multiplier's operation does not affect any flags in the STATUS register.

Making multiplication a hardware operation allows it to be completed in a single instruction cycle. This has the advantages of higher computational throughput and reduced code size for multiplication algorithms and allows the PIC18 devices to be used in many applications previously reserved for digital signal processors. A comparison of various hardware and software multiply operations, along with the savings in memory and execution time, is shown in Table 8-1.

8.2 Operation

Example 8-1 shows the instruction sequence for an 8 x 8 unsigned multiplication. Only one instruction is required when one of the arguments is already loaded in the WREG register.

Example 8-2 shows the sequence to do an 8 x 8 signed multiplication. To account for the signed bits of the arguments, each argument's Most Significant bit (MSb) is tested and the appropriate subtractions are done.

EXAMPLE 8-1: 8 x 8 UNSIGNED MULTIPLY ROUTINE

```
MOVF    ARG1, W     ;
MULWF   ARG2        ; ARG1 * ARG2 ->
                    ; PRODH:PRODL
```

EXAMPLE 8-2: 8 x 8 SIGNED MULTIPLY ROUTINE

```
MOVF    ARG1, W
MULWF   ARG2        ; ARG1 * ARG2 ->
                    ; PRODH:PRODL
BTFSC   ARG2, SB    ; Test Sign Bit
SUBWF   PRODH, F    ; PRODH = PRODH
                    ;         - ARG1
MOVF    ARG2, W
BTFSC   ARG1, SB    ; Test Sign Bit
SUBWF   PRODH, F    ; PRODH = PRODH
                    ;         - ARG2
```

TABLE 8-1: PERFORMANCE COMPARISON FOR VARIOUS MULTIPLY OPERATIONS

Routine	Multiply Method	Program Memory (Words)	Cycles (Max)	Time		
				@ 40 MHz	@ 10 MHz	@ 4 MHz
8 x 8 unsigned	Without hardware multiply	13	69	6.9 μs	27.6 μs	69 μs
	Hardware multiply	1	1	100 ns	400 ns	1 μs
8 x 8 signed	Without hardware multiply	33	91	9.1 μs	36.4 μs	91 μs
	Hardware multiply	6	6	600 ns	2.4 μs	6 μs
16 x 16 unsigned	Without hardware multiply	21	242	24.2 μs	96.8 μs	242 μs
	Hardware multiply	28	28	2.8 μs	11.2 μs	28 μs
16 x 16 signed	Without hardware multiply	52	254	25.4 μs	102.6 μs	254 μs
	Hardware multiply	35	40	4.0 μs	16.0 μs	40 μs

REGISTER 9-2: INTCON2: INTERRUPT CONTROL REGISTER 2

R/W-1	R/W-1	R/W-1	R/W-1	U-0	R/W-1	U-0	R/W-1
$\overline{\text{RBPU}}$	INTEDG0	INTEDG1	INTEDG2	—	TMR0IP	—	RBIP
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 **$\overline{\text{RBPU}}$:** PORTB Pull-up Enable bit
1 = All PORTB pull-ups are disabled
0 = PORTB pull-ups are enabled by individual port latch values

bit 6 **INTEDG0:** External Interrupt 0 Edge Select bit
1 = Interrupt on rising edge
0 = Interrupt on falling edge

bit 5 **INTEDG1:** External Interrupt 1 Edge Select bit
1 = Interrupt on rising edge
0 = Interrupt on falling edge

bit 4 **INTEDG2:** External Interrupt 2 Edge Select bit
1 = Interrupt on rising edge
0 = Interrupt on falling edge

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

bit 2 **TMR0IP:** TMR0 Overflow Interrupt Priority bit
1 = High priority
0 = Low priority

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

bit 0 **RBIP:** RB Port Change Interrupt Priority bit
1 = High priority
0 = Low priority

Note: Interrupt flag bits are set when an interrupt condition occurs regardless of the state of its corresponding enable bit or the global interrupt enable bit. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt. This feature allows for software polling.

