



Welcome to [E-XFL.COM](https://www.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	Obsolete
Core Processor	PIC
Core Size	8-Bit
Speed	40MHz
Connectivity	CANbus, I ² C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LVD, POR, PWM, WDT
Number of I/O	52
Program Memory Size	48KB (24K x 16)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	3.25K x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5.5V
Data Converters	A/D 12x10b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-TQFP
Supplier Device Package	64-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18lf6585t-i-pt

PIC18F6585/8585/6680/8680

Table of Contents

1.0	Device Overview	9
2.0	Oscillator Configurations	23
3.0	Reset	33
4.0	Memory Organization	51
5.0	Flash Program Memory	83
6.0	External Memory Interface	93
7.0	Data EEPROM Memory	101
8.0	8 x 8 Hardware Multiplier	107
9.0	Interrupts	109
10.0	I/O Ports	125
11.0	Timer0 Module	155
12.0	Timer1 Module	159
13.0	Timer2 Module	162
14.0	Timer3 Module	164
15.0	Capture/Compare/PWM (CCP) Modules	167
16.0	Enhanced Capture/Compare/PWM (ECCP) Module	175
17.0	Master Synchronous Serial Port (MSSP) Module	189
18.0	Enhanced Universal Synchronous Asynchronous Receiver Transmitter (USART)	229
19.0	10-bit Analog-to-Digital Converter (A/D) Module	249
20.0	Comparator Module	259
21.0	Comparator Voltage Reference Module	265
22.0	Low-Voltage Detect	269
23.0	ECAN Module	275
24.0	Special Features of the CPU	345
25.0	Instruction Set Summary	365
26.0	Development Support	407
27.0	Electrical Characteristics	413
28.0	DC and AC Characteristics Graphs and Tables	449
29.0	Packaging Information	465
	Appendix A: Revision History	469
	Appendix B: Device Differences	469
	Appendix C: Conversion Considerations	470
	Appendix D: Migration from Mid-Range to Enhanced Devices	470
	Appendix E: Migration from High-End to Enhanced Devices	471
	Index	473
	On-Line Support	487
	Systems Information and Upgrade Hot Line	487
	Reader Response	488
	PIC18F6585/8585/6680/8680 Product Identification System	489

PIC18F6585/8585/6680/8680

TABLE 3-3: 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
TXB0SIDH	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB0CON	PIC18F6X8X	PIC18F8X8X	0000 0-00	0000 0-00	uuuu u-uu
TXB1D7	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB1D6	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB1D5	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB1D4	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB1D3	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB1D2	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB1D1	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB1D0	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB1DLC	PIC18F6X8X	PIC18F8X8X	-x- - xxxx	-u- - uuuu	-u- - uuuu
TXB1EIDL	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB1EIDH	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB1SIDL	PIC18F6X8X	PIC18F8X8X	xxx- x-xx	uuu- u-uu	uuu- uu-u
TXB1SIDH	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	-uuu uuuu
TXB1CON	PIC18F6X8X	PIC18F8X8X	0000 0-00	0000 0-00	uuuu u-uu
TXB2D7	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	0uuu uuuu
TXB2D6	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	0uuu uuuu
TXB2D5	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	0uuu uuuu
TXB2D4	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	0uuu uuuu
TXB2D3	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	0uuu uuuu
TXB2D2	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	0uuu uuuu
TXB2D1	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	0uuu uuuu
TXB2D0	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	0uuu uuuu
TXB2DLC	PIC18F6X8X	PIC18F8X8X	-x- - xxxx	-u- - uuuu	-u- - uuuu
TXB2EIDL	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB2EIDH	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB2SIDL	PIC18F6X8X	PIC18F8X8X	xxx- x-xx	uuu- u-uu	uuu- u-uu
TXB2SIDH	PIC18F6X8X	PIC18F8X8X	xxx- x-xx	uuu- u-uu	uuu- u-uu
TXB2CON	PIC18F6X8X	PIC18F8X8X	0000 0-00	0000 0-00	uuuu u-uu
XXM1EIDL	PIC18F6X8X	PIC18F8X8X	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 3-2 for Reset value for specific condition.
- 5:** Bit 6 of PORTA, LATA, and TRISA are enabled in ECIO and RCIO Oscillator modes only. In all other oscillator modes, they are disabled and read '0'.
- 6:** Bit 6 of PORTA, LATA and TRISA are not available on all devices. When unimplemented, they read '0'.
- 7:** This register reads all '0's until ECAN is set up in Mode 1 or Mode 2.

PIC18F6585/8585/6680/8680

TABLE 3-3: 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
B3D4 ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
B3D3 ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
B3D2 ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
B3D1 ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
B3D0 ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
B3DLC ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	-xxx xxxx	-uuu uuuu	-uuu uuuu
B3EIDL ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
B3EIDH ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
B3SIDL ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	xxxx x-xx	uuuu u-uu	uuuu u-uu
B3SIDH ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
B3CON ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	0000 0000	0000 0000	uuuu uuuu
B2D7 ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
B2D6 ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
B2D5 ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
B2D4 ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
B2D3 ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
B2D2 ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
B2D1 ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
B2D0 ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
B2DLC ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	-xxx xxxx	-uuu uuuu	-uuu uuuu
B2EIDL ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
B2EIDH ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
B2SIDL ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	xxxx x-xx	uuuu u-uu	uuuu u-uu
B2SIDH ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
B2CON ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	0000 0000	0000 0000	uuuu uuuu
B1D7 ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
B1D6 ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
B1D5 ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
B1D4 ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
B1D3 ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	xxxx xxxx	uuuu uuuu	uuuu uuuu
B1D2 ⁽⁷⁾	PIC18F6X8X	PIC18F8X8X	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).

- 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).
- 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.
- See Table 3-2 for Reset value for specific condition.
- Bit 6 of PORTA, LATA, and TRISA are enabled in ECIO and RCIO Oscillator modes only. In all other oscillator modes, they are disabled and read '0'.
- Bit 6 of PORTA, LATA and TRISA are not available on all devices. When unimplemented, they read '0'.
- This register reads all '0's until ECAN is set up in Mode 1 or Mode 2.

