



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	Active
Core Processor	PIC
Core Size	8-Bit
Speed	32MHz
Connectivity	I ² C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	25
Program Memory Size	7KB (4K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	2.3V ~ 5.5V
Data Converters	A/D 17x10b; D/A 1x5b, 1x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-SSOP (0.209", 5.30mm Width)
Supplier Device Package	28-SSOP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f1713t-i-ss

Note 1: Default peripheral input. Alternate pins can be selected as the peripheral input with the PPS input selection registers.

2: All pin digital outputs default to PORT latch data. Alternate outputs can be selected as the peripheral digital output with the PPS output selection registers.

3: These peripheral functions are bidirectional. The output pin selections must be the same as the input pin selections.

4: Alternate outputs are excluded from solid shaded areas.

5: Alternate inputs are excluded from dot shaded areas.

3.3.1.1 STATUS Register

The STATUS register, shown in Register 3-1, contains:

- the arithmetic status of the ALU
- the Reset status

The STATUS register can be the destination for any instruction, like any other register. If the STATUS register is the destination for an instruction that affects the Z, DC or C bits, then the write to these three bits is disabled. These bits are set or cleared according to the device logic. Furthermore, the $\overline{\text{TO}}$ and $\overline{\text{PD}}$ bits are not writable. Therefore, the result of an instruction with the STATUS register as destination may be different than intended.

For example, CLRF STATUS will clear the upper three bits and set the Z bit. This leaves the STATUS register as '000u u1uu' (where u = unchanged).

It is recommended, therefore, that only BCF, BSF, SWAPF and MOVWF instructions are used to alter the STATUS register, because these instructions do not affect any Status bits. For other instructions not affecting any Status bits (Refer to **Section 33.0 "Instruction Set Summary"**).

Note: The C and DC bits operate as Borrow and Digit Borrow out bits, respectively, in subtraction.

3.4 Register Definitions: Status

REGISTER 3-1: STATUS: STATUS REGISTER

U-0	U-0	U-0	R-1/q	R-1/q	R/W-0/u	R/W-0/u	R/W-0/u
—	—	—	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC ⁽¹⁾	C ⁽¹⁾
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	q = Value depends on condition

bit 7-5 **Unimplemented:** Read as '0'

bit 4 **$\overline{\text{TO}}$:** Time-Out bit

- 1 = After power-up, CLRWD instruction or SLEEP instruction
- 0 = A WDT Time-out occurred

bit 3 **$\overline{\text{PD}}$:** Power-Down bit

- 1 = After power-up or by the CLRWD instruction
- 0 = By execution of the SLEEP instruction

bit 2 **Z:** Zero bit

- 1 = The result of an arithmetic or logic operation is zero
- 0 = The result of an arithmetic or logic operation is not zero

bit 1 **DC:** Digit Carry/Digit Borrow bit (ADDWF, ADDLW, SUBLW, SUBWF instructions)⁽¹⁾

- 1 = A carry-out from the 4th low-order bit of the result occurred
- 0 = No carry-out from the 4th low-order bit of the result

bit 0 **C:** Carry/Borrow bit⁽¹⁾ (ADDWF, ADDLW, SUBLW, SUBWF instructions)⁽¹⁾

- 1 = A carry-out from the Most Significant bit of the result occurred
- 0 = No carry-out from the Most Significant bit of the result occurred

Note 1: For Borrow, the polarity is reversed. A subtraction is executed by adding the two's complement of the second operand.

PIC16(L)F1713/6

TABLE 3-11: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Addr	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
Bank 2											
10Ch	LATA	LATA7	LATA6	LATA5	LATA4	LATA3	LATA2	LATA1	LATA0	xxxx xxxx	--uu -uuu
10Dh	LATB	LATB7	LATB6	LATB5	LATB4	LATB3	LATB2	LATB1	LATB0	xxxx xxxx	uuuu ----
10Eh	LATC	LATC7	LATC6	LATC5	LATC4	LATC3	LATC2	LATC1	LATC0	xxxx xxxx	uuuu uuuu
10Fh	—	Unimplemented								—	—
110h	—	Unimplemented								—	—
111h	CM1CON0	C1ON	C1OUT	—	C1POL	C1ZLF	C1SP	C1HYS	C1SYNC	00-0 0100	00-0 0100
112h	CM1CON1	C1INTP	C1INTN	C1PCH<2:0>			C1NCH<2:0>			0000 0000	0000 0000
113h	CM2CON0	C2ON	C2OUT	—	C2POL	C2ZLF	C2SP	C2HYS	C2SYNC	00-0 0100	00-0 0100
114h	CM2CON1	C2INTP	C2INTN	C2PCH<2:0>			C2NCH<2:0>			0000 0000	0000 0000
115h	CMOUT	—	—	—	—	—	—	MC2OUT	MC1OUT	---- --00	---- --00
116h	BORCON	SBOREN	BORFS	—	—	—	—	—	BORRDY	10-- ---q	uu-- ---u
117h	FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDAFVR<1:0>		ADFVR<1:0>		0q00 0000	0q00 0000
118h	DAC1CON0	DAC1EN	---	DAC1OE1	DAC1OE2	DAC1PSS<1:0>		---	DAC1NSS	0-00 00-0	0-00 00-0
119h	DAC1CON1	DAC1R<7:0>								0000 0000	0000 0000
11Ah	DAC2CON0	DAC2EN	—	DAC2OE1	DAC2OE2	DAC2PSS<1:0>		—	DAC2NSS	0-00 00-0	0-00 00-0
11Bh	DAC2CON1	—	—	—	DAC2R<4:0>					---0 0000	---0 0000
11Ch	ZCD1CON	ZCD1EN	—	ZCD1OUT	ZCD1POL	—	—	ZCD1INTP	ZCD1INTN	0-x0 --00	0-00 --00
11Dh	—	Unimplemented								—	—
11Eh	—	Unimplemented								—	—
11Fh	—	Unimplemented								—	—
Bank 3											
18Ch	ANSELA	—	—	ANSA5	ANSA4	ANSA3	ANSA2	ANSA1	ANSA0	--11 1111	---1 1111
18Dh	ANSELB	—	—	ANSB5	ANSB4	ANSB3	ANSB2	ANSB1	ANSB0	--11 1111	--11 ----
18Eh	ANSELC	ANSC7	ANSC6	ANSC5	ANSC4	ANSC3	ANSC2	—	—	1111 11--	1111 1111
18Fh	—	Unimplemented								—	—
190h	—	Unimplemented								—	—
191h	PMADRL	Program Memory Address Register Low Byte								0000 0000	0000 0000
192h	PMADRH	—	Program Memory Address Register High Byte							1000 0000	1000 0000
193h	PMDATL	Program Memory Read Data Register Low Byte								xxxx xxxx	uuuu uuuu
194h	PMDATH	—	—	Program Memory Read Data Register High Byte						--xx xxxx	--uu uuuu
195h	PMCON1	—	CFG5	LWLO	FREE	WRERR	WREN	WR	RD	-000 x000	-000 q000
196h	PMCON2	Program Memory Control Register 2								0000 0000	0000 0000
197h	VREGCON	—	—	—	—	—	—	VREGPM	Reserved	---- --01	---- --01
198h	—	Unimplemented								—	—
199h	RC1REG	USART Receive Data Register								0000 0000	0000 0000
19Ah	TX1REG	USART Transmit Data Register								0000 0000	0000 0000
19Bh	SP1BRGL	SP1BRG<7:0>								0000 0000	0000 0000
19Ch	SP1BRGH	SP1BRG<15:8>								0000 0000	0000 0000
19Dh	RC1STA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 0000	0000 0000
19Eh	TX1STA	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	0000 0010	0000 0010
19Fh	BAUD1CON	ABDOVF	RCIDL	—	SCKP	BRG16	—	WUE	ABDEN	01-0 0-00	01-0 0-00