TABLE 10-8: SUMMARY OF REGISTERS ASSOCIATED WITH PORTD

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset Values on page
PORTD(1)	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	54
LATD(1)	LATD Data Output Register								54
TRISD(1)	PORTD Data Direction Register								54
TRISE(1)	IBF	OBF	IBOV	PSPMODE	—	TRISE2	TRISE1	TRISE0	54
ECCP1CON(1)	EPWM1M1	EPWM1M0	EDC1B1	EDC1B0	ECCP1M3	ECCP1M2	ECCP1M1	ECCP1M0	53

Legend: — = unimplemented, read as '0'. Shaded cells are not used by PORTD.
Note 1: These registers are available on PIC18F4682/4685 devices only.

TABLE 15-5: REGISTERS ASSOCIATED WITH PWM AND TIMER2

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset Values on page
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	51
RCON	IPEN	SBOREN(2)	—	$\overline{RI}$	$\overline{TO}$	$\overline{PD}$	$\overline{POR}$	$\overline{BOR}$	52
PIR1	PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	54
PIE1	PSPIE(1)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	54
IPR1	PSPIP(1)	ADIP	RCIP	TXIP	SSPIP	CCP1IP	TMR2IP	TMR1IP	54
TRISB	PORTB Data Direction Register								54
TRISC	PORTC Data Direction Register								54
TMR2	Timer2 Register								52
PR2	Timer2 Period Register								52
T2CON	—	T2OUTPS3	T2OUTPS2	T2OUTPS1	T2OUTPS0	TMR2ON	T2CKPS1	T2CKPS0	52
CCPR1L	Capture/Compare/PWM Register 1 Low Byte								53
CCPR1H	Capture/Compare/PWM Register 1 High Byte								53
CCP1CON	—	—	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	53
ECCPR1L(1)	Enhanced Capture/Compare/PWM Register 1 Low Byte								53
ECCPR1H(1)	Enhanced Capture/Compare/PWM Register 1 High Byte								53
ECCP1CON(1)	EPWM1M1	EPWM1M0	EDC1B1	EDC1B0	ECCP1M3	ECCP1M2	ECCP1M1	ECCP1M0	53

Legend: — = unimplemented, read as '0'. Shaded cells are not used by PWM or Timer2.

Note 1: These bits or registers are available on PIC18F4682/4685 devices only.

2: The SBOREN bit is only available when CONFIG2L<1:0> = 01; otherwise, it is disabled and reads as '0'.

17.4.2 OPERATION

The MSSP module functions are enabled by setting MSSP Enable bit, SSPEN (SSPCON<5>).

The SSPCON1 register allows control of the I^2C operation. Four mode selection bits (SSPCON<3:0>) allow one of the following I^2C modes to be selected:

- I^2C Master mode, clock = (FOSC/4) x (SSPADD + 1)
- I^2C Slave mode (7-bit address)
- I^2C Slave mode (10-bit address)
- I^2C Slave mode (7-bit address) with Start and Stop bit interrupts enabled
- I^2C Slave mode (10-bit address) with Start and Stop bit interrupts enabled
- I^2C Firmware Controlled Master mode, slave is Idle

Selection of any I^2C mode with the SSPEN bit set, forces the SCL and SDA pins to be open-drain, provided these pins are programmed to inputs by setting the appropriate TRISC bits. To ensure proper operation of the module, pull-up resistors must be provided externally to the SCL and SDA pins.

17.4.3 SLAVE MODE

In Slave mode, the SCL and SDA pins must be configured as inputs (TRISC<4:3> set). The MSSP module will override the input state with the output data when required (slave-transmitter).

The I^2C Slave mode hardware will always generate an interrupt on an address match. Through the mode select bits, the user can also choose to interrupt on Start and Stop bits

When an address is matched, or the data transfer after an address match is received, the hardware automatically will generate the Acknowledge ($\overline{ACK}$) pulse and load the SSPBUF register with the received value currently in the SSPSR register.