PIC18F6585/8585/6680/8680

TABLE 4-2: SPECIAL FUNCTION REGISTER MAP

Address	Name	Address	Name	Address	Name	Address	Name
FFFh	TOSU	FDFh	INDF2 ⁽³⁾	FBFh	CCPR1H	F9Fh	IPR1
FFEh	TOSH	FDEh	POSTINC2 ⁽³⁾	FBEh	CCPR1L	F9Eh	PIR1
FFDh	TOSL	FDDh	POSTDEC2 ⁽³⁾	FBDh	CCP1CON	F9Dh	PIE1
FFCh	STKPTR	FDCh	PREINC2 ⁽³⁾	FBCh	CCPR2H	F9Ch	MEMCON ⁽²⁾
FFBh	PCLATU	FDBh	PLUSW2 ⁽³⁾	FBBh	CCPR2L	F9Bh	— ⁽¹⁾
FFAh	PCLATH	FDAh	FSR2H	FBAh	CCP2CON	F9Ah	TRISJ ⁽²⁾
FF9h	PCL	FD9h	FSR2L	FB9h	— ⁽¹⁾	F99h	TRISH ⁽²⁾
FF8h	TBLPTRU	FD8h	STATUS	FB8h	— ⁽¹⁾	F98h	TRISG
FF7h	TBLPTRH	FD7h	TMR0H	FB7h	— ⁽¹⁾	F97h	TRISF
FF6h	TBLPTRL	FD6h	TMR0L	FB6h	ECCP1AS	F96h	TRISE
FF5h	TABLAT	FD5h	T0CON	FB5h	CVRCON	F95h	TRISD
FF4h	PRODH	FD4h	— ⁽¹⁾	FB4h	CMCON	F94h	TRISC
FF3h	PRODL	FD3h	OSCCON	FB3h	TMR3H	F93h	TRISB
FF2h	INTCON	FD2h	LVDCON	FB2h	TMR3L	F92h	TRISA
FF1h	INTCON2	FD1h	WDTCN	FB1h	T3CON	F91h	LATJ ⁽²⁾
FF0h	INTCON3	FD0h	RCON	FB0h	PSPCON	F90h	LATH ⁽²⁾
FEFh	INDF0 ⁽³⁾	FCFh	TMR1H	FAFh	SPBRG	F8Fh	LATG
FEeh	POSTINC0 ⁽³⁾	FCEh	TMR1L	FAEh	RCREG	F8Eh	LATF
FEDh	POSTDEC0 ⁽³⁾	FCDh	T1CON	FADh	TXREG	F8Dh	LATE
FECh	PREINC0 ⁽³⁾	CCCh	TMR2	FACH	TXSTA	F8Ch	LATD
FEBh	PLUSW0 ⁽³⁾	FCBh	PR2	FABh	RCSTA	F8Bh	LATC
FEAh	FSR0H	FCAh	T2CON	FAAh	EEADRH	F8Ah	LATB
FE9h	FSR0L	FC9h	SSPBUF	FA9h	EEADR	F89h	LATA
FE8h	WREG	FC8h	SSPADD	FA8h	EEDATA	F88h	PORTJ ⁽²⁾
FE7h	INDF1 ⁽³⁾	FC7h	SSPSTAT	FA7h	EECON2	F87h	PORTH ⁽²⁾
FE6h	POSTINC1 ⁽³⁾	FC6h	SSPCON1	FA6h	EECON1	F86h	PORTG
FE5h	POSTDEC1 ⁽³⁾	FC5h	SSPCON2	FA5h	IPR3	F85h	PORTF
FE4h	PREINC1 ⁽³⁾	FC4h	ADRESH	FA4h	PIR3	F84h	PORTE
FE3h	PLUSW1 ⁽³⁾	FC3h	ADRESL	FA3h	PIE3	F83h	PORTD
FE2h	FSR1H	FC2h	ADCON0	FA2h	IPR2	F82h	PORTC
FE1h	FSR1L	FC1h	ADCON1	FA1h	PIR2	F81h	PORTB
FE0h	BSR	FC0h	ADCON2	FA0h	PIE2	F80h	PORTA

Note 1: Unimplemented registers are read as '0'.

2: This register is not available on PIC18F6X8X devices.

3: This is not a physical register.

PIC18F6585/8585/6680/8680

TABLE 4-3: REGISTER FILE SUMMARY

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:
TOSU	—	—	—	Top-of-Stack Upper Byte (TOS<20:16>)					--0 0000	36, 54
TOSH	Top-of-Stack High Byte (TOS<15:8>)								0000 0000	36, 54
TOSL	Top-of-Stack Low Byte (TOS<7:0>)								0000 0000	36, 54
STKPTR	STKFUL	STKUNF	—	Return Stack Pointer					00-0 0000	36, 55
PCLATU	—	—	bit 21	Holding Register for PC<20:16>					--00 0000	36, 56
PCLATH	Holding Register for PC<15:8>								0000 0000	36, 56
PCL	PC Low Byte (PC<7:0>)								0000 0000	36, 56
TBLPTRU	—	—	bit 21 ⁽²⁾	Program Memory Table Pointer Upper Byte (TBLPTR<20:16>)					--00 0000	36, 86
TBLPTRH	Program Memory Table Pointer High Byte (TBLPTR<15:8>)								0000 0000	36, 86
TBLPTRL	Program Memory Table Pointer Low Byte (TBLPTR<7:0>)								0000 0000	36, 86
TABLAT	Program Memory Table Latch								0000 0000	36, 86
PRODH	Product Register High Byte								xxxx xxxx	36, 107
PRODL	Product Register Low Byte								xxxx xxxx	36, 107
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	0000 000x	36, 111
INTCON2	RBPV	INTEDG0	INTEDG1	INTEDG2	INTEDG3	TMR0IP	INT3IP	RBIP	1111 1111	36, 112
INTCON3	INT2IP	INT1IP	INT3IE	INT2IE	INT1IE	INT3IF	INT2IF	INT1IF	1100 0000	36, 113
INDF0	Uses contents of FSR0 to address data memory – value of FSR0 not changed (not a physical register)								n/a	79
POSTINC0	Uses contents of FSR0 to address data memory – value of FSR0 post-incremented (not a physical register)								n/a	79
POSTDEC0	Uses contents of FSR0 to address data memory – value of FSR0 post-decremented (not a physical register)								n/a	79
PREINC0	Uses contents of FSR0 to address data memory – value of FSR0 pre-incremented (not a physical register)								n/a	79
PLUSW0	Uses contents of FSR0 to address data memory – value of FSR0 pre-incremented (not a physical register) – value of FSR0 offset by value in WREG								n/a	79
FSR0H	—	—	—	—	Indirect Data Memory Address Pointer 0 High Byte				---- 0000	36, 79
FSR0L	Indirect Data Memory Address Pointer 0 Low Byte								xxxx xxxx	36, 79
WREG	Working Register								xxxx xxxx	36
INDF1	Uses contents of FSR1 to address data memory – value of FSR1 not changed (not a physical register)								n/a	79
POSTINC1	Uses contents of FSR1 to address data memory – value of FSR1 post-incremented (not a physical register)								n/a	79
POSTDEC1	Uses contents of FSR1 to address data memory – value of FSR1 post-decremented (not a physical register)								n/a	79
PREINC1	Uses contents of FSR1 to address data memory – value of FSR1 pre-incremented (not a physical register)								n/a	79
PLUSW1	Uses contents of FSR1 to address data memory – value of FSR1 pre-incremented (not a physical register) – value of FSR1 offset by value in WREG								n/a	79
FSR1H	—	—	—	—	Indirect Data Memory Address Pointer 1 High Byte				---- 0000	37, 79
FSR1L	Indirect Data Memory Address Pointer 1 Low Byte								xxxx xxxx	37, 79
BSR	—	—	—	—	Bank Select Register				---- 0000	37, 78
INDF2	Uses contents of FSR2 to address data memory – value of FSR2 not changed (not a physical register)								n/a	79
POSTINC2	Uses contents of FSR2 to address data memory – value of FSR2 post-incremented (not a physical register)								n/a	79
POSTDEC2	Uses contents of FSR2 to address data memory – value of FSR2 post-decremented (not a physical register)								n/a	79
PREINC2	Uses contents of FSR2 to address data memory – value of FSR2 pre-incremented (not a physical register)								n/a	79
PLUSW2	Uses contents of FSR2 to address data memory – value of FSR2 pre-incremented (not a physical register) – value of FSR2 offset by value in WREG								n/a	79
FSR2H	—	—	—	—	Indirect Data Memory Address Pointer 2 High Byte				---- 0000	37, 79
FSR2L	Indirect Data Memory Address Pointer 2 Low Byte								xxxx xxxx	37, 79