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

- Note** 1: Unimplemented, read as '1'.
2: Unimplemented on PIC16(L)F1713/6.

PIC16(L)F1713/6

TABLE 3-11: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Addr	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
Bank 14-27											
x0Ch/ x8Ch — x1Fh/ x9Fh	—	Unimplemented								—	—
Bank 28											
E0Ch — E0Eh	—	Unimplemented								—	—
E0Fh	PPSLOCK	—	—	—	—	—	—	—	PPSLOCKED	---- --0	---- --0
E10h	INTPPS	—	—	—	INTPPS<4:0>					---0 1000	---u uuuu
E11h	T0CKIPPS	—	—	—	T0CKIPPS<4:0>					---0 0100	---u uuuu
E12h	T1CKIPPS	—	—	—	T1CKIPPS<4:0>					---1 0000	---u uuuu
E13h	T1GPPS	—	—	—	T1GPPS<4:0>					---0 1101	---u uuuu
E14h	CCP1PPS	—	—	—	CCP1PPS<4:0>					---1 0010	---u uuuu
E15h	CCP2PPS	—	—	—	CCP2PPS<4:0>					---1 0001	---u uuuu
E16h	—	Unimplemented								—	—
E17h	COGINPPS	—	—	—	COGINPPS<4:0>					---0 1000	---u uuuu
E18h	—	Unimplemented								—	—
E19h	—	Unimplemented								—	—
E1Ah E1Fh	—	Unimplemented								—	—
E20h	SSPCLKPPS	—	—	—	SSPCLKPPS<4:0>					---1 0011	---u uuuu
E21h	SSPDATPPS	—	—	—	SSPDATPPS<4:0>					---1 0100	---u uuuu
E22h	SSPSSPPS	—	—	—	SSPSSPPS<4:0>					---0 0101	---u uuuu
E23h	—	Unimplemented								—	—
E24h	RXPPS	—	—	—	RXPPS<4:0>					---1 0111	---u uuuu
E25h	CKPPS	—	—	—	CKPPS<4:0>					---1 0110	---u uuuu
E26h	—	Unimplemented								—	—
E27h	—	Unimplemented								—	—
E28h	CLCIN0PPS	—	—	—	CLCIN0PPS<4:0>					---0 0000	---u uuuu
E29h	CLCIN1PPS	—	—	—	CLCIN1PPS<4:0>					---0 0001	---u uuuu
E2Ah	CLCIN2PPS	—	—	—	CLCIN2PPS<4:0>					---0 1110	---u uuuu
E2Bh	CLCIN3PPS	—	—	—	CLCIN3PPS<4:0>					---0 1111	---u uuuu
E2Ch to E6Fh	—	Unimplemented								—	—

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

- Note** 1: Unimplemented, read as '1'.
2: Unimplemented on PIC16(L)F1713/6.

8.0 POWER-DOWN MODE (SLEEP)

The Power-down mode is entered by executing a SLEEP instruction.

Upon entering Sleep mode, the following conditions exist:

1. WDT will be cleared but keeps running, if enabled for operation during Sleep.
2. $\overline{\text{PD}}$ bit of the STATUS register is cleared.
3. $\overline{\text{TO}}$ bit of the STATUS register is set.
4. CPU clock is disabled.
5. 31 kHz LFINTOSC is unaffected and peripherals that operate from it may continue operation in Sleep.
6. Timer1 and peripherals that operate from Timer1 continue operation in Sleep when the Timer1 clock source selected is:
 - LFINTOSC
 - T1CKI
 - Secondary oscillator
7. ADC is unaffected, if the dedicated FRC oscillator is selected.
8. I/O ports maintain the status they had before SLEEP was executed (driving high, low or high-impedance).
9. Resets other than WDT are not affected by Sleep mode.