Any combination of the following conditions will cause the MSSP module not to give this $\overline{ACK}$ pulse:

- The Buffer Full bit, BF (SSPSTAT<0>), was set before the transfer was received.
- The overflow bit, SSPOV (SSPCON<6>), was set before the transfer was received.

In this case, the SSPSR register value is not loaded into the SSPBUF, but bit SSPIF (PIR1<3>) is set. The BF bit is cleared by reading the SSPBUF register, while bit SSPOV is cleared through software.

The SCL clock input must have a minimum high and low for proper operation. The high and low times of the I^2C specification, as well as the requirement of the MSSP module, are shown in timing parameter 100 and parameter 101.

17.4.3.1 Addressing

Once the MSSP module has been enabled, it waits for a Start condition to occur. Following the Start condition, the 8-bits are shifted into the SSPSR register. All incoming bits are sampled with the rising edge of the clock (SCL) line. The value of register SSPSR<7:1> is compared to the value of the SSPADD register. The address is compared on the falling edge of the eighth clock (SCL) pulse. If the addresses match and the BF and SSPOV bits are clear, the following events occur:

1. The SSPSR register value is loaded into the SSPBUF register.
2. The Buffer Full bit, BF, is set.
3. An $\overline{ACK}$ pulse is generated.
4. MSSP Interrupt Flag bit, SSPIF (PIR1<3>), is set (interrupt is generated, if enabled) on the falling edge of the ninth SCL pulse.

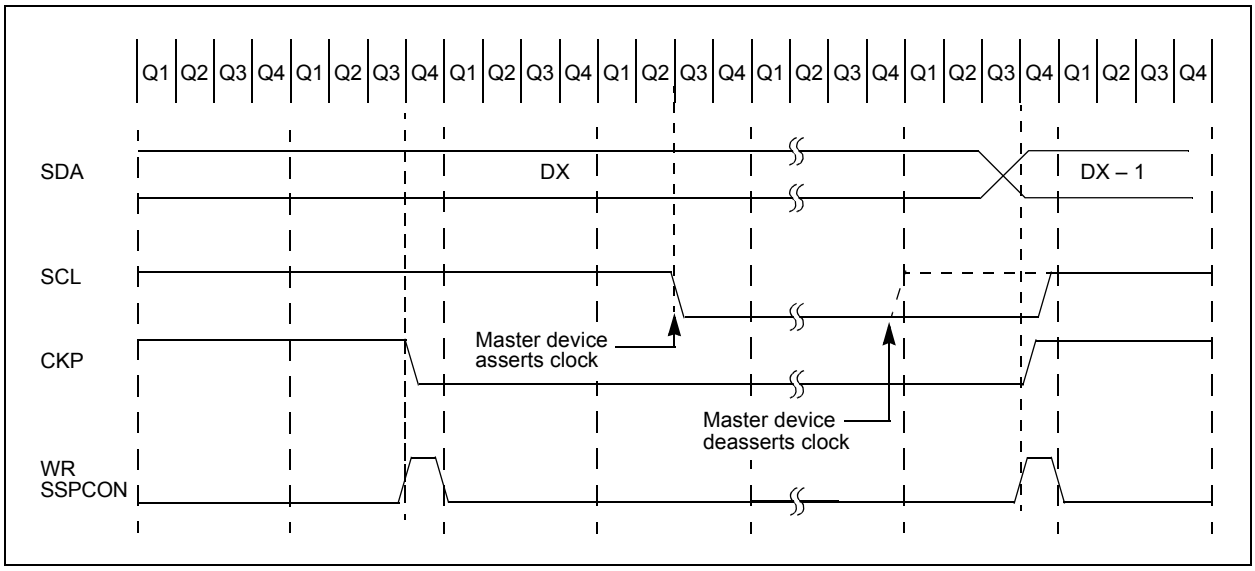
In 10-Bit Address mode, two address bytes need to be received by the slave. The five Most Significant bits (MSbs) of the first address byte specify if this is a 10-bit address. Bit R/$\overline{W}$ (SSPSTAT<2>) must specify a write so the slave device will receive the second address byte. For a 10-bit address, the first byte would equal '11110 A9 A8 0', where 'A9' and 'A8' are the two MSbs of the address. The sequence of events for 10-bit address is as follows, with steps 7 through 9 for the slave-transmitter:

