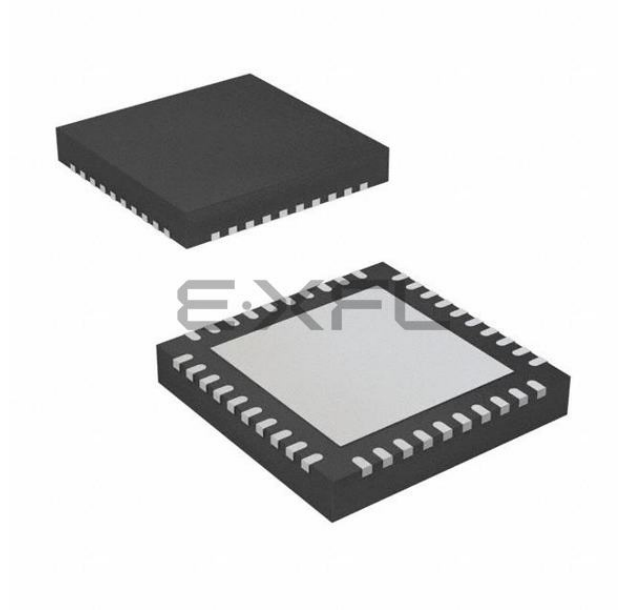


Welcome to E-XFL.COM

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	32MHz
Connectivity	I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LCD, POR, PWM, WDT
Number of I/O	35
Program Memory Size	14KB (8K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	2.3V ~ 5.5V
Data Converters	A/D 31x12b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	40-UFQFN Exposed Pad
Supplier Device Package	40-UQFN (5x5)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f19175-i-mv

TABLE 1-4: PIC16(L)F19185/86 PINOUT DESCRIPTION (CONTINUED)

Name	Function	Input Types	Output Types	Description
RC6/CK1(3)/TX1(1)/IOCC6/ANC6/SEG22/VLCD2	RC6	TTL/ST	CMOS/OD	General purpose I/O.
	CK1(3)	—	—	EUSART synchronous clock out
	TX1(1)	—	—	EUSART asynchronous TX data out
	IOCC6	TTL/ST	—	Interrupt-on-change input.
	ANC6	AN	—	ADC Channel input.
	SEG22	—	AN	LCD Analog output.
	VLCD2	AN	—	LCD analog input
RC7/DT1(3)/RX1(1)/IOCC7/ANC7/SEG23/VLCD1	RC7	TTL/ST	CMOS/OD	General purpose I/O.
	DT1(3)	—	—	EUSART synchronous data output
	RX1(1)	—	—	EUSART receive input.
	IOCC7	TTL/ST	—	Interrupt-on-change input.
	ANC7	AN	—	ADC Channel input.
	SEG23	—	AN	LCD Analog output.
	VLCD1	AN	—	LCD analog input
RD0/AND0/SEG24	RD0	TTL/ST	CMOS/OD	General purpose I/O.
	AND0	AN	—	ADC Channel input.
	SEG24	AN	—	LCD Analog output.
RD1/AND1/SEG25	RD1	TTL/ST	CMOS/OD	General purpose I/O.
	AND1	AN	—	ADC Channel input.
	SEG25	—	AN	LCD Analog output.
RD2/AND2/COM5	RD2	TTL/ST	CMOS/OD	General purpose I/O.
	AND2	AN	—	ADC Channel input.
	SEG26	—	AN	LCD Analog output.
	COM5	—	AN	LCD Driver Common Outputs.
RD3/AND3/SEG27/COM4	RD3	TTL/ST	CMOS/OD	General purpose I/O.
	AND3	AN	—	ADC Channel input.
	SEG27	—	AN	LCD Analog output.
	COM4	—	AN	LCD Driver Common Outputs.
RD4/AND4/SEG28	RD4	TTL/ST	CMOS/OD	General purpose I/O.
	AND4	AN	—	ADC Channel input.
	SEG28	—	AN	LCD Analog output.
RD5/AND5/SEG29	RD5	TTL/ST	CMOS/OD	General purpose I/O.
	AND5	AN	—	ADC Channel input.
	SEG29	AN	—	LCD Analog output.
RD6/AND6/SEG30	RD6	TTL/ST	CMOS/OD	General purpose I/O.
	AND6	AN	—	ADC Channel input.
	SEG30	—	AN	LCD Analog output.

Legend: AN = Analog input or output; CMOS = CMOS compatible input or output; OD = Open-Drain; TTL = TTL compatible input'; ST = Schmitt Trigger input with CMOS levels; I^2C = Schmitt Trigger input with I^2C; HV = High Voltage; XTAL = Crystal levels

Note 1: This is a PPS remappable input signal. The input function may be moved from the default location shown to one of several other PORTx pins. Refer to Table 14-2 for details on which PORT pins may be used for this signal.

2: All output signals shown in this row are PPS remappable. These signals may be mapped to output onto one of several PORTx pin options as described in Table 14-3.

3: This is a bidirectional signal. For normal module operation, the firmware should map this signal to the same pin in both the PPS input and PPS output registers.

4: These pins are configured for I^2C logic levels. The SCLx/SDAx signals may be assigned to any of the RB1/RB2/RC3/RC4 pins. PPS assignments to the other pins (e.g., RA5) will operate, but input logic levels will be standard TTL/ST, as selected by the INLVL register, instead of the I^2C specific or SMBus input buffer thresholds.