Refer to individual chapters for more details on peripheral operation during Sleep.

To minimize current consumption, the following conditions should be considered:

- I/O pins should not be floating
- External circuitry sinking current from I/O pins
- Internal circuitry sourcing current from I/O pins
- Current draw from pins with internal weak pull-ups
- Modules using 31 kHz LFINTOSC
- Modules using secondary oscillator

I/O pins that are high-impedance inputs should be pulled to VDD or VSS externally to avoid switching currents caused by floating inputs.

Examples of internal circuitry that might be sourcing current include modules such as the DAC and FVR modules. See **Section 22.0 “Operational Amplifier (OPA) Modules”** and **Section 14.0 “Fixed Voltage Reference (FVR)”** for more information on these modules.

8.1 Wake-up from Sleep

The device can wake-up from Sleep through one of the following events:

1. External Reset input on $\overline{\text{MCLR}}$ pin, if enabled
2. BOR Reset, if enabled
3. POR Reset
4. Watchdog Timer, if enabled
5. Any external interrupt
6. Interrupts by peripherals capable of running during Sleep (see individual peripheral for more information)

The first three events will cause a device Reset. The last three events are considered a continuation of program execution. To determine whether a device Reset or wake-up event occurred, refer to **Section 5.12 “Determining the Cause of a Reset”**.

When the SLEEP instruction is being executed, the next instruction (PC + 1) is prefetched. For the device to wake-up through an interrupt event, the corresponding interrupt enable bit must be enabled. Wake-up will occur regardless of the state of the GIE bit. If the GIE bit is disabled, the device continues execution at the instruction after the SLEEP instruction. If the GIE bit is enabled, the device executes the instruction after the SLEEP instruction, the device will then call the Interrupt Service Routine. In cases where the execution of the instruction following SLEEP is not desirable, the user should have a NOP after the SLEEP instruction.

The WDT is cleared when the device wakes up from Sleep, regardless of the source of wake-up.

PIC16(L)F1713/6

8.3 Register Definitions: Voltage Regulator Control

REGISTER 8-1: VREGCON: VOLTAGE REGULATOR CONTROL REGISTER⁽¹⁾

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0/0	R/W-1/1
—	—	—	—	—	—	VREGPM	Reserved
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 7-2 **Unimplemented:** Read as '0'
bit 1 **VREGPM:** Voltage Regulator Power Mode Selection bit
 1 = Low-Power Sleep mode enabled in Sleep⁽²⁾
 Draws lowest current in Sleep, slower wake-up
 0 = Normal-Power mode enabled in Sleep⁽²⁾
 Draws higher current in Sleep, faster wake-up
bit 0 **Reserved:** Read as '1'. Maintain this bit set.

Note 1: PIC16F1713/6 only.
Note 2: See Section 34.0 "Electrical Specifications".

TABLE 8-1: SUMMARY OF REGISTERS ASSOCIATED WITH POWER-DOWN MODE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
STATUS	—	—	—	TO	PD	Z	DC	C	19
VREGCON ⁽¹⁾	—	—	—	—	—	—	VREGPM	Reserved	94
WDTCON	—	—	WDTPS<4:0>					SWDTEN	98

Legend: — = unimplemented location, read as '0'. Shaded cells are not used in Power-Down mode.

Note 1: PIC16F1713/6 only.

PIC16(L)F1713/6

11.6 Register Definitions: PORTC

REGISTER 11-17: PORTC: PORTC 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
RC7	RC6	RC5	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 7-0 **RC<7:0>**: PORTC General Purpose I/O Pin bits⁽¹⁾
1 = Port pin is $\geq V_{IH}$
0 = Port pin is $\leq V_{IL}$

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 11-18: TRISC: PORTC TRI-STATE REGISTER

R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
TRISC7	TRISC6	TRISC5	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 7-0 **TRISC<7:0>**: PORTC Tri-State Control bits
1 = PORTC pin configured as an input (tri-stated)
0 = PORTC pin configured as an output

REGISTER 11-19: LATC: PORTC DATA LATCH 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
LATC7	LATC6	LATC5	LATC4	LATC3	LATC2	LATC1	LATC0
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 **LATC<7:0>**: PORTC Output Latch Value bits

REGISTER 13-3: IOCAF: INTERRUPT-ON-CHANGE PORTA FLAG REGISTER

R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0
IOCAF7	IOCAF6	IOCAF5	IOCAF4	IOCAF3	IOCAF2	IOCAF1	IOCAF0
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	HS - Bit is set in hardware

bit 7-0 **IOCAF<7:0>**: Interrupt-on-Change PORTA Flag bits
 1 = An enabled change was detected on the associated pin.
 Set when IOCAPx = 1 and a rising edge was detected on RAX, or when IOCANx = 1 and a falling edge was detected on RAX.
 0 = No change was detected, or the user cleared the detected change.

REGISTER 13-4: IOCBP: INTERRUPT-ON-CHANGE PORTB POSITIVE EDGE 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	R/W-0/0	R/W-0/0
IOCBP7	IOCBP6	IOCBP5	IOCBP4	IOCBP3	IOCBP2	IOCBP1	IOCBP0
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 **IOCBP<7:0>**: Interrupt-on-Change PORTB Positive Edge Enable bits
 1 = Interrupt-on-Change enabled on the pin for a positive going edge. IOCBFx bit and IOCIF flag will be set upon detecting an edge.
 0 = Interrupt-on-Change disabled for the associated pin.

PIC16(L)F1713/6

REGISTER 13-5: IOCBN: INTERRUPT-ON-CHANGE PORTB NEGATIVE EDGE 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	R/W-0/0	R/W-0/0
IOCBN7	IOCBN6	IOCBN5	IOCBN4	IOCBN3	IOCBN2	IOCBN1	IOCBN0
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

IOCBN<7:0>: Interrupt-on-Change PORTB Negative Edge Enable bits

1 = Interrupt-on-Change enabled on the pin for a negative going edge. IOCBFx bit and IOCIF flag will be set upon detecting an edge.