Legend: x = unknown, u = unchanged, – = unimplemented, q = value depends on condition

Note 1: RA6 and associated bits are configured as port pins in RCIO and ECIO Oscillator mode only and read '0' in all other oscillator modes.

2: Bit 21 of the TBLPTRU allows access to the device configuration bits.

3: These registers are unused on PIC18F6X80 devices; always maintain these clear.

4: These bits have multiple functions depending on the CAN module mode selection.

5: Meaning of this register depends on whether this buffer is configured as transmit or receive.

6: RG5 is available as an input when MCLR is disabled.

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

PIC18F6585/8585/6680/8680

5.4 Erasing Flash Program Memory

The minimum erase block is 32 words or 64 bytes. Only through the use of an external programmer or through ICSP control can larger blocks of program memory be bulk erased. Word erase in the Flash array is not supported.

When initiating an erase sequence from the microcontroller itself, a block of 64 bytes of program memory is erased. The Most Significant 16 bits of the TBLPTR<21:6> point to the block being erased. TBLPTR<5:0> are ignored.

The EECON1 register commands the erase operation. The EEPGD bit must be set to point to the Flash program memory. The WREN bit must be set to enable write operations. The FREE bit is set to select an erase operation.

For protection, the write initiate sequence for EECON2 must be used.

A long write is necessary for erasing the internal Flash. Instruction execution is halted while in a long write cycle. The long write will be terminated by the internal programming timer.

5.4.1 FLASH PROGRAM MEMORY ERASE SEQUENCE

The sequence of events for erasing a block of internal program memory location is:

1. Load table pointer with address of row being erased.
2. Set the EECON1 register for the erase operation:
 - set EEPGD bit to point to program memory;
 - clear the CFGS bit to access program memory;
 - set WREN bit to enable writes;
 - set FREE bit to enable the erase.
3. Disable interrupts.
4. Write 55h to EECON2.
5. Write 0AAh to EECON2.
6. Set the WR bit. This will begin the row erase cycle.
7. The CPU will stall for duration of the erase (about 2 ms using internal timer).
8. Execute a NOP.
9. Re-enable interrupts.

EXAMPLE 5-2: ERASING A FLASH PROGRAM MEMORY ROW

ERASE_ROW	MOVLW	upper(CODE_ADDR)	; load TBLPTR with the base
	MOVWF	TBLPTRU	; address of the memory block
	MOVLW	high(CODE_ADDR)	
	MOVWF	TBLPTRH	
	MOVLW	low(CODE_ADDR)	
	MOVWF	TBLPTRL	
Required Sequence	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	
	MOVWF	EECON2	; write 55h
	MOVLW	0AAh	
	MOVWF	EECON2	; write 0AAh
	BSF	EECON1, WR	; start erase (CPU stall)
	NOP		
	BSF	INTCON, GIE	; re-enable interrupts

PIC18F6585/8585/6680/8680

REGISTER 9-12: IPR3: PERIPHERAL INTERRUPT PRIORITY REGISTER 3

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
IRXIP	WAKIP	ERRIP	TXB2IP/ TXBnIP	TXB1IP ⁽¹⁾	TXB0IP ⁽¹⁾	RXB1IP/ RXBnIP	RXB0IP/ FIFOWMIP

bit 7

bit 0

bit 7 **IRXIP:** CAN Invalid Received Message Interrupt Priority bit

1 = High priority

0 = Low priority

bit 6 **WAKIP:** CAN bus Activity Wake-up Interrupt Priority bit

1 = High priority

0 = Low priority

bit 5 **ERRIP:** CAN bus Error Interrupt Priority bit

1 = High priority

0 = Low priority

bit 4 When CAN is in Mode 0:

TXB2IP: CAN Transmit Buffer 2 Interrupt Priority bit

1 = High priority

0 = Low priority

When CAN is in Mode 1 or 2:

TXBnIP: CAN Transmit Buffer Interrupt Priority bit

1 = High priority

0 = Low priority

bit 3 **TXB1IP:** CAN Transmit Buffer 1 Interrupt Priority bit⁽¹⁾

1 = High priority

0 = Low priority

bit 2 **TXB0IP:** CAN Transmit Buffer 0 Interrupt Priority bit⁽¹⁾

1 = High priority

0 = Low priority

bit 1 When CAN is in Mode 0:

RXB1IP: CAN Receive Buffer 1 Interrupt Priority bit

1 = High priority

0 = Low priority

When CAN is in Mode 1 or 2:

RXBnIP: CAN Receive Buffer Interrupts Priority bit

1 = High priority

0 = Low priority

bit 0 When CAN is in Mode 0:

RXB0IP: CAN Receive Buffer 0 Interrupt Priority bit

1 = High priority

0 = Low priority

When CAN is in Mode 1:

Unimplemented: Read as '0'

When CAN is in Mode 2:

FIFOWMIP: FIFO Watermark Interrupt Priority bit

1 = High priority

0 = Low priority

Note 1: In CAN Mode 1 and 2, this bit is forced to '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

10.4 PORTD, TRISD and LATD Registers

PORTD is an 8-bit wide, bidirectional port. The corresponding data direction register is TRISD. Setting a TRISD bit (= 1) will make the corresponding PORTD pin an input (i.e., put the corresponding output driver in a high-impedance mode). Clearing a TRISD bit (= 0) will make the corresponding PORTD pin an output (i.e., put the contents of the output latch on the selected pin).

The Data Latch register (LATD) is also memory mapped. Read-modify-write operations on the LATD register read and write the latched output value for PORTD.

PORTD is an 8-bit port with Schmitt Trigger input buffers. Each pin is individually configurable as an input or output.

Note: On a Power-on Reset, these pins are configured as digital inputs.