TABLE 4-10: PIC16(L)F19155/56/75/76/85/86 MEMORY MAP, BANKS 58-63

	Bank 58		Bank 59		Bank 60		Bank 61		Bank 62		Bank 63
1D00h	Core Registers (Table 4-3)	1D80h	Core Registers (Table 4-3)	1E00h	Core Registers (Table 4-3)	1E80h	Core Registers (Table 4-3)	1F00h	Core Registers (Table 4-3)	1F80h	Core Registers (Table 4-3)
1D0Ch	LCDCON	1D8Ch	—	1E0Ch	—	1E8Ch	—	1F0Ch	—	1F8Ch	—
1D0Dh	LCDPS	1D8Dh	—	1E0Dh	—	1E8Dh	—	1F0Dh	—	1F8Dh	—
1D0Eh	LCDSE0	1D8Eh	—	1E0Eh	—	1E8Eh	—	1F0Eh	—	1F8Eh	—
1D0Fh	LCDSE1	1D8Fh	—	1E0Fh	CLCDATA	1E8Fh	PPSLOCK	1F0Fh	—	1F8Fh	—
1D10h	LCDSE2	1D90h	—	1E10h	CLC1CON	1E90h	INTPPS	1F10h	RA0PPS	1F90h	—
1D11h	LCDSE3	1D91h	—	1E11h	CLC1POL	1E91h	T0CKIPPS	1F11h	RA1PPS	1F91h	—
1D12h	LCDSE4	1D92h	—	1E12h	CLC1SEL0	1E92h	T1CKIPPS	1F12h	RA2PPS	1F92h	—
1D13h	LCDSE5	1D93h	—	1E13h	CLC1SEL1	1E93h	T1GPPS	1F13h	RA3PPS	1F93h	—
1D14h	LCDVCON1	1D94h	—	1E14h	CLC1SEL2	1E94h	—	1F14h	RA4PPS	1F94h	—
1D15h	LCDVCON2	1D95h	—	1E15h	CLC1SEL3	1E95h	—	1F15h	RA5PPS	1F95h	—
1D16h	LCDREF	1D96h	—	1E16h	CLC1GLS0	1E96h	—	1F16h	RA6PPS	1F96h	—
1D17h	LCDRL	1D97h	—	1E17h	CLC1GLS1	1E97h	—	1F17h	RA7PPS	1F97h	—
1D18h	LCDDATA0	1D98h	—	1E18h	CLC1GLS2	1E98h	—	1F18h	RB0PPS	1F98h	—
1D19h	LCDDATA1	1D99h	—	1E19h	CLC1GLS3	1E99h	—	1F19h	RB1PPS	1F99h	—
1D1Ah	LCDDATA2	1D9Ah	—	1E1Ah	CLC2CON	1E9Ah	—	1F1Ah	RB2PPS	1F9Ah	—
1D1Bh	LCDDATA3	1D9Bh	—	1E1Bh	CLC2POL	1E9Bh	—	1F1Bh	RB3PPS	1F9Bh	—
1D1Ch	LCDDATA4	1D9Ch	—	1E1Ch	CLC2SEL0	1E9Ch	T2AINPPS	1F1Ch	RB4PPS	1F9Ch	—
1D1Dh	LCDDATA5	1D9Dh	—	1E1Dh	CLC2SEL1	1E9Dh	T4AINPPS	1F1Dh	RB5PPS	1F9Dh	—
1D1Eh	LCDDATA6	1D9Eh	—	1E1Eh	CLC2SEL2	1E9Eh	—	1F1Eh	RB6PPS	1F9Eh	—
1D1Fh	LCDDATA7	1D9Fh	—	1E1Fh	CLC2SEL3	1E9Fh	—	1F1Fh	RB7PPS	1F9Fh	—
1D20h	LCDDATA8	1DA0h	—	1E20h	CLC2GLS0	1EA0h	—	1F20h	RC0PPS	1FA0h	—
1D21h	LCDDATA9	1DA1h	—	1E21h	CLC2GLS1	1EA1h	CCP1PPS	1F21h	RC1PPS	1FA1h	—
1D22h	LCDDATA10	1DA2h	—	1E22h	CLC2GLS2	1EA2h	CCP2PPS	1F22h	RC2PPS	1FA2h	—
1D23h	LCDDATA11	1DA3h	—	1E23h	CLC2GLS3	1EA3h	—	1F23h	RC3PPS	1FA3h	—
1D24h	LCDDATA12	1DA4h	—	1E24h	CLC3CON	1EA4h	—	1F24h	RC4PPS	1FA4h	—
1D25h	LCDDATA13	1DA5h	—	1E25h	CLC3POL	1EA5h	—	1F25h	—	1FA5h	—
1D26h	LCDDATA14	1DA6h	—	1E26h	CLC3SEL0	1EA6h	—	1F26h	RC6PPS	1FA6h	—
1D27h	LCDDATA15	1DA7h	—	1E27h	CLC3SEL1	1EA7h	—	1F27h	RC7PPS	1FA7h	—
1D28h	LCDDATA16	1DA8h	—	1E28h	CLC3SEL2	1EA8h	—	1F28h	RD0PPS(2)	1FA8h	—
1D29h	LCDDATA17	1DA9h	—	1E29h	CLC3SEL3	1EA9h	SMT1WINPPS	1F29h	RD1PPS(2)	1FA9h	—
1D2Ah	LCDDATA18	1DAAh	—	1E2Ah	CLC3GLS0	1EAAh	SMT1SIGPPS	1F2Ah	RD2PPS(2)	1FAAh	—
1D2Bh	LCDDATA19	1DABh	—	1E2Bh	CLC3GLS1	1EABh	—	1F2Bh	RD3PPS(2)	1FABh	—
1D2Ch	LCDDATA20	1DACh	—	1E2Ch	CLC3GLS2	1EACh	—	1F2Ch	RD4PPS(2)	1FACh	—
1D2Dh	LCDDATA21	1DADh	—	1E2Dh	CLC3GLS3	1EADh	—	1F2Dh	RD5PPS(2)	1FADh	—
1D2Eh	LCDDATA22	1DAEh	—	1E2Eh	CLC4CON	1EAEh	—	1F2Eh	RD6PPS(2)	1FAEh	—
1D2Fh	LCDDATA23	1DAFh	—	1E2Fh	CLC4POL	1EAFh	—	1F2Fh	RD7PPS(2)	1FAFh	—
1D30h	LCDDATA24	1DB0h	—	1E30h	CLC4SEL0	1EB0h	—	1F30h	RE0PPS(2)	1FB0h	—
1D31h	LCDDATA25	1DB1h	—	1E31h	CLC4SEL1	1EB1h	CWG1PPS	1F31h	RE1PPS(2)	1FB1h	—
1D32h	LCDDATA26	1DB2h	—	1E32h	CLC4SEL2	1EB2h	—	1F32h	RE2PPS(2)	1FB2h	—
1D33h	LCDDATA27	1DB3h	—	1E33h	CLC4SEL3	1EB3h	—	1F33h	RE3PPS	1FB3h	—
1D34h	LCDDATA28	1DB4h	—	1E34h	CLC4GLS0	1EB4h	—	1F34h	RE4PPS	1FB4h	—
1D35h	LCDDATA29	1DB5h	—	1E35h	CLC4GLS1	1EB5h	—	1F35h	RE5PPS	1FB5h	—
1D36h	LCDDATA30	1DB6h	—	1E36h	CLC4GLS2	1EB6h	—	1F36h	RE6PPS	1FB6h	—
1D37h	LCDDATA31	1DB7h	—	1E37h	CLC4GLS3	1EB7h	—	1F37h	RE7PPS	1FB7h	—
1D38h	LCDDATA32	1DB8h	—	1E38h	RF0PPS(3)	1EB8h	—	1F38h	ANSELA	1FB8h	—
1D39h	LCDDATA33	1DB9h	—	1E39h	RF1PPS(3)	1EB9h	—	1F39h	WPUA	1FB9h	—
1D3Ah	LCDDATA34	1DBAh	—	1E3Ah	RF2PPS(3)	1EBAh	—	1F3Ah	ODCONA	1FBAh	—

Note 1: Unimplemented locations read as '0'.
2: Present only on PIC16(L)F19156/76/86.
3: Present only on PIC16(L)F19185/86.

TABLE 4-12: SPECIAL FUNCTION REGISTER SUMMARY BANKS 0-63 PIC16(L)F19155/56/75/76/85/86 (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on: MCLR
Bank 16											
CPU CORE REGISTERS; see Table 4-3 for specifics											
80Ch	WDTCON0	—	—	WDTPS<4:0>					SWDTEN	--qq qqq0	--qq qqq0
80Dh	WDTCON1	—	WDTCS<2:0>			—	WINDOW<2:0>			-qqq -qqq	-qqq -qqq
80Eh	WDTPSL	PSCNT								0000 0000	0000 0000
80Fh	WDTPSH	PSCNT								0000 0000	0000 0000
810h	WDTTMR	—	WDTTMR<3:0>				STATE	PSCNT17	PSCNT16	-xxx x000	-xxx x000
811h	BORCON	SBOREN	—	—	—	—	—	—	BORRDY	1--- ---q	u--- ---u
812h	VREGCON	—	—	—	—	—	—	VREGPM	—	---- --x-	---- --x-
813h	PCON0	STKOVF	STKUNF	$\overline{\text{WDTWV}}$	$\overline{\text{RWDT}}$	$\overline{\text{RMCLR}}$	$\overline{\text{RI}}$	$\overline{\text{POR}}$	$\overline{\text{BOR}}$	0011 110q	qqqq qquu
814h	PCON1	—	—	—	—	—	—	$\overline{\text{MEMV}}$	$\overline{\text{VBATBOR}}$	---- --1x	---- --qu
815h	—	Unimplemented								---- ----	---- ----
816h	—	Unimplemented								---- ----	---- ----
817h	—	Unimplemented								---- ----	---- ----
818h	—	Unimplemented								---- ----	---- ----
819h	—	Unimplemented								---- ----	---- ----
81Ah	NVMADRL	NVMADR7	NVMADR6	NVMADR5	NVMADR4	NVMADR3	NVMADR2	NVMADR1	NVMADR0	xxxx xxxx	uuuu uuuu
81Bh	NVMADRH	—	NVMADR14	NVMADR13	NVMADR12	NVMADR11	NVMADR10	NVMADR9	NVMADR8	1xxx xxxx	1uuu uuuu
81Ch	NVMDATL	NVMDAT7	NVMDAT6	NVMDAT5	NVMDAT4	NVMDAT3	NVMDAT2	NVMDAT1	NVMDAT0	0000 0000	0000 0000
81Dh	NVMDATH	—	—	NVMDAT13	NVMDAT12	NVMDAT11	NVMDAT10	NVMDAT9	NVMDAT8	0000 0000	0000 0000
81Eh	NVMCON1	—	NVMREGS	LWLO	FREE	WRERR	WREN	WR	RD	0000 0000	0000 0000
81Fh	NVMCON2	NVMCON2<7:0>								---- ----	---- ----