1. Receive first (high) byte of address (bits SSPIF, BF and UA (SSPSTAT<1>) are set).
2. Update the SSPADD register with second (low) byte of address (clears bit UA and releases the SCL line).
3. Read the SSPBUF register (clears bit BF) and clear flag bit, SSPIF.
4. Receive second (low) byte of address (bits SSPIF, BF and UA are set).
5. Update the SSPADD register with the first (high) byte of address. If match releases SCL line, this will clear bit UA.
6. Read the SSPBUF register (clears bit BF) and clear flag bit, SSPIF.
7. Receive Repeated Start condition.
8. Receive first (high) byte of address (bits SSPIF and BF are set).
9. Read the SSPBUF register (clears bit BF) and clear flag bit, SSPIF.

17.4.4.5 Clock Synchronization and the CKP bit

When the CKP bit is cleared, the SCL output is forced to '0'. However, setting the CKP bit will not assert the SCL output low until the SCL output is already sampled low. Therefore, the CKP bit will not assert the SCL line until an external I^2C master device has already asserted the SCL line. The SCL output will remain low until the CKP bit is set and all other devices on the I^2C bus have deasserted SCL. This ensures that a write to the CKP bit will not violate the minimum high time requirement for SCL (see Figure 17-12).

FIGURE 17-12: CLOCK SYNCHRONIZATION TIMING



REGISTER 23-47: RXFBCONn: RECEIVE FILTER BUFFER CONTROL REGISTER n(1)

RXFBCON0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	F1BP_3	F1BP_2	F1BP_1	F1BP_0	F0BP_3	F0BP_2	F0BP_1	F0BP_0
RXFBCON1	R/W-0	R/W-0	R/W-0	R/W-1	R/W-0	R/W-0	R/W-0	R/W-1
	F3BP_3	F3BP_2	F3BP_1	F3BP_0	F2BP_3	F2BP_2	F2BP_1	F2BP_0
RXFBCON2	R/W-0	R/W-0	R/W-0	R/W-1	R/W-0	R/W-0	R/W-0	R/W-1
	F5BP_3	F5BP_2	F5BP_1	F5BP_0	F4BP_3	F4BP_2	F4BP_1	F4BP_0
RXFBCON3	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	F7BP_3	F7BP_2	F7BP_1	F7BP_0	F6BP_3	F6BP_2	F6BP_1	F6BP_0
RXFBCON4	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	F9BP_3	F9BP_2	F9BP_1	F9BP_0	F8BP_3	F8BP_2	F8BP_1	F8BP_0
RXFBCON5	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	F11BP_3	F11BP_2	F11BP_1	F11BP_0	F10BP_3	F10BP_2	F10BP_1	F10BP_0
RXFBCON6	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	F13BP_3	F13BP_2	F13BP_1	F13BP_0	F12BP_3	F12BP_2	F12BP_1	F12BP_0
RXFBCON7	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	F15BP_3	F15BP_2	F15BP_1	F15BP_0	F14BP_3	F14BP_2	F14BP_1	F14BP_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 7-0 **FnBP_3:FnBP_0:** Filter n Buffer Pointer Nibble bits

0000 = Filter n is associated with RXB0
0001 = Filter n is associated with RXB1
0010 = Filter n is associated with B0
0011 = Filter n is associated with B1
...
0111 = Filter n is associated with B5
1111-1000 = Reserved

Note 1: This register is available in Mode 1 and 2 only.