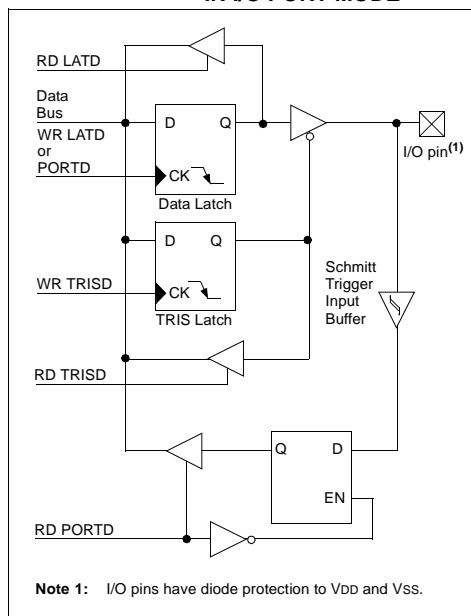
On PIC18F8X8X devices, PORTD is multiplexed with the system bus as the external memory interface; I/O port functions are only available when the system bus is disabled by setting the EBDIS bit in the MEMCOM register (MEMCON<7>). When operating as the external memory interface, PORTD is the low-order byte of the multiplexed address/data bus (AD7:AD0).

PORTD can also be configured as an 8-bit wide micro-processor port (Parallel Slave Port) by setting control bit, PSPMODE (TRISE<4>). In this mode, the input buffers are TTL. See **Section 10.10 “Parallel Slave Port (PSP)”** for additional information.

EXAMPLE 10-4: INITIALIZING PORTD

```
CLRF  PORTD    ; Initialize PORTD by
                ; clearing output
                ; data latches
CLRF  LATD      ; Alternate method
                ; to clear output
                ; data latches
MOVLW 0CFh     ; Value used to
                ; initialize data
                ; direction
MOVWF TRISD    ; Set RD<3:0> as inputs
                ; RD<5:4> as outputs
                ; RD<7:6> as inputs
```

FIGURE 10-9: PORTD BLOCK DIAGRAM IN I/O PORT MODE



PIC18F6585/8585/6680/8680

FIGURE 10-30: PARALLEL SLAVE PORT READ WAVEFORMS

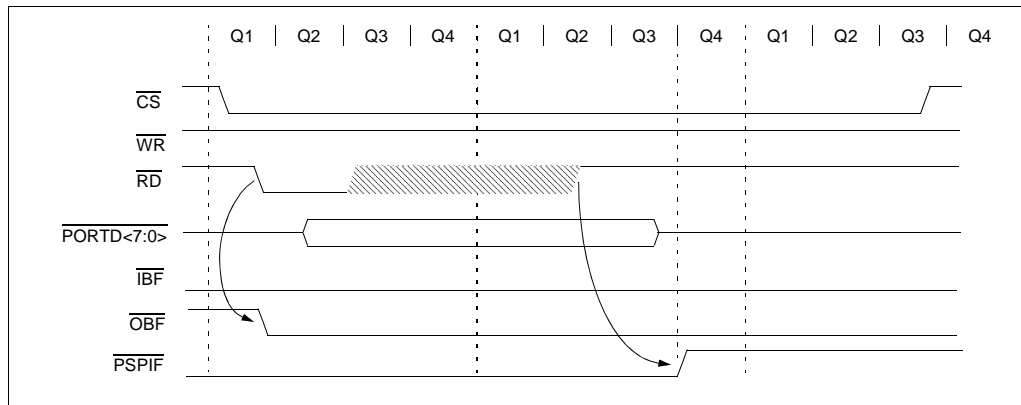


TABLE 10-19: REGISTERS ASSOCIATED WITH PARALLEL SLAVE PORT

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
PORTD	Port Data Latch when Written; Port pins when Read								xxxx xxxx	uuuu uuuu
LATD	LATD Data Output bits								xxxx xxxx	uuuu uuuu
TRISD	PORTD Data Direction bits								1111 1111	1111 1111
PORTE	RE7/CCP2/ AD15	RE6/AD14/ P1B	RE5/AD13/ P1C	RE4/ AD12	RE3/ AD11	RE2/ $\overline{CS}^{(1)}$ / AD10	RE1/ $\overline{WR}^{(1)}$ / AD9	RE0/ $\overline{RD}^{(1)}$ / AD8	xxxx xxxx	uuuu uuuu
LATE	LATE Data Output bits								xxxx xxxx	uuuu uuuu
TRISE	PORTE Data Direction bits								1111 1111	1111 1111
PSPCON	IBF	OBF	IBOV	PSPMODE	—	—	—	—	0000 ----	0000 ----
INTCON	GIE/ GIEH	PEIE/ GIEL	TMR0IF	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	0000 0000	0000 0000
PIR1	PSPIF ⁽¹⁾	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
PIE1	PSPIE ⁽¹⁾	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
IPR1	PSPIP ⁽¹⁾	ADIP	RCIP	TXIP	SSPIP	CCP1IP	TMR2IP	TMR1IP	1111 1111	1111 1111

Legend: x = unknown, u = unchanged, — = unimplemented, read as '0'. Shaded cells are not used by the Parallel Slave Port.

Note 1: Enabled only in Microcontroller mode.

PIC18F6585/8585/6680/8680

REGISTER 15-2: CCP2CON REGISTER

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	DC2B1	DC2B0	CCP2M3	CCP2M2	CCP2M1	CCP2M0
bit 7		bit 0					

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

bit 5-4 **DC2B1:DC2B0:** PWM Duty Cycle bit 1 and bit 0

Capture mode:

Unused.

Compare mode:

Unused.

PWM mode:

These bits are the two LSbs of the 10-bit PWM duty cycle. The eight MSbs of the duty cycle are found in CCPR2L.

bit 3-0 **CCP2M3:CCP2M0:** CCP2 Mode Select bits

0000 = Capture/Compare/PWM off (resets CCP2 module)

0001 = Reserved

0010 = Compare mode, toggle output on match

0011 = Reserved

0100 = Capture mode, every falling edge

0101 = Capture mode, every rising edge

0110 = Capture mode, every 4th rising edge

0111 = Capture mode, every 16th rising edge

1000 = Compare mode, initialize CCP pin low, on compare match force CCP pin high

1001 = Compare mode, initialize CCP pin high, on compare match force CCP pin low

1010 = Compare mode, generate software interrupt only, CCP pin is unaffected

1011 = Compare mode, trigger special event, resets TMR1 or TMR3 and starts A/D conversion if A/D module is enabled

11xx = PWM mode

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

PIC18F6585/8585/6680/8680

TABLE 18-3: BAUD RATES FOR ASYNCHRONOUS MODES

BAUD RATE (K)	SYNC = 0, BRGH = 0, BRG16 = 0											
	Fosc = 40.000 MHz			Fosc = 20.000 MHz			Fosc = 10.000 MHz			Fosc = 8.000 MHz		
	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)
0.3	—	—	—	—	—	—	—	—	—	—	—	—
1.2	—	—	—	1.221	1.73	255	1.202	0.16	129	1201	-0.16	103
2.4	2.441	1.73	255	2.404	0.16	129	2.404	0.16	64	2403	-0.16	51
9.6	9.615	0.16	64	9.766	1.73	31	9.766	1.73	15	9615	-0.16	12
19.2	19.531	1.73	31	19.531	1.73	15	19.531	1.73	7	—	—	—
57.6	56.818	-1.36	10	62.500	8.51	4	52.083	-9.58	2	—	—	—
115.2	125.000	8.51	4	104.167	-9.58	2	78.125	-32.18	1	—	—	—