Legend: x = unknown, u = unchanged, q = depends on condition, - = unimplemented, read as '0', r = reserved. Shaded locations unimplemented, read as '0'.

Note 1: Unimplemented data memory locations, read as '0'.

FIGURE 4-5: ACCESSING THE STACK EXAMPLE 2

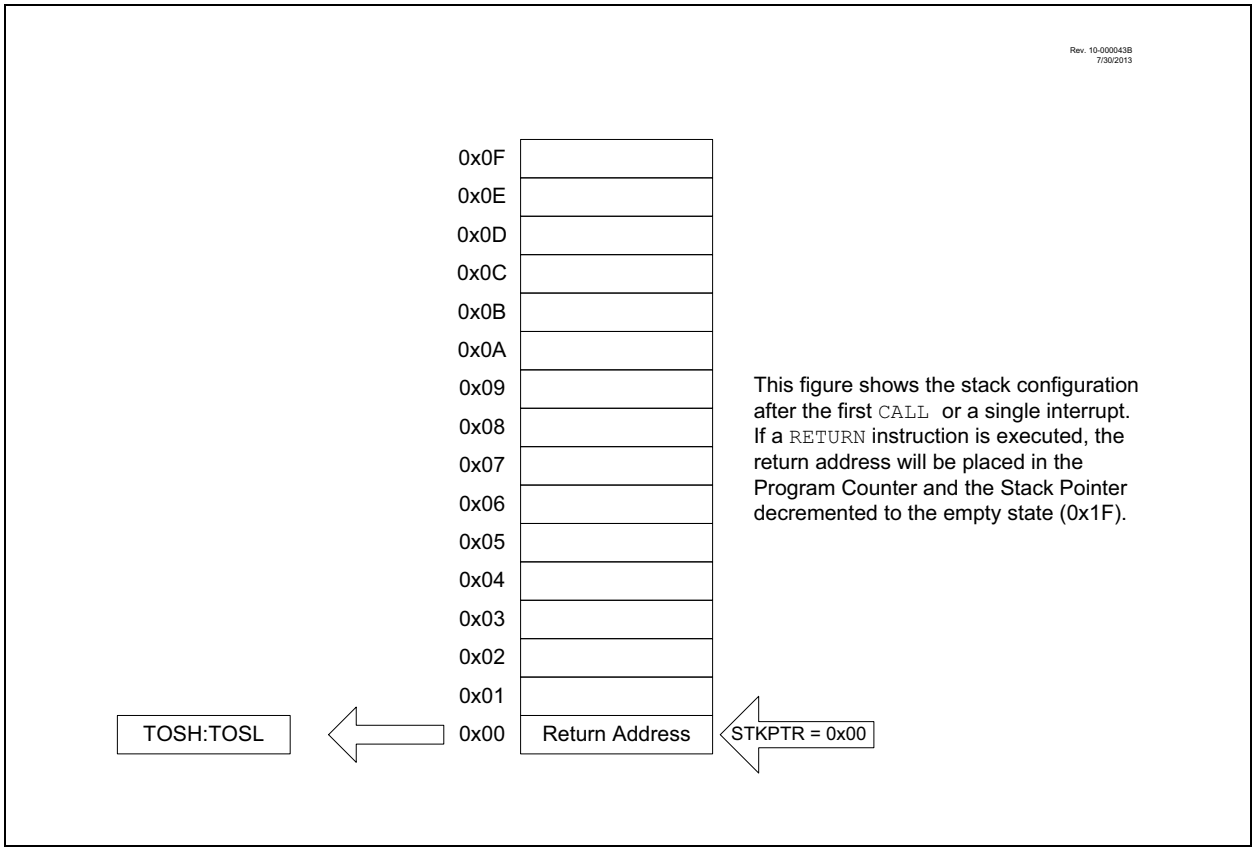
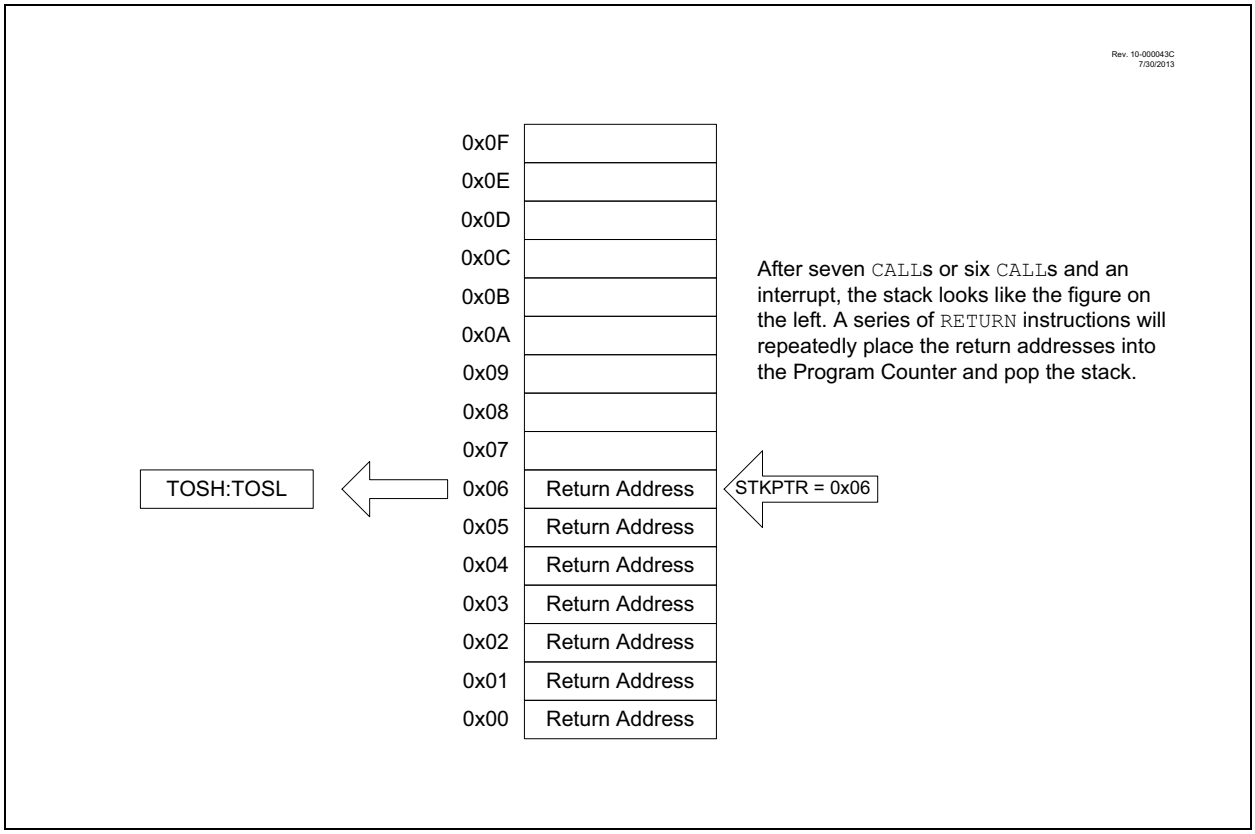