0 = Interrupt-on-Change disabled for the associated pin.

REGISTER 13-6: IOCBF: INTERRUPT-ON-CHANGE PORTB FLAG REGISTER

R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0
IOCBF7	IOCBF6	IOCBF5	IOCBF4	IOCBF3	IOCBF2	IOCBF1	IOCBF0
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

HS - Bit is set in hardware

bit 7-0

IOCBF<7:0>: Interrupt-on-Change PORTB Flag bits

1 = An enabled change was detected on the associated pin.

Set when IOCBPx = 1 and a rising edge was detected on RBx, or when IOCBNx = 1 and a falling edge was detected on RBx.

0 = No change was detected, or the user cleared the detected change.

FIGURE 18-2: SIMPLIFIED COG BLOCK DIAGRAM (STEERED PWM MODE, GXMD = 0)

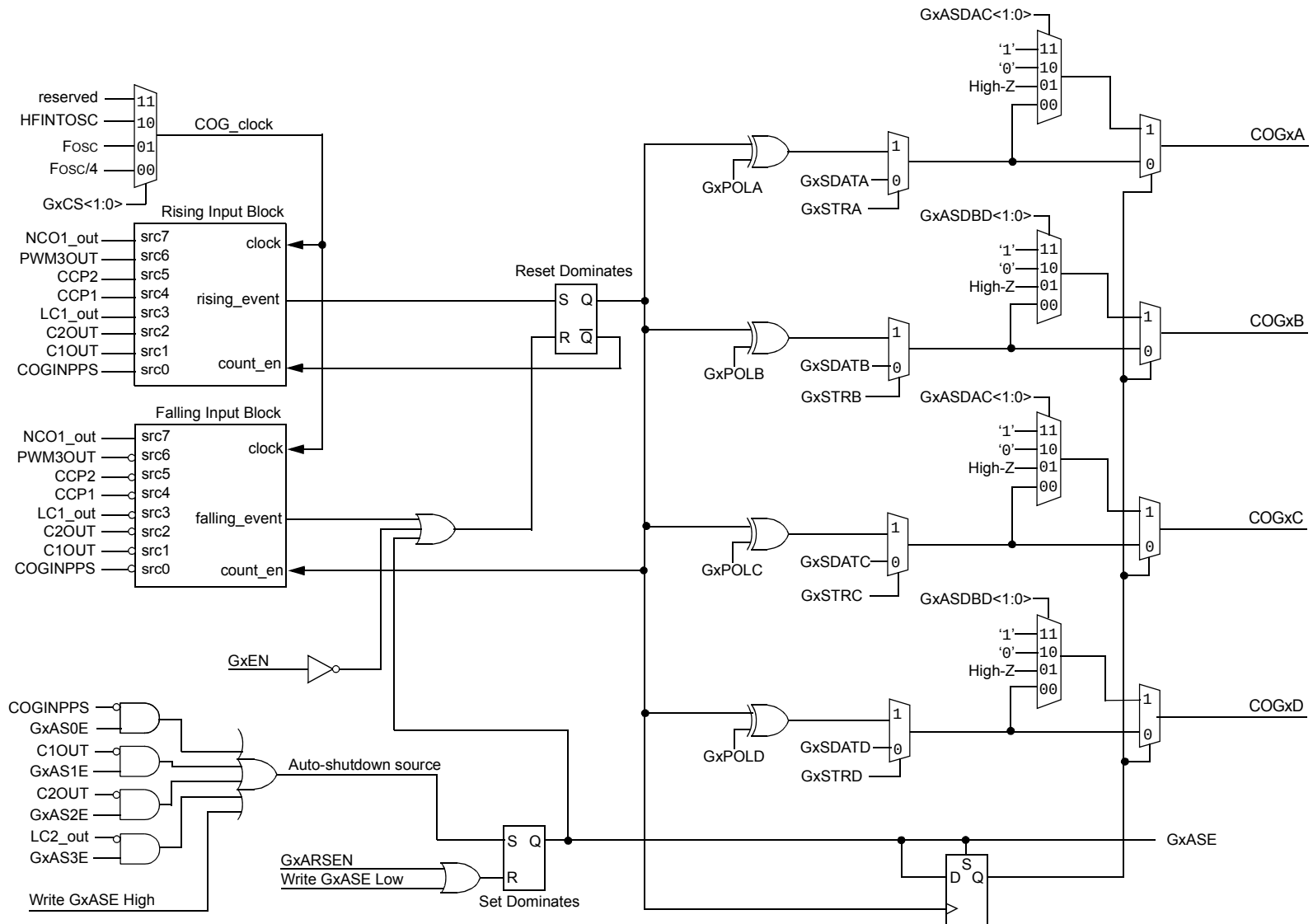
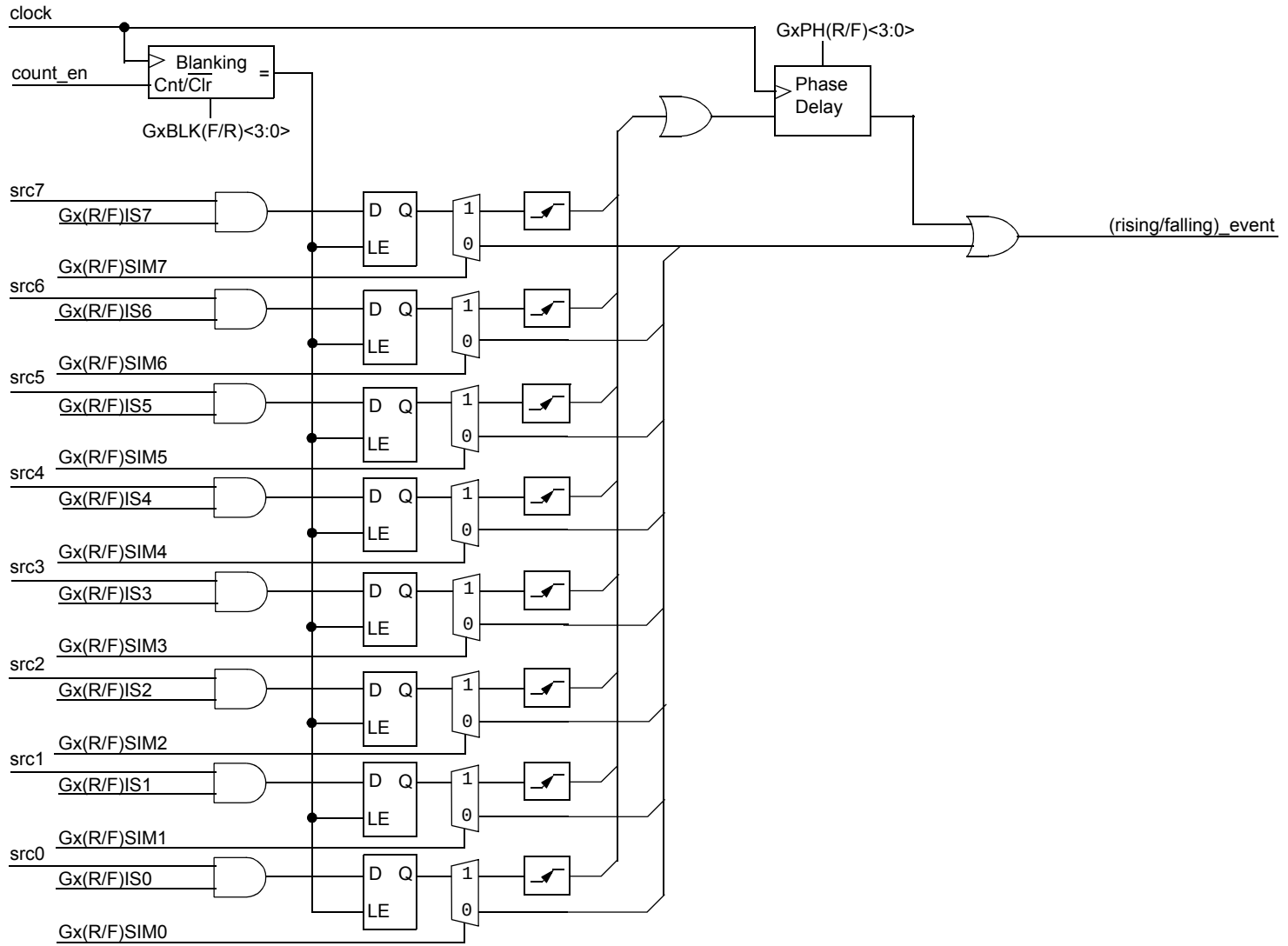