BAUD RATE (K)	SYNC = 0, BRGH = 0, BRG16 = 0								
	Fosc = 4.000 MHz			Fosc = 2.000 MHz			Fosc = 1.000 MHz		
	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)
0.3	0.300	0.16	207	300	-0.16	103	300	-0.16	51
1.2	1.202	0.16	51	1201	-0.16	25	1201	-0.16	12
2.4	2.404	0.16	25	2403	-0.16	12	—	—	—
9.6	8.929	-6.99	6	—	—	—	—	—	—
19.2	20.833	8.51	2	—	—	—	—	—	—
57.6	62.500	8.51	0	—	—	—	—	—	—
115.2	62.500	-45.75	0	—	—	—	—	—	—

BAUD RATE (K)	SYNC = 0, BRGH = 1, BRG16 = 0											
	Fosc = 40.000 MHz			Fosc = 20.000 MHz			Fosc = 10.000 MHz			Fosc = 8.000 MHz		
	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)
0.3	—	—	—	—	—	—	—	—	—	—	—	—
1.2	—	—	—	—	—	—	—	—	—	—	—	—
2.4	—	—	—	—	—	—	2.441	1.73	255	2403	-0.16	207
9.6	9.766	1.73	255	9.615	0.16	129	9.615	0.16	64	9615	-0.16	51
19.2	19.231	0.16	129	19.231	0.16	64	19.531	1.73	31	19230	-0.16	25
57.6	58.140	0.94	42	56.818	-1.36	21	56.818	-1.36	10	55555	3.55	8
115.2	113.636	-1.36	21	113.636	-1.36	10	125.000	8.51	4	—	—	—

BAUD RATE (K)	SYNC = 0, BRGH = 1, BRG16 = 0								
	Fosc = 4.000 MHz			Fosc = 2.000 MHz			Fosc = 1.000 MHz		
	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)
0.3	—	—	—	—	—	—	300	-0.16	207
1.2	1.202	0.16	207	1201	-0.16	103	1201	-0.16	51
2.4	2.404	0.16	103	2403	-0.16	51	2403	-0.16	25
9.6	9.615	0.16	25	9615	-0.16	12	—	—	—
19.2	19.231	0.16	12	—	—	—	—	—	—
57.6	62.500	8.51	3	—	—	—	—	—	—
115.2	125.000	8.51	1	—	—	—	—	—	—

PIC18F6585/8585/6680/8680

18.1.2 AUTO-BAUD RATE DETECT

The enhanced USART module supports the automatic detection and calibration of baud rate. This feature is active only in Asynchronous mode and while the WUE bit is clear.

The automatic baud rate measurement sequence (Figure 18-1) begins whenever a Start bit is received and the ABDEN bit is set. The calculation is self-averaging.

In the Auto-Baud Rate Detect (ABD) mode, the clock to the BRG is reversed. Rather than the BRG clocking the incoming RX signal, the RX signal is timing the BRG. In ABD mode, the internal Baud Rate Generator is used as a counter to time the bit period of the incoming serial byte stream.

Once the ABDEN bit is set, the state machine will clear the BRG and look for a Start bit. The auto-baud detect must receive a byte with the value 55h (ASCII "U", which is also the LIN bus sync character) in order to calculate the proper bit rate. The measurement is taken over both a low and a high bit time in order to minimize any effects caused by asymmetry of the incoming signal. After a Start bit, the SPBRG begins counting up using the preselected clock source on the first rising edge of RX. After eight bits on the RX pin or the fifth rising edge, an accumulated value totalling the proper BRG period is left in the SPBRG:SPBRGH registers. Once the 5th edge is seen (should correspond to the Stop bit), the ABDEN bit is automatically cleared.

While calibrating the baud rate period, the BRG registers are clocked at 1/8th the preconfigured clock rate. Note that the BRG clock will be configured by the BRG16 and BRGH bits. Independent of the BRG16 bit setting, both the SPBRG and SPBRGH will be used as a 16-bit counter. This allows the user to verify that no

carry occurred for 8-bit modes by checking for 00h in the SPBRGH register. Refer to Table 18-4 for counter clock rates to the BRG.

While the ABD sequence takes place, the USART state machine is held in Idle. The RCIF interrupt is set once the fifth rising edge on RX is detected. The value in the RCREG needs to be read to clear the RCIF interrupt. RCREG content should be discarded.

Note 1: If the WUE bit is set with the ABDEN bit, auto-baud rate detection will occur on the byte *following* the break character.

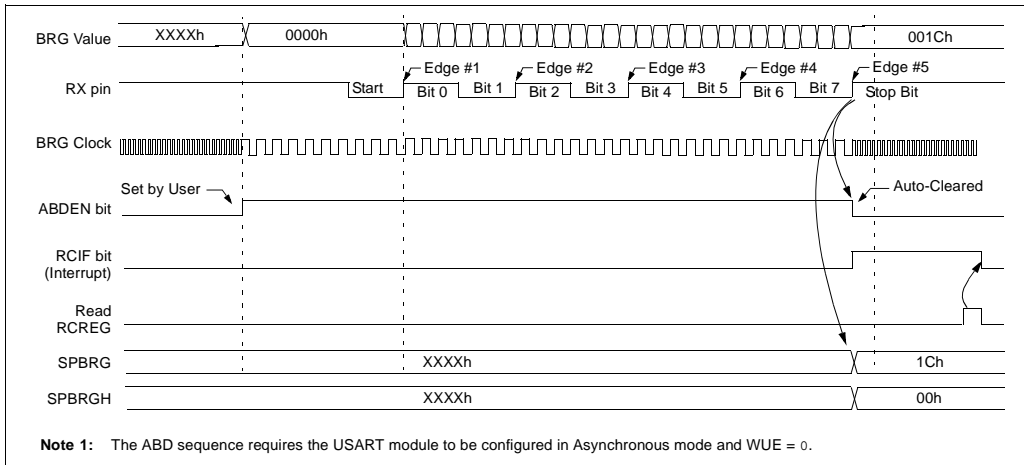
2: It is up to the user to determine that the incoming character baud rate is within the range of the selected BRG clock source. Some combinations of oscillator frequency and USART baud rates are not possible due to bit error rates. Overall system timing and communication baud rates must be taken into consideration when using the auto-baud rate detection feature.

TABLE 18-4: BRG COUNTER CLOCK RATES

BRG16	BRGH	BRG Counter Clock
0	0	Fosc/512
0	1	Fosc/128
1	0	Fosc/128
1	1	Fosc/32

Note: During the ABD sequence, SPBRG and SPBRGH are both used as a 16-bit counter independent of BRG16 setting.