FIGURE 4-6: ACCESSING THE STACK EXAMPLE 3



8.15 Power Control (PCON) Registers

The Power Control (PCON) registers contain flag bits to differentiate between a:

- Power-on Reset ($\overline{POR}$)
- Brown-out Reset ($\overline{BOR}$)
- Reset Instruction Reset ($\overline{RI}$)
- $\overline{MCLR}$ Reset ($\overline{RMCLR}$)
- Watchdog Timer Reset ($\overline{RWDT}$)
- Watchdog Timer Window Violation Reset ($\overline{WDTWV}$)
- Stack Underflow Reset (STKUNF)
- Stack Overflow Reset (STKOVF)
- Memory Violation Reset ($\overline{MEMV}$)
- V_{BAT} Reset ($\overline{VBATBOR}$)

The PCON0 register bits are shown in Register 8-2.

The PCON1 register bits are shown in Register 8-3.

Hardware will change the corresponding register bit during the Reset process; if the Reset was not caused by the condition, the bit remains unchanged (Table 8-4).

Software should reset the bit to the inactive state after the restart (hardware will not reset the bit).

Software may also set any PCON bit to the active state, so that user code may be tested, but no reset action will be generated.

All bits in PCON1 and PCON0 will maintain their state when either V_{DD} or V_{BAT} is powered.

9.3 Clock Switching

The system clock source can be switched between external and internal clock sources via software using the New Oscillator Source (NOSC) and New Divider selection request (NDIV) bits of the OSCCON1 register.

9.3.1 NEW OSCILLATOR SOURCE (NOSC) AND NEW DIVIDER SELECTION REQUEST (NDIV) BITS

The New Oscillator Source (NOSC) and New Divider selection request (NDIV) bits of the OSCCON1 register select the system clock source and the frequency that are used for the CPU and peripherals.

When new values of NOSC and NDIV are written to OSCCON1, the current oscillator selection will continue to operate while waiting for the new clock source to indicate that it is stable and ready. In some cases, the newly requested source may already be in use, and is ready immediately. In the case of a divider-only change, the new and old sources are the same, and will be immediately ready. The device may enter Sleep while waiting for the switch as described in **Section 9.3.3 "Clock Switch and Sleep"**.

When the new oscillator is ready, the New Oscillator is Ready (NOSCR) bit of OSCCON3 and the Clock Switch Interrupt Flag (CSWIF) bit of PIR1 become set (CSWIF = 1). If Clock Switch Interrupts are enabled (CSWIE = 1), an interrupt will be generated at that time. The Oscillator Ready (ORDY) bit of OSCCON3 can also be polled to determine when the oscillator is ready in lieu of an interrupt.

If the Clock Switch Hold (CSWHOLD) bit of OSCCON3 is clear, the oscillator switch will occur when the new Oscillator's READY bit (NOSCR) is set, and the interrupt (if enabled) will be serviced at the new oscillator setting.

If CSWHOLD is set, the oscillator switch is suspended, while execution continues using the current (old) clock source. When the NOSCR bit is set, software should:

- set CSWHOLD = 0 so the switch can complete, or
- copy COSC into NOSC to abandon the switch.

If DOZE is in effect, the switch occurs on the next clock cycle, whether or not the CPU is operating during that cycle.

Changing the clock post-divider without changing the clock source (e.g., changing FOSC from 1 MHz to 2 MHz) is handled in the same manner as a clock source change, as described previously. The clock source will already be active, so the switch is relatively quick. CSWHOLD must be clear (CSWHOLD = 0) for the switch to complete.

The current COSC and CDIV are indicated in the OSCCON2 register up to the moment when the switch actually occurs, at which time OSCCON2 is updated and ORDY is set. NOSCR is cleared by hardware to indicate that the switch is complete.

9.3.2 PLL INPUT SWITCH

Switching between the PLL and any non-PLL source is managed as described above. The input to the PLL is established when NOSC selects the PLL, and maintained by the COSC setting.

When NOSC and COSC select the PLL with different input sources, the system continues to run using the COSC setting, and the new source is enabled per NOSC. When the new oscillator is ready (and CSWHOLD = 0), system operation is suspended while the PLL input is switched and the PLL acquires lock.

Note:	If the PLL fails to lock, the FSCM will trigger.

9.3.3 CLOCK SWITCH AND SLEEP

If OSCCON1 is written with a new value and the device is put to Sleep before the switch completes, the switch will not take place and the device will enter Sleep mode.

When the device wakes from Sleep and the CSWHOLD bit is clear, the device will wake with the 'new' clock active, and the clock switch interrupt flag bit (CSWIF) will be set.

When the device wakes from Sleep and the CSWHOLD bit is set, the device will wake with the 'old' clock active and the new clock will be requested again.

REGISTER 9-5: OSCEN: OSCILLATOR MANUAL ENABLE REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	U-0
EXTOEN	HFOEN	MFOEN	LFOEN	SOSCEN	ADOEN	—	—
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7 **EXTOEN:** External Oscillator Manual Request Enable bit
1 = EXTOSC is explicitly enabled, operating as specified by FEXTOSC
0 = EXTOSC could be enabled by another module

bit 6 **HFOEN:** HFINTOSC Oscillator Manual Request Enable bit
1 = HFINTOSC is explicitly enabled, operating as specified by OSCFRQ
0 = HFINTOSC could be enabled by another module

bit 5 **MFOEN:** MFINTOSC Oscillator Manual Request Enable bit
1 = MFINTOSC is explicitly enabled
0 = MFINTOSC could be enabled by another module

bit 4 **LFOEN:** LFINTOSC (31 kHz) Oscillator Manual Request Enable bit
1 = LFINTOSC is explicitly enabled
0 = LFINTOSC could be enabled by another module

bit 3 **SOSCEN:** Secondary (Timer1) Oscillator Manual Request bit
1 = Secondary oscillator is explicitly enabled, operating as specified by SOSCPWR
0 = Secondary oscillator could be enabled by another module

bit 2 **ADOEN:** FRC Oscillator Manual Request Enable bit
1 = FRC is explicitly enabled
0 = FRC could be enabled by another module

bit 1-0 **Unimplemented:** Read as ‘0’

11.0 POWER-SAVING OPERATION MODES

The purpose of the Power-Down modes is to reduce power consumption. There are three Power-Down modes: DOZE mode, IDLE mode and SLEEP mode.

11.1 DOZE Mode

DOZE mode saves power by reducing CPU execution and program memory (PFM) access, without affecting peripheral operation.

11.1.1 DOZE OPERATION

When the Doze Enable bit is set (DOZEN = 1), the CPU executes one instruction cycle out of every N cycles as defined by the DOZE<2:0> bits of the CPUDOZE register. FOSC and FOSC/4 clock sources are unaffected in Doze mode and peripherals can continue using these sources.

11.1.2 SYSTEM BEHAVIOR FOR INTERRUPTS DURING DOZE

If an interrupt occurs during Doze, it can be configured using the Recover-On-Interrupt bit (ROI) and the Doze-On-Exit bit (DOE). Refer to Table 11-1 for details about system behavior in all cases for a transition from *Main → ISR → Main*.