FIGURE 18-7: COG (RISING/FALLING) INPUT BLOCK

18.7.1 CUMULATIVE UNCERTAINTY

It is not possible to create more than one COG_clock of uncertainty by successive stages. Consider that the phase-delay stage comes after the blanking stage, the dead-band stage comes after either the blanking or phase-delay stages, and the blanking stage comes after the dead-band stage. When the preceding stage is enabled, the output of that stage is necessarily synchronous with the COG_clock, which removes any possibility of uncertainty in the succeeding stage.

EQUATION 18-1: PHASE, DEAD-BAND, AND BLANKING TIME CALCULATION

$$T_{\min} = \frac{\text{Count}}{F_{\text{COG_clock}}}$$

$$T_{\max} = \frac{\text{Count} + 1}{F_{\text{COG_clock}}}$$

$$T_{\text{uncertainty}} = T_{\max} - T_{\min}$$

Also:

$$T_{\text{uncertainty}} = \frac{1}{F_{\text{COG_clock}}}$$

Where:

T	Count
Rising Phase Delay	COGxPHR
Falling Phase Delay	COGxPHF
Rising Dead Band	COGxDBR
Falling Dead Band	COGxDBF
Rising Event Blanking	COGxBLKR
Falling Event Blanking	COGxBLKF

EXAMPLE 18-1: TIMER UNCERTAINTY

Given:

$$\text{Count} = Ah = 10d$$

$$F_{\text{COG_Clock}} = 8\text{MHz}$$

Therefore:

$$T_{\text{uncertainty}} = \frac{1}{F_{\text{COG_clock}}}$$

$$= \frac{1}{8\text{MHz}} = 125\text{ns}$$

Proof:

$$T_{\min} = \frac{\text{Count}}{F_{\text{COG_clock}}}$$

$$= 125\text{ns} \cdot 10d = 1.25\mu\text{s}$$

$$T_{\max} = \frac{\text{Count} + 1}{F_{\text{COG_clock}}}$$

$$= 125\text{ns} \cdot (10d + 1)$$

$$= 1.375\mu\text{s}$$

Therefore:

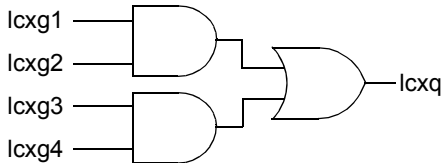
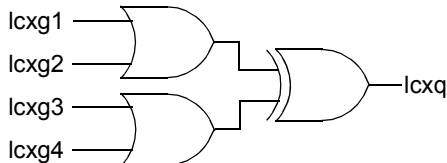
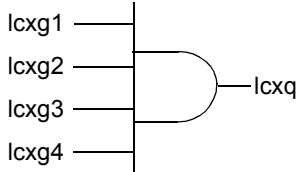
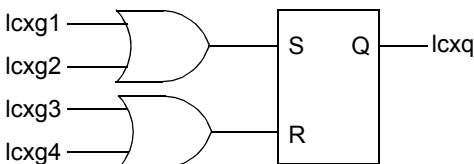
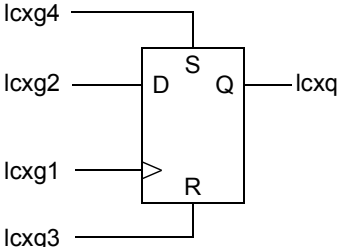
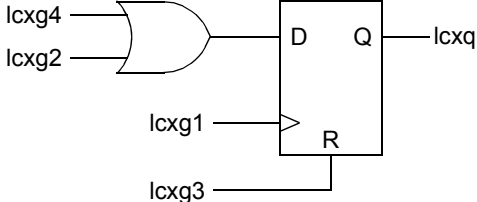
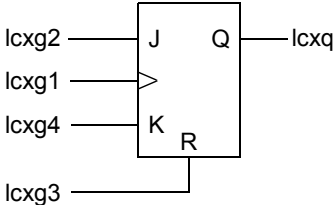
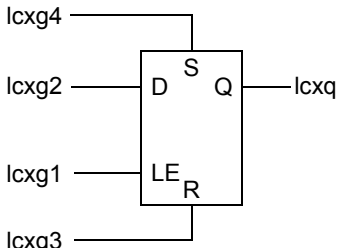
$$T_{\text{uncertainty}} = T_{\max} - T_{\min}$$

$$= 1.375\mu\text{s} - 1.25\mu\text{s}$$

$$= 125\text{ns}$$

PIC16(L)F1713/6

FIGURE 19-3: PROGRAMMABLE LOGIC FUNCTIONS

AND - OR  $LCxMODE<2:0>= 000$	OR - XOR  $LCxMODE<2:0>= 001$
4-Input AND  $LCxMODE<2:0>= 010$	S-R Latch  $LCxMODE<2:0>= 011$
1-Input D Flip-Flop with S and R  $LCxMODE<2:0>= 100$	2-Input D Flip-Flop with R  $LCxMODE<2:0>= 101$
J-K Flip-Flop with R  $LCxMODE<2:0>= 110$	1-Input Transparent Latch with S and R  $LCxMODE<2:0>= 111$

PIC16(L)F1713/6

REGISTER 19-10: CLCxGLS3: GATE 4 LOGIC SELECT 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
LCxG4D4T	LCxG4D4N	LCxG4D3T	LCxG4D3N	LCxG4D2T	LCxG4D2N	LCxG4D1T	LCxG4D1N
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 **LCxG4D4T:** Gate 4 Data 4 True (non-inverted) bit
1 = lcx4T is gated into lcxg4
0 = lcx4T is not gated into lcxg4
- bit 6 **LCxG4D4N:** Gate 4 Data 4 Negated (inverted) bit
1 = lcx4N is gated into lcxg4
0 = lcx4N is not gated into lcxg4
- bit 5 **LCxG4D3T:** Gate 4 Data 3 True (non-inverted) bit
1 = lcx3T is gated into lcxg4
0 = lcx3T is not gated into lcxg4
- bit 4 **LCxG4D3N:** Gate 4 Data 3 Negated (inverted) bit
1 = lcx3N is gated into lcxg4
0 = lcx3N is not gated into lcxg4
- bit 3 **LCxG4D2T:** Gate 4 Data 2 True (non-inverted) bit
1 = lcx2T is gated into lcxg4
0 = lcx2T is not gated into lcxg4
- bit 2 **LCxG4D2N:** Gate 4 Data 2 Negated (inverted) bit
1 = lcx2N is gated into lcxg4
0 = lcx2N is not gated into lcxg4
- bit 1 **LCxG4D1T:** Gate 4 Data 1 True (non-inverted) bit
1 = lcx1T is gated into lcxg4
0 = lcx1T is not gated into lcxg4
- bit 0 **LCxG4D1N:** Gate 4 Data 1 Negated (inverted) bit
1 = lcx1N is gated into lcxg4
0 = lcx1N is not gated into lcxg4

31.4.1 AUTO-BAUD DETECT

The EUSART module supports automatic detection and calibration of the baud rate.

In the Auto-Baud 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. The Baud Rate Generator is used to time the period of a received 55h (ASCII “U”) which is the Sync character for the LIN bus. The unique feature of this character is that it has five rising edges including the Stop bit edge.

Setting the ABDEN bit of the BAUDCON register starts the auto-baud calibration sequence. While the ABD sequence takes place, the EUSART state machine is held in Idle. On the first rising edge of the receive line, after the Start bit, the SPBRG begins counting up using the BRG counter clock as shown in Figure 31-6. The fifth rising edge will occur on the RX pin at the end of the eighth bit period. At that time, an accumulated value totaling the proper BRG period is left in the SPBRGH, SPBRGL register pair, the ABDEN bit is automatically cleared and the RCIF interrupt flag is set. The value in the RCREG needs to be read to clear the RCIF interrupt. RCREG content should be discarded. When calibrating for modes that do not use the SPBRGH register the user can verify that the SPBRGL register did not overflow by checking for 00h in the SPBRGH register.

The BRG auto-baud clock is determined by the BRG16 and BRGH bits as shown in Table 31-6. During ABD, both the SPBRGH and SPBRGL registers are used as a 16-bit counter, independent of the BRG16 bit setting. While calibrating the baud rate period, the SPBRGH

and SPBRGL registers are clocked at 1/8th the BRG base clock rate. The resulting byte measurement is the average bit time when clocked at full speed.

Note 1: If the WUE bit is set with the ABDEN bit, auto-baud detection will occur on the byte following the Break character (see **Section 31.4.3 “Auto-Wake-up on Break”**).

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 EUSART baud rates are not possible.