FIGURE 18-1: AUTOMATIC BAUD RATE CALCULATION



18.2.5 BREAK CHARACTER SEQUENCE

The enhanced USART module has the capability of sending the special break character sequences that are required by the LIN bus standard. The break character transmit consists of a Start bit, followed by twelve '0' bits and a Stop bit. The frame break character is sent whenever the SENDB and TXEN bits (TXSTA<3> and TXSTA<5>) are set while the Transmit Shift register is loaded with data. Note that the value of data written to TXREG will be ignored and all '0's will be transmitted.

The SENDB bit is automatically reset by hardware after the corresponding Stop bit is sent. This allows the user to preload the transmit FIFO with the next transmit byte following the break character (typically, the sync character in the LIN specification).

Note that the data value written to the TXREG for the break character is ignored. The write simply serves the purpose of initiating the proper sequence.

The TRMT bit indicates when the transmit operation is active or Idle, just as it does during normal transmission. See Figure 18-9 for the timing of the break character sequence.

18.2.5.1 Break and Sync Transmit Sequence

The following sequence will send a message frame header made up of a break, followed by an auto-baud sync byte. This sequence is typical of a LIN bus master.

1. Configure the USART for the desired mode.
2. Set the TXEN and SENDB bits to set up the break character.
3. Load the TXREG with a dummy character to initiate transmission (the value is ignored).
4. Write '55h' to TXREG to load the sync character into the transmit FIFO buffer.
5. After the break has been sent, the SENDB bit is reset by hardware. The sync character now transmits in the preconfigured mode.

When the TXREG becomes empty, as indicated by the TXIF, the next data byte can be written to TXREG.

18.2.6 RECEIVING A BREAK CHARACTER

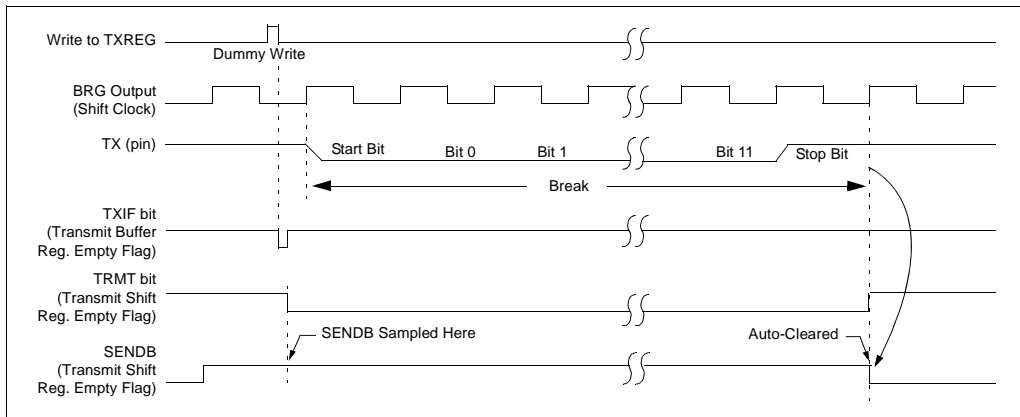
The enhanced USART module can receive a break character in two ways.

The first method forces the configuration of the baud rate at a frequency of 9/13 the typical speed. This allows for the Stop bit transition to be at the correct sampling location (13 bits for break versus Start bit and 8 data bits for typical data).

The second method uses the auto-wake-up feature described in **Section 18.2.4 "Auto-Wake-up on Sync Break Character"**. By enabling this feature, the USART will sample the next two transitions on RX/DT, cause an RCIF interrupt, and receive the next data byte followed by another interrupt.

Note that following a break character, the user will typically want to enable the auto-baud rate detect feature. For both methods, the user can set the ABD bit once the TXIF interrupt is observed.

FIGURE 18-9: SEND BREAK CHARACTER SEQUENCE



PIC18F6585/8585/6680/8680

FIGURE 18-11: SYNCHRONOUS TRANSMISSION (THROUGH TXEN)

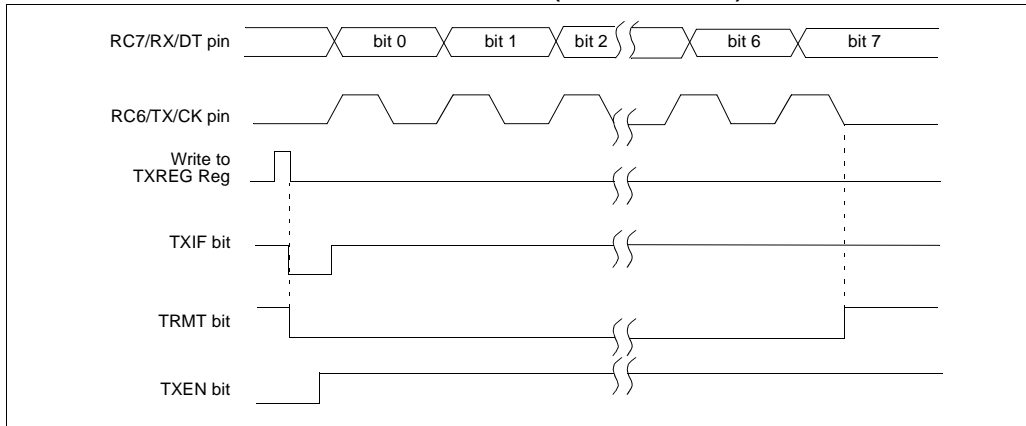


TABLE 18-7: REGISTERS ASSOCIATED WITH SYNCHRONOUS MASTER TRANSMISSION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	0000 0000	0000 0000
PIR1	PSPIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
PIE1	PSPIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
IPR1	PSPIP	ADIP	RCIP	TXIP	SSPIP	CCP1IP	TMR2IP	TMR1IP	1111 1111	1111 1111
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
TXREG	USART Transmit Register								0000 0000	0000 0000
TXSTA	CSRC	TX9	TXEN	SYNC	SEnDB	BRGH	TRMT	TX9D	0000 0010	0000 0010
BAUDCON	—	RCIDL	—	SCKP	BRG16	—	WUE	ABDEN	-1-0 0-00	-1-0 0-00
SPBRGH	Baud Rate Generator Register, High Byte								0000 0000	0000 0000
SPBRG	Baud Rate Generator Register, Low Byte								0000 0000	0000 0000

Legend: x = unknown, - = unimplemented, read as '0'. Shaded cells are not used for synchronous master transmission.

22.1 Control Register

The Low-Voltage Detect Control register controls the operation of the Low-Voltage Detect circuitry.