TABLE 11-1: SYSTEM BEHAVIOR FOR INTERRUPT DURING DOZE

DOZEN	ROI	Code flow			
		Main	ISR (1)	Return to Main	
0	0	Normal operation	Normal operation and DOE = DOZEN (in hardware) DOZEN = 0 (unchanged)	If DOE = 1 when return from interrupt; Doze operation and DOZEN = 1 (in hardware)	If DOE = 0 when return from interrupt; Normal operation and DOZEN = 0 (in hardware)
0	1	Normal operation	Normal operation and DOE = DOZEN (in hardware) DOZEN = 0 (in hardware)		
1	0	Doze operation	Doze operation and DOE = DOZEN (in hardware) DOZEN = 1 (unchanged)		
1	1	Doze operation	Normal operation and DOE = DOZEN (in hardware) DOZEN = 0 (in hardware)		

Note 1: User software can change the DOE bit in ISR.

For example, if ROI = 1 and DOZE<2:0> = 001, the instruction cycle ratio is 1:4. The CPU and memory operate for one instruction cycle and stay idle for the next three instruction cycles. The Doze operation is illustrated in Figure 11-1.

14.7 Register Definitions: PORTC

REGISTER 14-18: PORTC: PORTC REGISTER

R/W-x/u	R/W-x/u	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
RC7	RC6	—	RC4	RC3	RC2	RC1	RC0
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7-6 **RC<7:6>**: PORTC General Purpose I/O Pin bits(1)
1 = Port pin is ≥ VIH
0 = Port pin is ≤ VIL

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

bit 4-0 **RC<4:0>**: PORTC General Purpose I/O Pin bits(1)
1 = Port pin is ≥ VIH
0 = Port pin is ≤ VIL

Note 1: Writes to PORTC are actually written to corresponding LATC register. Reads from PORTC register is return of actual I/O pin values.

REGISTER 14-19: TRISC: PORTC TRI-STATE REGISTER

R/W-1/1	R/W-1/1	U-0	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
TRISC7	TRISC6	—	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7-6 **TRISC<7:6>:** PORTC Tri-State Control bits
1 = PORTC pin configured as an input (tri-stated)
0 = PORTC pin configured as an output

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

bit 4-0 **TRISC<4:0>:** PORTC Tri-State Control bits
1 = PORTC pin configured as an input (tri-stated)
0 = PORTC pin configured as an output

14.8 PORTD Registers

Note:	PORTD functionality is not available on the PIC16(L)F19155/56 family of devices.

14.8.1 DATA REGISTER

PORTD is an 8-bit wide, bidirectional port. The corresponding data direction register is TRISD (Register 14-2). Setting a TRISD bit (= 1) will make the corresponding PORTD pin an input (i.e., disable the output driver). Clearing a TRISD bit (= 0) will make the corresponding PORTD pin an output (i.e., enables output driver and puts the contents of the output latch on the selected pin). Example 14.2.8 shows how to initialize PORTD.

Reading the PORTD register (Register 14-1) reads the status of the pins, whereas writing to it will write to the PORT latch.

The PORT data latch LATD (Register 14-3) holds the output port data, and contains the latest value of a LATD or PORTD write.

EXAMPLE 14-3: INITIALIZING PORTD

```
; This code example illustrates
; initializing the PORTD register. The
; other ports are initialized in the same
; manner.

BANKSEL  PORTD        ;
CLRF     PORTD        ;Init PORTD
BANKSEL  LATD         ;Data Latch
CLRF     LATD         ;
BANKSEL  ANSELD       ;
CLRF     ANSELD       ;digital I/O
BANKSEL  TRISD        ;
MOVLW    B'00111000'  ;Set RD<5:3> as inputs
MOVWF    TRISD        ;and set RD<2:0> as
                      ;outputs
```

14.8.2 DIRECTION CONTROL

The TRISD register (Register 14-2) controls the PORTD pin output drivers, even when they are being used as analog inputs. The user should ensure the bits in the TRISD register are maintained set when using them as analog inputs. I/O pins configured as analog inputs always read '0'.

14.8.3 OPEN-DRAIN CONTROL

The ODCOND register (Register 14-6) controls the open-drain feature of the port. Open-drain operation is independently selected for each pin. When an ODCOND bit is set, the corresponding port output becomes an open-drain driver capable of sinking current only. When an ODCOND bit is cleared, the corresponding port output pin is the standard push-pull drive capable of sourcing and sinking current.

Note:	It is not necessary to set open-drain control when using the pin for I^2C; the I^2C module controls the pin and makes the pin open-drain.

14.8.4 SLEW RATE CONTROL

The SLRCOND register (Register 14-7) controls the slew rate option for each port pin. Slew rate control is independently selectable for each port pin. When an SLRCOND bit is set, the corresponding port pin drive is slew rate limited. When an SLRCOND bit is cleared, The corresponding port pin drive slews at the maximum rate possible.

14.8.5 INPUT THRESHOLD CONTROL

The INLVLD register (Register 14-8) controls the input voltage threshold for each of the available PORTD input pins. A selection between the Schmitt Trigger CMOS or the TTL Compatible thresholds is available. The input threshold is important in determining the value of a read of the PORTD register and also the level at which an interrupt-on-change occurs, if that feature is enabled. See Table 39-4 for more information on threshold levels.

Note:	Changing the input threshold selection should be performed while all peripheral modules are disabled. Changing the threshold level during the time a module is active may inadvertently generate a transition associated with an input pin, regardless of the actual voltage level on that pin.

REGISTER 19-15: ADCNT: ADC REPEAT COUNTER REGISTER

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
CNT<7:0>							
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7-0 **CNT<7:0>**: ADC Repeat Count bits
Counts the number of times that the ADC has been triggered and is used along with RPT to determine when the error threshold is checked when the computation is Low-pass Filter, Burst Average, or Average modes. See Table 19-2 for more details.

REGISTER 19-16: ADFLTRH: ADC FILTER HIGH BYTE REGISTER

R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
FLTR<15:8>							
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7-0 **FLTR<15:8>**: ADC Filter Output Most Significant bits
In Accumulate, Average, and Burst Average mode, this is equal to ACC right shifted by the ADCRS bits of ADCON2. In LPF mode, this is the output of the Low-pass Filter.

REGISTER 19-17: ADFLTRL: ADC FILTER LOW BYTE REGISTER

R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
FLTR<7:0>							
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7-0 **FLTR<7:0>**: ADC Filter Output Least Significant bits
In Accumulate, Average, and Burst Average mode, this is equal to ACC right shifted by the ADCRS bits of ADCON2. In LPF mode, this is the output of the Low-pass Filter.

33.2.6 SPI OPERATION IN SLEEP MODE

In SPI Master mode, module clocks may be operating at a different speed than when in Full-Power mode; in the case of the Sleep mode, all clocks are halted.

In SPI Master mode, when the Sleep mode is selected, all module clocks are halted and the transmission/reception will remain in that state until the device wakes. After the device returns to Run mode, the module will resume transmitting and receiving data.

In SPI Slave mode, the SPI Transmit/Receive Shift register operates asynchronously to the device. This allows the device to be placed in Sleep mode and data to be shifted into the SPI Transmit/Receive Shift register. When all eight bits have been received, the MSSP interrupt flag bit will be set and if enabled, will wake the device.

33.3 I²C MODE OVERVIEW

The Inter-Integrated Circuit (I^2C) bus is a multi-master serial data communication bus. Devices communicate in a master/slave environment where the master devices initiate the communication. A slave device is controlled through addressing.

The I^2C bus specifies two signal connections:

- Serial Clock (SCL)
- Serial Data (SDA)

Figure 33-2 and Figure 33-3 shows the block diagram of the MSSP module when operating in I^2C mode.

Both the SCL and SDA connections are bidirectional open-drain lines, each requiring pull-up resistors for the supply voltage. Pulling the line to ground is considered a logical zero and letting the line float is considered a logical one.