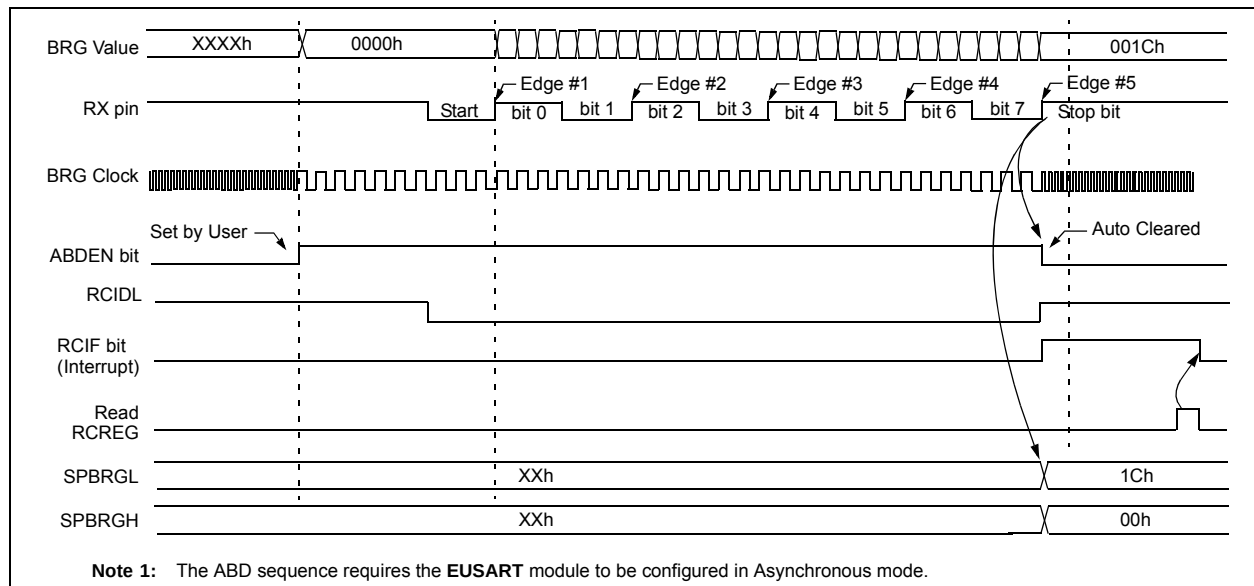
3: During the auto-baud process, the auto-baud counter starts counting at one. Upon completion of the auto-baud sequence, to achieve maximum accuracy, subtract 1 from the SPBRGH:SPBRGL register pair.

TABLE 31-6: BRG COUNTER CLOCK RATES

BRG16	BRGH	BRG Base Clock	BRG ABD Clock
0	0	Fosc/64	Fosc/512
0	1	Fosc/16	Fosc/128
1	0	Fosc/16	Fosc/128
1	1	Fosc/4	Fosc/32

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

FIGURE 31-6: AUTOMATIC BAUD RATE CALIBRATION



31.5 EUSART Synchronous Mode

Synchronous serial communications are typically used in systems with a single master and one or more slaves. The master device contains the necessary circuitry for baud rate generation and supplies the clock for all devices in the system. Slave devices can take advantage of the master clock by eliminating the internal clock generation circuitry.

There are two signal lines in Synchronous mode: a bidirectional data line and a clock line. Slaves use the external clock supplied by the master to shift the serial data into and out of their respective receive and transmit shift registers. Since the data line is bidirectional, synchronous operation is half-duplex only. Half-duplex refers to the fact that master and slave devices can receive and transmit data but not both simultaneously. The EUSART can operate as either a master or slave device.

Start and Stop bits are not used in synchronous transmissions.

31.5.1 SYNCHRONOUS MASTER MODE

The following bits are used to configure the EUSART for synchronous master operation:

- SYNC = 1
- CSRC = 1
- SREN = 0 (for transmit); SREN = 1 (for receive)
- CREN = 0 (for transmit); CREN = 1 (for receive)
- SPEN = 1

Setting the SYNC bit of the TXSTA register configures the device for synchronous operation. Setting the CSRC bit of the TXSTA register configures the device as a master. Clearing the SREN and CREN bits of the RCSTA register ensures that the device is in the Transmit mode, otherwise the device will be configured to receive. Setting the SPEN bit of the RCSTA register enables the EUSART.

31.5.1.1 Master Clock

Synchronous data transfers use a separate clock line, which is synchronous with the data. A device configured as a master transmits the clock on the TX/CK line. The TX/CK pin output driver is automatically enabled when the EUSART is configured for synchronous transmit or receive operation. Serial data bits change on the leading edge to ensure they are valid at the trailing edge of each clock. One clock cycle is generated for each data bit. Only as many clock cycles are generated as there are data bits.

31.5.1.2 Clock Polarity

A clock polarity option is provided for Microwire compatibility. Clock polarity is selected with the SCKP bit of the BAUDCON register. Setting the SCKP bit sets the clock Idle state as high. When the SCKP bit is set, the data changes on the falling edge of each clock. Clearing the SCKP bit sets the Idle state as low. When the SCKP bit is cleared, the data changes on the rising edge of each clock.

31.5.1.3 Synchronous Master Transmission

Data is transferred out of the device on the RX/DT pin. The RX/DT and TX/CK pin output drivers are automatically enabled when the EUSART is configured for synchronous master transmit operation.

A transmission is initiated by writing a character to the TXREG register. If the TSR still contains all or part of a previous character the new character data is held in the TXREG until the last bit of the previous character has been transmitted. If this is the first character, or the previous character has been completely flushed from the TSR, the data in the TXREG is immediately transferred to the TSR. The transmission of the character commences immediately following the transfer of the data to the TSR from the TXREG.