REGISTER 24-10: CONFIG7L: CONFIGURATION REGISTER 7 LOW (BYTE ADDRESS 30000Ch)

U-0	U-0	R/C-1	R/C-1	R/C-1	R/C-1	R/C-1	R/C-1
—	—	EBTR5(1)	EBTR4	EBTR3	EBTR2	EBTR1	EBTR0
bit 7							bit 0

Legend:		
R = Readable bit	C = Clearable bit	U = Unimplemented bit, read as '0'
-n = Value when device is unprogrammed		u = Unchanged from programmed state

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

bit 5 **EBTR5:** Table Read Protection bit(1)
1 = Block 5 (014000-017FFFh) not protected from table reads executed in other blocks
0 = Block 5 (014000-017FFFh) protected from table reads executed in other blocks

bit 4 **EBTR4:** Table Read Protection bit
1 = Block 4 (010000-013FFFh) not protected from table reads executed in other blocks
0 = Block 4 (010000-013FFFh) protected from table reads executed in other blocks

bit 3 **EBTR3:** Table Read Protection bit
1 = Block 3 (00C000-00FFFFh) not protected from table reads executed in other blocks
0 = Block 3 (00C000-00FFFFh) protected from table reads executed in other blocks

bit 2 **EBTR2**: Table Read Protection bit(1)
1 = Block 2 (008000-00BFFFh) not protected from table reads executed in other blocks
0 = Block 2 (008000-00BFFFh) protected from table reads executed in other blocks

bit 1 **EBTR1:** Table Read Protection bit
1 = Block 1 (004000-007FFFh) not protected from table reads executed in other blocks
0 = Block 1 (004000-007FFFh) protected from table reads executed in other blocks

bit 0 **EBTR0:** Table Read Protection bit
1 = Block 0 (000800-003FFFh) not protected from table reads executed in other blocks
0 = Block 0 (000800-003FFFh) protected from table reads executed in other blocks

Note 1: Unimplemented in PIC18F2682/4682 devices; maintain this bit set.

REGISTER 24-11: CONFIG7H: CONFIGURATION REGISTER 7 HIGH (BYTE ADDRESS 30000Dh)

U-0	R/C-1	U-0	U-0	U-0	U-0	U-0	U-0
—	EBTRB	—	—	—	—	—	—
bit 7							bit 0

Legend:		
R = Readable bit	C = Clearable bit	U = Unimplemented bit, read as '0'
-n = Value when device is unprogrammed		u = Unchanged from programmed state

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

bit 6 **EBTRB:** Boot Block Table Read Protection bit
1 = Boot Block (000000-0007FFh) not protected from table reads executed in other blocks
0 = Boot Block (000000-0007FFh) protected from table reads executed in other blocks

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

24.5 Program Verification and Code Protection

The overall structure of the code protection on the PIC18 Flash devices differs significantly from other PIC devices.

The user program memory is divided into five blocks. One of these is a boot block of 2 Kbytes. The remainder of the memory is divided into four blocks on binary boundaries.

Each of the five blocks has three code protection bits associated with them. They are:

- Code-Protect bit (CPn)
- Write-Protect bit (WRTn)
- External Block Table Read bit (EBTRn)

Figure 24-5 shows the program memory organization for 80- and 96-Kbyte devices and the specific code protection bit associated with each block. The actual locations of the bits are summarized in Table 24-3.

FIGURE 24-5: CODE-PROTECTED PROGRAM MEMORY FOR PIC18F2682/2685/4682/4685

MEMORY SIZE/DEVICE			
80 Kbytes (PIC18F2682/4682)	**96 Kbytes (PIC18F2685/4685)**	**Address Range**	**Block Code Protection Controlled By:**
Boot Block	Boot Block	000000h 0007FFh	CPB, WRTB, EBTRB
Block 0	Block 0	000800h 003FFFh	CP0, WRT0, EBTR0
Block 1	Block 1	004000h 007FFFh	CP1, WRT1, EBTR1
Block 2	Block 2	008000h 00BFFFh	CP2, WRT2, EBTR2
Block 3	Block 3	00C000h 00FFFFh	CP3, WRT3, EBTR3
Block 4	Block 4	010000h 013FFFh	CP4, WRT4, EBTR4
Unimplemented Read '0's	Block 5	014000h 017FFFh	CP5, WRT5, EBTR5
Unimplemented Read '0's	Unimplemented Read '0's	018000h 1FFFFFh	(Unimplemented Memory Space)