Note:	When using designated I^2C pins, the associated pin values in INLVLx will be ignored.

Figure 33-11 shows a typical connection between two processors configured as master and slave devices.

The I^2C bus can operate with one or more master devices and one or more slave devices.

There are four potential modes of operation for a given device:

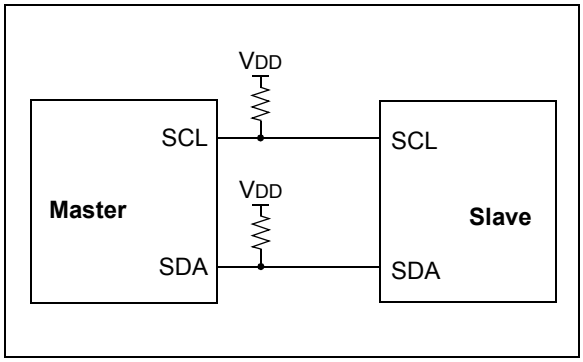
- Master Transmit mode
  (master is transmitting data to a slave)
- Master Receive mode
  (master is receiving data from a slave)
- Slave Transmit mode
  (slave is transmitting data to a master)
- Slave Receive mode
  (slave is receiving data from the master)

To begin communication, the master device sends out a Start condition followed by the address byte of the slave it intends to communicate with.

This is followed by a single Read/Write bit, which determines whether the master intends to transmit to or receive data from the slave device.

If the requested slave exists on the bus, it will respond with an Acknowledge bit, otherwise known as an $\overline{ACK}$. The master then continues to either transmit or receive data from the slave.

FIGURE 33-11: I²C MASTER/ SLAVE CONNECTION



On the last byte of data communicated, the master device may end the transmission by sending a Stop bit. If the master device is in Receive mode, it sends the Stop condition in place of the last $\overline{ACK}$ bit. A Stop condition is indicated by a low-to-high transition of the SDA line while the SCL line is held high.

In some cases, the master may want to maintain control of the bus and re-initiate another transmission. If so, the master device may send Restart condition in place of the Stop condition.

FIGURE 33-34: BUS COLLISION DURING START CONDITION (SCL = 0)