REGISTER 22-1: LVDCON REGISTER

U-0	U-0	R-0	R/W-0	R/W-0	R/W-1	R/W-0	R/W-1
—	—	IRVST	LV DEN	LV DL3	LV DL2	LV DL1	LV DL0
bit 7							bit 0

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

bit 5 **IRVST:** Internal Reference Voltage Stable Flag bit

1 = Indicates that the Low-Voltage Detect logic will generate the interrupt flag at the specified voltage range

0 = Indicates that the Low-Voltage Detect logic will not generate the interrupt flag at the specified voltage range and the LVD interrupt should not be enabled

bit 4 **LV DEN:** Low-Voltage Detect Power Enable bit

1 = Enables LVD, powers up LVD circuit

0 = Disables LVD, powers down LVD circuit

bit 3-0 **LV DL3:LV DL0:** Low-Voltage Detection Limit bits

1111 = External analog input is used (input comes from the LVDIN pin)

1110 = 4.5V-4.77V

1101 = 4.2V-4.45V

1100 = 4.0V-4.24V

1011 = 3.8V-4.03V

1010 = 3.6V-3.82V

1001 = 3.5V-3.71V

1000 = 3.3V-3.50V

0111 = 3.0V-3.18V

0110 = 2.8V-2.97V

0101 = 2.7V-2.86V

0100 = 2.5V-2.65V

0011 = 2.4V-2.54V

0010 = 2.2V-2.33V

0001 = 2.0V-2.12V

0000 = Reserved

Note: LV DL3:LV DL0 modes which result in a trip point below the valid operating voltage of the device are not tested.

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

PIC18F6585/8585/6680/8680

REGISTER 23-6: TXBnSIDH: TRANSMIT BUFFER n STANDARD IDENTIFIER REGISTERS, HIGH BYTE [$0 \leq n \leq 2$]

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3

bit 7 bit 0

bit 7-0 **SID10:SID3:** Standard Identifier bits, if EXIDE (TXBnSIDL<3>) = 0;
Extended Identifier bits EID28:EID21, if EXIDE = 1.

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

REGISTER 23-7: TXBnSIDL: TRANSMIT BUFFER n STANDARD IDENTIFIER REGISTERS, LOW BYTE [$0 \leq n \leq 2$]

R/W-x	R/W-x	R/W-x	U-0	R/W-x	U-0	R/W-x	R/W-x
SID2	SID1	SID0	—	EXIDE	—	EID17	EID16

bit 7 bit 0

bit 7-5 **SID2:SID0:** Standard Identifier bits, if EXIDE (TXBnSIDL<3>) = 0;
Extended Identifier bits EID20:EID18, if EXIDE = 1.

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

bit 3 **EXIDE:** Extended Identifier Enable bit

1 = Message will transmit extended ID, SID10:SID0 becomes EID28:EID18
0 = Message will transmit standard ID, EID17:EID0 are ignored

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

bit 1-0 **EID17:EID16:** Extended Identifier bits

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

REGISTER 23-8: TXBnEIDH: TRANSMIT BUFFER n EXTENDED IDENTIFIER REGISTERS, HIGH BYTE [$0 \leq n \leq 2$]

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8

bit 7 bit 0

bit 7-0 **EID15:EID8:** Extended Identifier bits (not used when transmitting standard identifier message)

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

PIC18F6585/8585/6680/8680

REGISTER 23-32: BnDm: TX/RX BUFFER n DATA FIELD BYTE m REGISTERS IN RECEIVE MODE [$0 \leq n \leq 5$, $0 \leq m \leq 7$, TXnEN (BSEL<n>) = 0]⁽¹⁾

R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
BnDm7	BnDm6	BnDm5	BnDm4	BnDm3	BnDm2	BnDm1	BnDm0
bit 7				bit 0			

bit 7-0

BnDm7:BnDm0: Receive Buffer n Data Field Byte m bits (where $0 \leq n < 3$ and $0 < m < 8$)
Each receive buffer has an array of registers. For example, Receive Buffer 0 has 7 registers: B0D0 to B0D7.

Note 1: These registers are available in Mode 1 and 2 only.

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

REGISTER 23-33: BnDm: TX/RX BUFFER n DATA FIELD BYTE m REGISTERS IN TRANSMIT MODE [$0 \leq n \leq 5$, $0 \leq m \leq 7$, TXnEN (BSEL<n>) = 1]⁽¹⁾

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
BnDm7	BnDm6	BnDm5	BnDm4	BnDm3	BnDm2	BnDm1	BnDm0
bit 7				bit 0			

bit 7-0

BnDm7:BnDm0: Transmit Buffer n Data Field Byte m bits (where $0 \leq n < 3$ and $0 < m < 8$)
Each transmit buffer has an array of registers. For example, Transmit Buffer 0 has 7 registers: TXB0D0 to TXB0D7.

Note 1: These registers are available in Mode 1 and 2 only.

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

PIC18F6585/8585/6680/8680

SUBWFB Subtract W from f with Borrow

Syntax: `[label] SUBWFB f[,d[,a]]`

Operands: $0 \leq f \leq 255$

$d \in [0,1]$

$a \in [0,1]$

Operation: $(f) - (W) - (\overline{C}) \rightarrow \text{dest}$

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

Encoding:

0101	10da	ffff	ffff
------	------	------	------

Description: Subtract W and the Carry flag (borrow) from register 'f' (2's complement method). If 'd' is '0', the result is stored in W. If 'd' is '1', the result is stored back in register 'f' (default). If 'a' is '0', the Access Bank will be selected, overriding the BSR value. If 'a' is '1', then the bank will be selected as per the BSR value (default).

Words: 1

Cycles: 1

Q Cycle Activity:

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

Example 1: SUBWFB REG, 1, 0

Before Instruction

REG = 0x19 (0001 1001)
W = 0x0D (0000 1101)
C = 1

After Instruction

REG = 0x0C (0000 1011)
W = 0x0D (0000 1101)
C = 1
Z = 0
N = 0 ; result is positive

Example 2: SUBWFB REG, 0, 0

Before Instruction

REG = 0x1B (0001 1011)
W = 0x1A (0001 1010)
C = 0

After Instruction

REG = 0x1B (0001 1011)
W = 0x00 (0000 1101)
C = 1
Z = 1 ; result is zero
N = 0

Example 3: SUBWFB REG, 1, 0

Before Instruction

REG = 0x03 (0000 0011)
W = 0x0E (0000 1101)
C = 1

After Instruction

REG = 0xF5 (1111 0100)
; [2's comp]
W = 0x0E (0000 1101)
C = 0
Z = 0
N = 1 ; result is negative

SWAPF Swap f

Syntax: `[label] SWAPF f[,d[,a]]`

Operands: $0 \leq f \leq 255$

$d \in [0,1]$

$a \in [0,1]$

Operation: $(f<3:0>) \rightarrow \text{dest}<7:4>$,
 $(f<7:4>) \rightarrow \text{dest}<3:0>$

Status Affected: None

Encoding:

0011	10da	ffff	ffff
------	------	------	------

Description: The upper and lower nibbles of register 'f' are exchanged. If 'd' is '0', the result is placed in W. If 'd' is '1', the result is placed in register 'f' (default). If 'a' is '0', the Access Bank will be selected, overriding the BSR value. If 'a' is '1', then the bank will be selected as per the BSR value (default).