Each data bit changes on the leading edge of the master clock and remains valid until the subsequent leading clock edge.

Note: The TSR register is not mapped in data memory, so it is not available to the user.

31.5.1.4 Synchronous Master Transmission Setup:

1. Initialize the SPBRGH, SPBRGL register pair and the BRGH and BRG16 bits to achieve the desired baud rate (see **Section 31.4 “EUSART Baud Rate Generator (BRG)”**).
2. Enable the synchronous master serial port by setting bits SYNC, SPEN and CSRC.
3. Disable Receive mode by clearing bits SREN and CREN.
4. Enable Transmit mode by setting the TXEN bit.
5. If 9-bit transmission is desired, set the TX9 bit.
6. If interrupts are desired, set the TXIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
7. If 9-bit transmission is selected, the ninth bit should be loaded in the TX9D bit.
8. Start transmission by loading data to the TXREG register.

PIC16(L)F1713/6

CALL Call Subroutine

Syntax: [*label*] CALL *k*

Operands: $0 \leq k \leq 2047$

Operation: $(PC) + 1 \rightarrow TOS$,
 $k \rightarrow PC<10:0>$,
 $(PCLATH<6:3>) \rightarrow PC<14:11>$

Status Affected: None

Description: Call Subroutine. First, return address $(PC + 1)$ is pushed onto the stack. The 11-bit immediate address is loaded into PC bits $<10:0>$. The upper bits of the PC are loaded from PCLATH. CALL is a 2-cycle instruction.

CLRWDTClear Watchdog Timer

Syntax: [*label*] CLRWDTClear Watchdog Timer

Operands: None

Operation: $00h \rightarrow WDT$
 $0 \rightarrow WDT \text{ prescaler}$,
 $1 \rightarrow \overline{TO}$
 $1 \rightarrow \overline{PD}$

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

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

CALLW Subroutine Call With W

Syntax: [*label*] CALLW

Operands: None

Operation: $(PC) + 1 \rightarrow TOS$,
 $(W) \rightarrow PC<7:0>$,
 $(PCLATH<6:0>) \rightarrow PC<14:8>$

Status Affected: None

Description: Subroutine call with W. First, the return address $(PC + 1)$ is pushed onto the return stack. Then, the contents of W is loaded into $PC<7:0>$, and the contents of PCLATH into $PC<14:8>$. CALLW is a 2-cycle instruction.

COMF Complement f

Syntax: [*label*] COMF *f*,*d*

Operands: $0 \leq f \leq 127$
 $d \in [0,1]$

Operation: $(\bar{f}) \rightarrow (\text{destination})$

Status Affected: Z

Description: The contents of register 'f' are complemented. If 'd' is '0', the result is stored in W. If 'd' is '1', the result is stored back in register 'f'.

CLRF Clear f

Syntax: [*label*] CLRF *f*

Operands: $0 \leq f \leq 127$

Operation: $00h \rightarrow (f)$
 $1 \rightarrow Z$

Status Affected: Z

Description: The contents of register 'f' are cleared and the Z bit is set.

DECF Decrement f

Syntax: [*label*] DECF *f*,*d*

Operands: $0 \leq f \leq 127$
 $d \in [0,1]$

Operation: $(f) - 1 \rightarrow (\text{destination})$

Status Affected: Z

Description: Decrement register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f'.

CLRWClear W

Syntax: [*label*] CLRW

Operands: None

Operation: $00h \rightarrow (W)$
 $1 \rightarrow Z$

Status Affected: Z

Description: W register is cleared. Zero bit (Z) is set.

PIC16(L)F1713/6

TABLE 34-24: SPI MODE REQUIREMENTS

Standard Operating Conditions (unless otherwise stated)							
Param No.	Symbol	Characteristic	Min.	Typ†	Max.	Units	Conditions
SP70*	TssL2sCH, TssL2sCL	$\overline{SS}\downarrow$ to SCK $\downarrow$ or SCK $\uparrow$ input	2.25 Tcy	—	—	ns	
SP71*	TsCH	SCK input high time (Slave mode)	Tcy + 20	—	—	ns	
SP72*	TsCL	SCK input low time (Slave mode)	Tcy + 20	—	—	ns	
SP73*	TdIV2sCH, TdIV2sCL	Setup time of SDI data input to SCK edge	100	—	—	ns	
SP74*	TsCH2dIL, TsCL2dIL	Hold time of SDI data input to SCK edge	100	—	—	ns	
SP75*	TdoR	SDO data output rise time	—	10	25	ns	3.0V ≤ VDD ≤ 5.5V
			—	25	50	ns	1.8V ≤ VDD ≤ 5.5V
SP76*	TdoF	SDO data output fall time	—	10	25	ns	
SP77*	TssH2doZ	$\overline{SS}\uparrow$ to SDO output high-impedance	10	—	50	ns	
SP78*	TsCR	SCK output rise time (Master mode)	—	10	25	ns	3.0V ≤ VDD ≤ 5.5V
			—	25	50	ns	1.8V ≤ VDD ≤ 5.5V
SP79*	TsCF	SCK output fall time (Master mode)	—	10	25	ns	
SP80*	TsCH2doV, TsCL2doV	SDO data output valid after SCK edge	—	—	50	ns	3.0V ≤ VDD ≤ 5.5V
			—	—	145	ns	1.8V ≤ VDD ≤ 5.5V
SP81*	TdoV2sCH, TdoV2sCL	SDO data output setup to SCK edge	1 Tcy	—	—	ns	
SP82*	TssL2doV	SDO data output valid after $\overline{SS}\downarrow$ edge	—	—	50	ns	
SP83*	TsCH2ssH, TsCL2ssH	$\overline{SS}\uparrow$ after SCK edge	1.5 Tcy + 40	—	—	ns	

* These parameters are characterized but not tested.

† Data in “Typ” column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

36.0 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
- 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

36.1 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