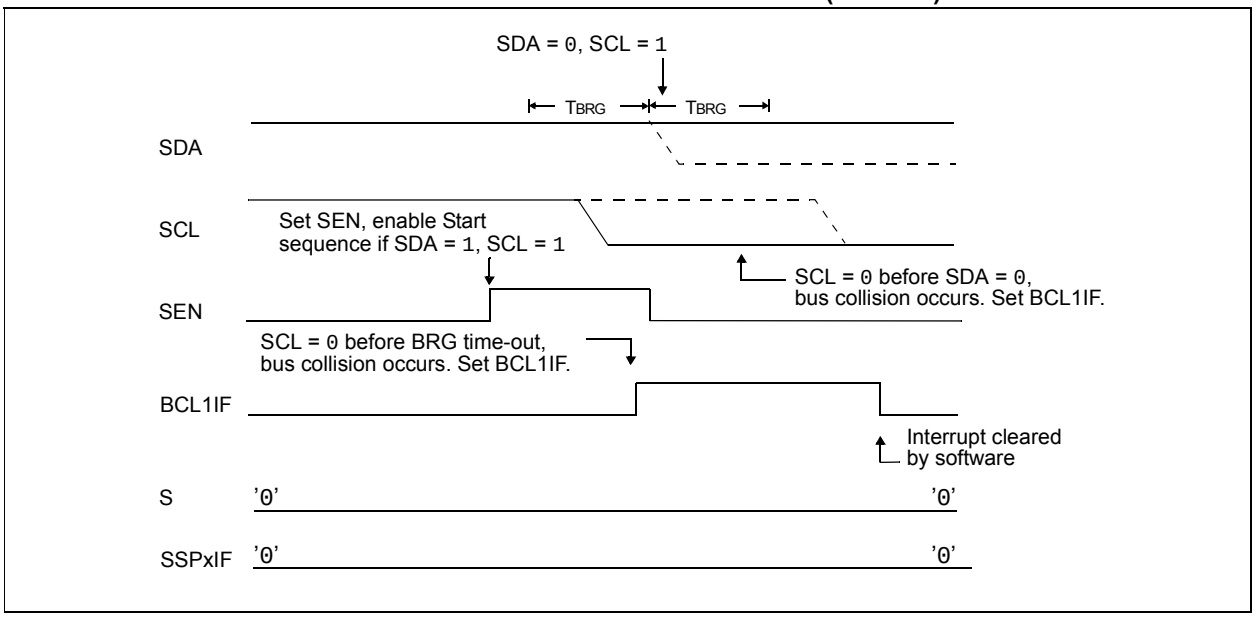


FIGURE 33-35: BRG RESET DUE TO SDA ARBITRATION DURING START CONDITION

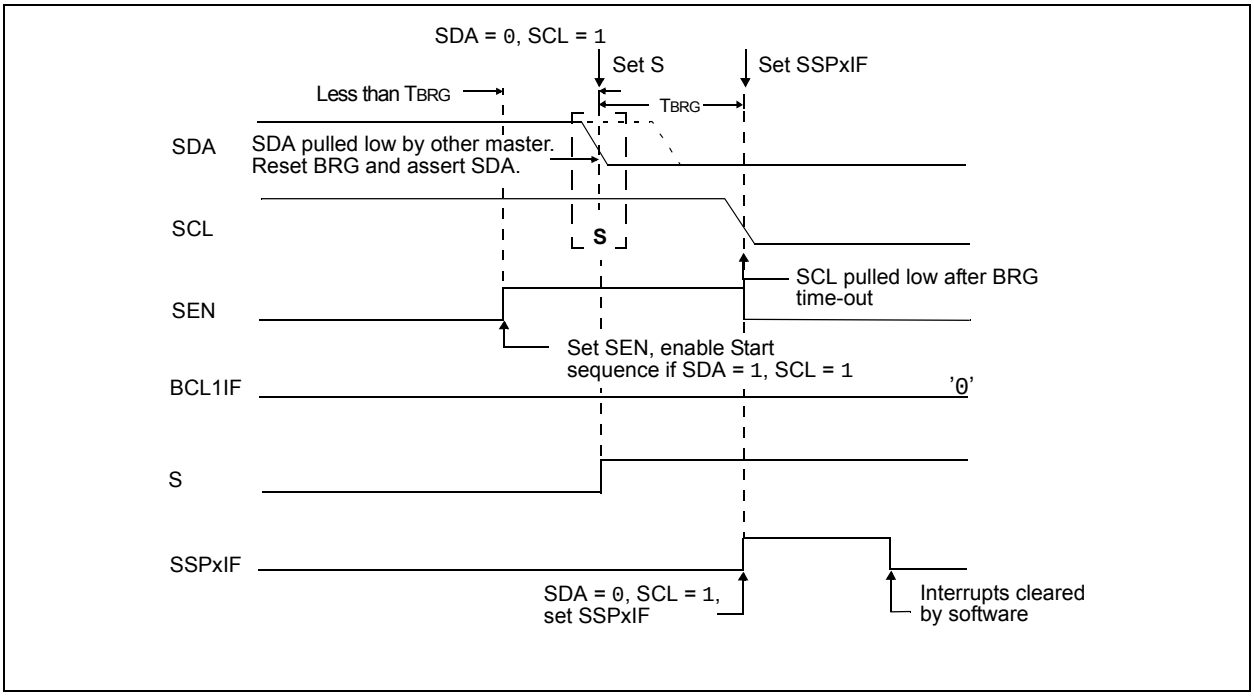


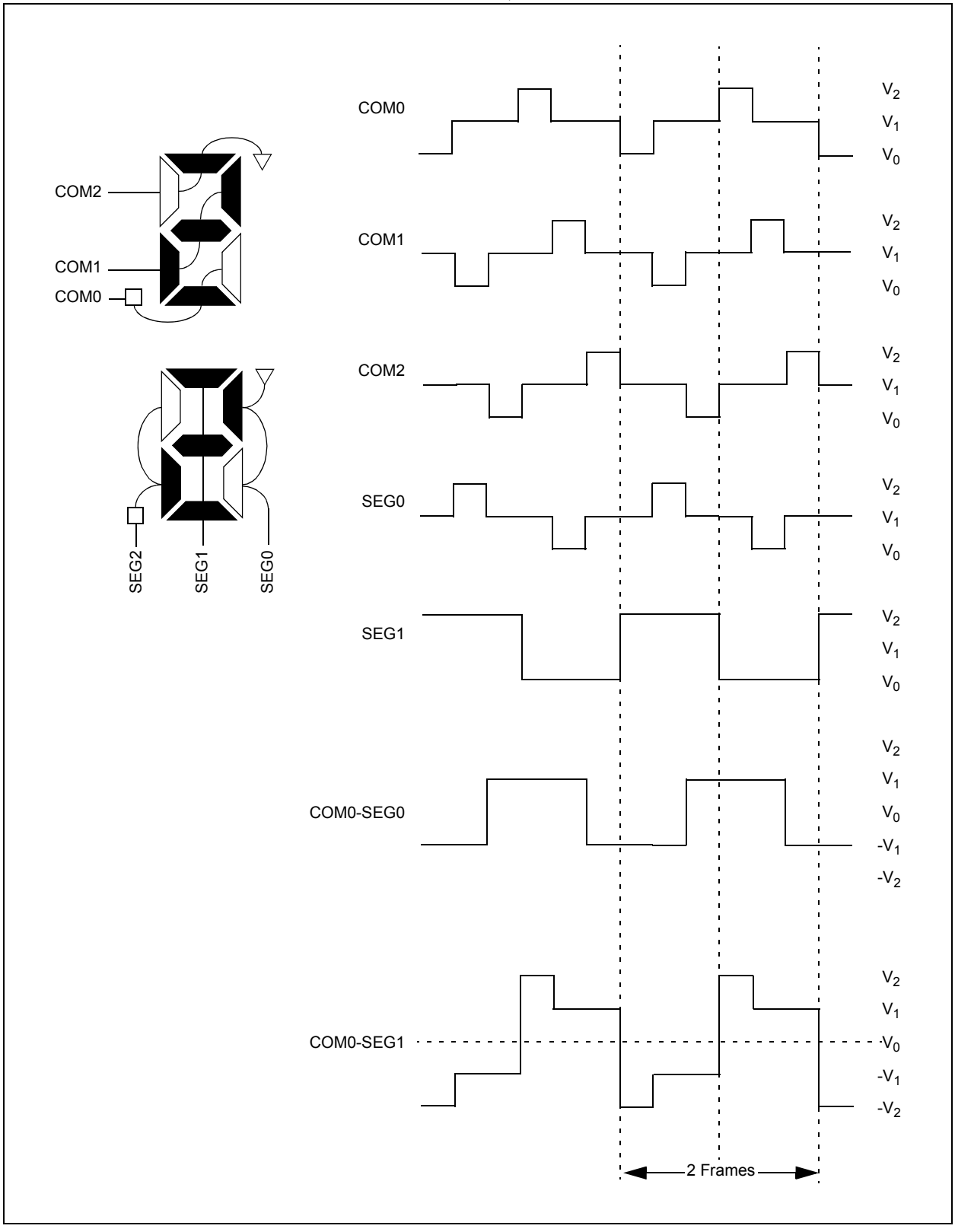
TABLE 34-4: BAUD RATE FOR ASYNCHRONOUS MODES (CONTINUED)

BAUD RATE	SYNC = 0, BRGH = 1, BRG16 = 0											
	Fosc = 32.000 MHz			Fosc = 20.000 MHz			Fosc = 18.432 MHz			Fosc = 11.0592 MHz		
	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)
300	—	—	—	—	—	—	—	—	—	—	—	—
1200	—	—	—	—	—	—	—	—	—	—	—	—
2400	—	—	—	—	—	—	—	—	—	—	—	—
9600	9615	0.16	207	9615	0.16	129	9600	0.00	119	9600	0.00	71
10417	10417	0.00	191	10417	0.00	119	10378	-0.37	110	10473	0.53	65
19.2k	19.23k	0.16	103	19.23k	0.16	64	19.20k	0.00	59	19.20k	0.00	35
57.6k	57.14k	-0.79	34	56.82k	-1.36	21	57.60k	0.00	19	57.60k	0.00	11
115.2k	117.64k	2.12	16	113.64k	-1.36	10	115.2k	0.00	9	115.2k	0.00	5

BAUD RATE	SYNC = 0, BRGH = 1, BRG16 = 0											
	Fosc = 8.000 MHz			Fosc = 4.000 MHz			Fosc = 3.6864 MHz			Fosc = 1.000 MHz		
	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)
300	—	—	—	—	—	—	—	—	—	300	0.16	207
1200	—	—	—	1202	0.16	207	1200	0.00	191	1202	0.16	51
2400	2404	0.16	207	2404	0.16	103	2400	0.00	95	2404	0.16	25
9600	9615	0.16	51	9615	0.16	25	9600	0.00	23	—	—	—
10417	10417	0.00	47	10417	0.00	23	10473	0.53	21	10417	0.00	5
19.2k	19231	0.16	25	19.23k	0.16	12	19.2k	0.00	11	—	—	—
57.6k	55556	-3.55	8	—	—	—	57.60k	0.00	3	—	—	—
115.2k	—	—	—	—	—	—	115.2k	0.00	1	—	—	—

BAUD RATE	SYNC = 0, BRGH = 0, BRG16 = 1											
	Fosc = 32.000 MHz			Fosc = 20.000 MHz			Fosc = 18.432 MHz			Fosc = 11.0592 MHz		
	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)
300	300.0	0.00	6666	300.0	-0.01	4166	300.0	0.00	3839	300.0	0.00	2303
1200	1200	-0.02	3332	1200	-0.03	1041	1200	0.00	959	1200	0.00	575
2400	2401	-0.04	832	2399	-0.03	520	2400	0.00	479	2400	0.00	287
9600	9615	0.16	207	9615	0.16	129	9600	0.00	119	9600	0.00	71
10417	10417	0.00	191	10417	0.00	119	10378	-0.37	110	10473	0.53	65
19.2k	19.23k	0.16	103	19.23k	0.16	64	19.20k	0.00	59	19.20k	0.00	35
57.6k	57.14k	-0.79	34	56.818	-1.36	21	57.60k	0.00	19	57.60k	0.00	11
115.2k	117.6k	2.12	16	113.636	-1.36	10	115.2k	0.00	9	115.2k	0.00	5