TABLE 25-1: OPCODE FIELD DESCRIPTIONS

Field	Description
a	RAM access bit a = 0: RAM location in Access RAM (BSR register is ignored) a = 1: RAM bank is specified by BSR register
bbb	Bit address within an 8-bit file register (0 to 7).
BSR	Bank Select Register. Used to select the current RAM bank.
C, DC, Z, OV, N	ALU Status bits: **C**arry, **D**igit **C**arry, **Z**ero, **Ov**erflow, **N**egative.
d	Destination select bit d = 0: store result in WREG d = 1: store result in file register f
dest	Destination: either the WREG register or the specified register file location.
f	8-bit Register file address (00h to FFh), or 2-bit FSR designator (0h to 3h).
f_s	12-bit Register file address (000h to FFFh). This is the source address.
f_d	12-bit Register file address (000h to FFFh). This is the destination address.
GIE	Global Interrupt Enable bit.
k	Literal field, constant data or label (may be either an 8-bit, 12-bit or a 20-bit value)
label	Label name
mm * *+ *- +*	The mode of the TBLPTR register for the table read and table write instructions. Only used with table read and table write instructions: No change to register (such as TBLPTR with table reads and writes) Post-Increment register (such as TBLPTR with table reads and writes) Post-Decrement register (such as TBLPTR with table reads and writes) Pre-Increment register (such as TBLPTR with table reads and writes)
n	The relative address (2's complement number) for relative branch instructions or the direct address for Call/Branch and Return instructions
PC	Program Counter.
PCL	Program Counter Low Byte.
PCH	Program Counter High Byte.
PCLATH	Program Counter High Byte Latch.
PCLATU	Program Counter Upper Byte Latch.
$\overline{PD}$	Power-down bit.
PRODH	Product of Multiply High Byte.
PRODL	Product of Multiply Low Byte.
s	Fast Call/Return mode select bit s = 0: do not update into/from shadow registers s = 1: certain registers loaded into/from shadow registers (Fast mode)
TBLPTR	21-bit Table Pointer (points to a Program Memory location).
TABLAT	8-bit Table Latch.
$\overline{TO}$	Time-out bit.
TOS	Top-of-Stack.
u	Unused or unchanged.
WDT	Watchdog Timer.
WREG	Working register (accumulator).
x	Don't care ('0' or '1'). The assembler will generate code with x = 0. It is the recommended form of use for compatibility with all Microchip software tools.
z_s	7-bit offset value for indirect addressing of register files (source).
z_d	7-bit offset value for indirect addressing of register files (destination).
{ }	Optional argument.
[text]	Indicates an indexed address.
(text)	The contents of text.
[expr]<n>	Specifies bit n of the register indicated by the pointer expr.
→	Assigned to.
< >	Register bit field.
∈	In the set of.
italics	User defined term (font is Courier).

CLRF **Clear f**

Syntax: CLRF f {,a}

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

Operation: 000h → f
1 → Z

Status Affected: Z

Encoding:

0110	101a	ffff	ffff

Description: Clears the contents of the specified register.
If 'a' is '0', the Access Bank is selected. If 'a' is '1', the BSR is used to select the GPR bank (default).
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 25.2.3 "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: CLRF FLAG_REG,1

Before Instruction
FLAG_REG = 5Ah
After Instruction
FLAG_REG = 00h

CLRWDT **Clear Watchdog Timer**

Syntax: CLRWDT

Operands: None

Operation: 000h → WDT,
000h → WDT postscaler,
1 → $\overline{TO}$,
1 → $\overline{PD}$

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

Encoding:

0000	0000	0000	0100

Description: CLRWDT instruction resets the Watchdog Timer. It also resets the postscaler of the WDT. Status bits $\overline{TO}$ and $\overline{PD}$ are set.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	No operation	Process Data	No operation

Example: CLRWDT

Before Instruction
WDT Counter = ?
After Instruction
WDT Counter = 00h
WDT Postscaler = 0
$\overline{TO}$ = 1
$\overline{PD}$ = 1