Words: 1

Cycles: 1

Q Cycle Activity:

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

Example: SWAPF REG, 1, 0

Before Instruction

REG = 0x53

After Instruction

REG = 0x35

PIC18F6585/8585/6680/8680

TABLE 27-4: MEMORY PROGRAMMING REQUIREMENTS

DC Characteristics			Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for extended				
Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
Internal Program Memory Programming Specifications (Note 1)							
D110	VPP	Voltage on $\overline{\text{MCLR}}$ /VPP pin	9.00	—	13.25	V	(Note 2)
D112	IPP	Current into $\overline{\text{MCLR}}$ /VPP pin	—	—	5	μA	
D113	IDDP	Supply Current during Programming	—	—	10	mA	
Data EEPROM Memory							
D120	ED	Cell Endurance	100K	1M	—	E/W	-40°C to $+85^{\circ}\text{C}$
D120A	ED	Cell Endurance	10K	100K	—	E/W	$+85^{\circ}\text{C}$ to $+125^{\circ}\text{C}$
D121	VDRW	VDD for Read/Write	VMIN	—	5.5	V	Using EECON to read/write, VMIN = Minimum operating voltage
D122	TDEW	Erase/Write Cycle Time	—	4	—	ms	
D123	TRETD	Characteristic Retention	40	—	—	Year	-40°C to $+85^{\circ}\text{C}$ (Note 3)
D123A	TRETD	Characteristic Retention	100	—	—	Year	25°C (Note 3)
Program Flash Memory							
D130	EP	Cell Endurance	10K	100K	—	E/W	-40°C to $+85^{\circ}\text{C}$
D130A	EP	Cell Endurance	1000	10K	—	E/W	$+85^{\circ}\text{C}$ to $+125^{\circ}\text{C}$
D131	VPR	VDD for Read	VMIN	—	5.5	V	VMIN = Minimum operating voltage
D132	VIE	VDD for Block Erase	4.5	—	5.5	V	Using ICSP port
D132A	VIW	VDD for Externally Timed Erase or Write	4.5	—	5.5	V	Using ICSP port
D132B	VPEW	VDD for Self-timed Write	VMIN	—	5.5	V	VMIN = Minimum operating voltage
D133	TIE	ICSP Block Erase Cycle Time	—	5	—	ms	VDD > 4.5V
D133A	TIW	ICSP Erase or Write Cycle Time (externally timed)	1	—	—	ms	VDD > 4.5V
D133A	TIW	Self-timed Write Cycle Time	—	2.5	—	ms	
D134	TRETD	Characteristic Retention	40	—	—	Year	-40°C to $+85^{\circ}\text{C}$ (Note 3)
D134A	TRETD	Characteristic Retention	100	—	—	Year	25°C (Note 3)

† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: These specifications are for programming the on-chip program memory through the use of table write instructions.

2: The pin may be kept in this range at times other than programming but it is not recommended.

3: Retention time is valid provided no other specifications are violated.

PIC18F6585/8585/6680/8680

Timer2	162	Example SPI Slave Mode	
Associated Registers	163	(CKE = 0)	439
Operation	162	Example SPI Slave Mode	
Postscaler. See Postscaler, Timer2.		(CKE = 1)	440
PR2 Register	162, 173, 177	External Clock (All Modes	
Prescaler. See Prescaler, Timer2.		except PLL)	428
SSP Clock Shift	162, 163	External Program Memory Bus	
TMR2 Register	162	(16-bit Mode)	99
TMR2 to PR2 Match		First Start Bit	217
Interrupt	162, 163, 173, 177	Full-Bridge PWM Output	181
Timer3	164	Half-Bridge PWM Output	180
Associated Registers	166	I ² C Bus Data	441
Operation	165	I ² C Bus Start/Stop Bits	441
Oscillator	164, 166	I ² C Master Mode (7 or	
Overflow Interrupt	164, 166	10-bit Transmission)	220
Special Event Trigger		I ² C Master Mode	
(CCP)	166	(7-bit Reception)	221
TMR3H Register	164	I ² C Slave Mode (10-bit Reception,	
TMR3L Register	164	SEN = 0)	206
Timing Diagrams		I ² C Slave Mode (10-bit Reception,	
A/D Conversion	447	SEN = 1)	211
Acknowledge Sequence	222	I ² C Slave Mode	
Asynchronous Reception	241	(10-bit Transmission)	207
Asynchronous Transmission	238	I ² C Slave Mode (7-bit Reception,	
Asynchronous Transmission		SEN = 0)	204
(Back to Back)	238	I ² C Slave Mode (7-bit Reception,	
Automatic Baud Rate		SEN = 1)	210
Calculation	236	I ² C Slave Mode	
Auto-Wake-up Bit (WUE) During		(7-bit Transmission)	205
Normal Operation	242	Low-Voltage Detect	272
Auto-Wake-up Bit (WUE)		Master SSP I ² C Bus Data	443
During Sleep	242	Master SSP I ² C Bus	
Baud Rate Generator with		Start/Stop Bits	443
Clock Arbitration	216	Parallel Slave Port	
BRG Reset Due to SDA Arbitration During		(PIC18FXX8X)	436
Start Condition	225	Parallel Slave Port (PSP)	
Brown-out Reset (BOR)	433	Read	154
Bus Collision During a Repeated		Parallel Slave Port (PSP)	
Start Condition (Case 1)	226	Write	153
Bus Collision During a Repeated		Program Memory Read	430
Start Condition (Case 2)	226	Program Memory Write	431
Bus Collision During a Stop Condition		PWM Auto-Shutdown (PRSEN = 0,	
(Case 1)	227	Auto-Restart Disabled)	186
Bus Collision During a Stop Condition		PWM Auto-Shutdown (PRSEN = 1,	
(Case 2)	227	Auto-Restart Enabled)	186
Bus Collision During Start Condition		PWM Output	173
(SCL = 0)	225	Repeat Start Condition	218
Bus Collision During Start Condition		Reset, Watchdog Timer (WDT),	
(SDA only)	224	Oscillator Start-up Timer (OST)	
Bus Collision for Transmit and		and Power-up Timer (PWRT)	432
Acknowledge	223	Send Break Character Sequence	243
Capture/Compare/PWM		Slave Mode General Call Address	
(All CCP Modules)	435	Sequence (7 or 10-bit	
CLKO and I/O	429	Address Mode)	212
Clock Synchronization	209	Slave Synchronization	195
Clock/Instruction Cycle	56	Slow Rise Time (MCLR Tied to V _{DD}	
Example SPI Master Mode		via 1 k Ω Resistor)	50
(CKE = 0)	437	SPI Mode (Master Mode)	194
Example SPI Master Mode		SPI Mode (Slave Mode with	
(CKE = 1)	438	CKE = 0)	196