FIGURE 35-14: TYPE-B WAVEFORMS IN 1/3 MUX, 1/2 BIAS DRIVE



REGISTER 35-5: LCDVCON1: LCD VOLTAGE CONTROL 1 BITS

R/W-0/0	R/W-0/0	R-0	R-0	R-0	R/W-0/0	R/W-0/0	R/W-0/0
LPEN	EN5V	—	—	—	BIAS2	BIAS1	BIAS0
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 **LPEN**: LCD Charge Pump Low Power Enable (Low-Current (LC) mode enable)
1 = LCD Charge Pump is operating in Low-Current mode
0 = LCD Charge Pump is operating in Normal-Current mode

bit 6 **EN5V:** 5V Range Enable bit
1 = The pump generates 5.0V voltage range
0 = The pump generates 3.5V voltage range

bit 5-3 **Reserved**: Read as '0'

bit 2-0 **BIAS<2:0>:** Boost Pump Voltage Output Control bits (Only valid when LCDVSRC<2:0> = 100, 101, 110)
When EN5V = 0
111 = Set boost pump output to 3.50V
110 = Set boost pump output to 3.40V
101 = Set boost pump output to 3.30V
100 = Set boost pump output to 3.20V
011 = Set boost pump output to 3.10V
010 = Set boost pump output to 3.00V
001 = Set boost pump output to 2.90V
000 = Set boost pump output to 2.80V
When EN5V = 1
111 = Set boost pump output to 5.01V
110 = Set boost pump output to 4.83V
101 = Set boost pump output to 4.66V
100 = Set boost pump output to 4.48V
011 = Set boost pump output to 4.31V
010 = Set boost pump output to 4.13V
001 = Set boost pump output to 3.95V
000 = Set boost pump output to 3.78V

39.0 ELECTRICAL SPECIFICATIONS

39.1 Absolute Maximum Ratings(†)

Ambient temperature under bias ..... -40°C to +125°C
Storage temperature ..... -65°C to +150°C
Voltage on pins with respect to V_{SS}
- on V_{DD} pin
  - PIC16F19155/56/75/76/85/86 ..... -0.3V to +6.5V
  - PIC16(L)F19155/56/75/76/85/86 ..... -0.3V to +4.0V
- on $\overline{MCLR}$ pin ..... -0.3V to +9.0V
- on all other pins ..... -0.3V to (V_{DD} + 0.3V)

Maximum current
- on V_{SS} pin(1)
  - -40°C ≤ T_A ≤ +85°C ..... 350 mA
  - 85°C < T_A ≤ +125°C ..... 120 mA
- on V_{DD} pin for 28-Pin devices(1)
  - -40°C ≤ T_A ≤ +85°C ..... 250 mA
  - 85°C < T_A ≤ +125°C ..... 85 mA
- on V_{DD} pin for 40-Pin devices(1)
  - -40°C ≤ T_A ≤ +85°C ..... 350 mA
  - 85°C < T_A ≤ +125°C ..... 120 mA
- on any standard I/O pin ..... ±50 mA

Clamp current, I_K (V_{PIN} < 0 or V_{PIN} > V_{DD}) ..... ±20 mA
Total power dissipation(2) ..... 800 mW

Note 1: Maximum current rating requires even load distribution across I/O pins. Maximum current rating may be limited by the device package power dissipation characterizations, see Table 39-6 to calculate device specifications.

2: Power dissipation is calculated as follows:

$$P_{DIS} = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD} - V_{OH}) \times I_{OH}\} + \sum (V_{OI} \times I_{OL})$$

† NOTICE: Stresses above those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure above maximum rating conditions for extended periods may affect device reliability.

TABLE 39-26: I^2C BUS DATA REQUIREMENTS

Standard Operating Conditions (unless otherwise stated)							
Param. No.	**Symbol**	**Characteristic**		**Min.**	**Max.**	**Units**	**Conditions**
SP100*	THIGH	Clock high time	100 kHz mode	4.0	—	μs	Device must operate at a minimum of 1.5 MHz
			400 kHz mode	0.6	—	μs	Device must operate at a minimum of 10 MHz
			SSP module	1.5TCY	—		
SP101*	TLOW	Clock low time	100 kHz mode	4.7	—	μs	Device must operate at a minimum of 1.5 MHz
			400 kHz mode	1.3	—	μs	Device must operate at a minimum of 10 MHz
			SSP module	1.5TCY	—		
SP102*	TR	SDA and SCL rise time	100 kHz mode	—	1000	ns	
			400 kHz mode	20 + 0.1CB	300	ns	CB is specified to be from 10-400 pF
SP103*	TF	SDA and SCL fall time	100 kHz mode	—	250	ns	
			400 kHz mode	20 + 0.1CB	250	ns	CB is specified to be from 10-400 pF
SP106*	THD:DAT	Data input hold time	100 kHz mode	0	—	ns	
			400 kHz mode	0	0.9	μs	
SP107*	TSU:DAT	Data input setup time	100 kHz mode	250	—	ns	**(Note 2)**
			400 kHz mode	100	—	ns	
SP109*	TAA	Output valid from clock	100 kHz mode	—	3500	ns	**(Note 1)**
			400 kHz mode	—	—	ns	
SP110*	TBUF	Bus free time	100 kHz mode	4.7	—	μs	Time the bus must be free before a new transmission can start
			400 kHz mode	1.3	—	μs	
SP111	CB	Bus capacitive loading		—	400	pF	

* These parameters are characterized but not tested.

Note 1: As a transmitter, the device must provide this internal minimum delay time to bridge the undefined region (min. 300 ns) of the falling edge of SCL to avoid unintended generation of Start or Stop conditions.

2: A Fast mode (400 kHz) I^2C bus device can be used in a Standard mode (100 kHz) I^2C bus system, but the requirement $T_{SU:DAT} \geq 250$ ns must then be met. This will automatically be the case if the device does not stretch the low period of the SCL signal. If such a device does stretch the low period of the SCL signal, it must output the next data bit to the SDA line T_R max. + $T_{SU:DAT}$ = 1000 + 250 = 1250 ns (according to the Standard mode I^2C bus specification), before the SCL line is released.

41.0 DEVELOPMENT SUPPORT

The PIC® microcontrollers (MCU) and dsPIC® digital signal controllers (DSC) are supported with a full range of software and hardware development tools:

- Integrated Development Environment
  - MPLAB® X IDE Software
  - MPLAB® XPRESS IDE Software
- Compilers/Assemblers/Linkers
  - MPLAB XC Compiler
  - MPASM™ Assembler
  - MPLINK™ Object Linker/ MPLIB™ Object Librarian
  - MPLAB Assembler/Linker/Librarian for Various Device Families
- Simulators
  - MPLAB X SIM Software Simulator
- Emulators
  - MPLAB REAL ICE™ In-Circuit Emulator
- In-Circuit Debuggers/Programmers
  - MPLAB ICD 3
  - PICkit™ 3
- Device Programmers
  - MPLAB PM3 Device Programmer
- Low-Cost Demonstration/Development Boards, Evaluation Kits and Starter Kits
- Third-party development tools

41.1 MPLAB X Integrated Development Environment Software

The MPLAB X IDE is a single, unified graphical user interface for Microchip and third-party software, and hardware development tool that runs on Windows®, Linux and Mac OS® X. Based on the NetBeans IDE, MPLAB X IDE is an entirely new IDE with a host of free software components and plug-ins for high-performance application development and debugging. Moving between tools and upgrading from software simulators to hardware debugging and programming tools is simple with the seamless user interface.

With complete project management, visual call graphs, a configurable watch window and a feature-rich editor that includes code completion and context menus, MPLAB X IDE is flexible and friendly enough for new users. With the ability to support multiple tools on multiple projects with simultaneous debugging, MPLAB X IDE is also suitable for the needs of experienced users.

Feature-Rich Editor:

- Color syntax highlighting
- Smart code completion makes suggestions and provides hints as you type
- Automatic code formatting based on user-defined rules
- Live parsing

User-Friendly, Customizable Interface:

- Fully customizable interface: toolbars, toolbar buttons, windows, window placement, etc.
- Call graph window

Project-Based Workspaces:

- Multiple projects
- Multiple tools
- Multiple configurations
- Simultaneous debugging sessions

File History and Bug Tracking:

- Local file history feature
- Built-in support for Bugzilla issue tracker

APPENDIX A: DATA SHEET REVISION HISTORY

Revision A (6/2017)

This is the initial release of the document.