27.2 DC Characteristics: Power-Down and Supply Current PIC18F2682/2685/4682/4685 (Industrial) PIC18LF2682/2685/4682/4685 (Industrial) (Continued)

PIC18LF2682/2685/4682/4685 (Industrial)		**Standard Operating Conditions (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +85°C for industrial					
PIC18F2682/2685/4682/4685 (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)(2,3)						
	PIC18LF268X/468X	2.9	8	µA	-40°C	VDD = 2.0V	FOSC = 31 kHz (**RC_IDLE** mode, Internal oscillator source)
		3.1	8	µA	+25°C		
		3.6	12	µA	+85°C		
	PIC18LF268X/468X	4.5	12	µA	-40°C	VDD = 3.0V	
		4.8	12	µA	+25°C		
		5.8	17	µA	+85°C		
	All devices	9.2	25	µA	-40°C	VDD = 5.0V	
		9.8	25	µA	+25°C		
		11.4	36	µA	+85°C		
	Extended devices only	21	180	µA	+125°C		
	PIC18LF268X/468X	165	400	µA	-40°C	VDD = 2.0V	FOSC = 1 MHz (**RC_IDLE** mode, Internal oscillator source)
		175	400	µA	+25°C		
		190	400	µA	+85°C		
	PIC18LF268X/468X	250	600	µA	-40°C	VDD = 3.0V	
		270	600	µA	+25°C		
		290	600	µA	+85°C		
	All devices	0.5	1	mA	-40°C	VDD = 5.0V	
		0.5	1	mA	+25°C		
		0.5	1	mA	+85°C		
	Extended devices only	0.6	1.4	mA	+125°C		
	PIC18LF268X/468X	0.34	1.1	mA	-40°C	VDD = 2.0V	FOSC = 4 MHz (**RC_IDLE** mode, Internal oscillator source)
		0.35	1.1	mA	+25°C		
		0.36	1.1	mA	+85°C		
	PIC18LF268X/468X	0.52	1.5	mA	-40°C	VDD = 3.0V	
		0.54	1.5	mA	+25°C		
		0.58	1.5	mA	+85°C		
	All devices	1	2.7	mA	-40°C	VDD = 5.0V	
		1.1	2.7	mA	+25°C		
		1.1	2.7	mA	+85°C		
	Extended devices only	1.1	3.6	mA	+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 high-impedance state and tied to VDD or VSS and all features that add delta current disabled (such as WDT, Timer1 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;
MCLR = VDD; WDT enabled/disabled as specified.

3: For RC oscillator configurations, current through REXT is not included. The current through the resistor can be estimated by the formula Ir = VDD/2REXT (mA) with REXT in kΩ.

4: 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.

PIC18F2682/2685/4682/4685 PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.

PART NO.	X	/XX	XXX
Device	Temperature Range	Package	Pattern

Device	PIC18F2682/2685(1), PIC18F4682/4685(1), PIC18F2682/2685T(2), PIC18F4682/4685T(2); VDD range 4.2V to 5.5V PIC18LF2682/2685(1), PIC18LF4682/4685(1), PIC18LF2682/2685T(2), PIC18LF4682/4685T(2); VDD range 2.0V to 5.5V
Temperature Range	I = -40°C to +85°C (Industrial)
Package	PT = TQFP (Thin Quad Flatpack) SO = SOIC SP = Skinny Plastic DIP P = PDIP ML = QFN
Pattern	QTP, SQTP, Code or Special Requirements (blank otherwise)

Examples:

a) PIC18LF4685-I/P 301 = Industrial temp., PDIP package, Extended VDD limits, QTP pattern #301.

b) PIC18LF2685-I/SO = Industrial temp., SOIC package, Extended VDD limits.

c) PIC18F4685-I/P = Industrial temp., PDIP package, normal VDD limits.

Note 1: F = Standard Voltage Range
LF = Wide Voltage Range

2: T = in tape and reel PLCC and TQFP packages only.