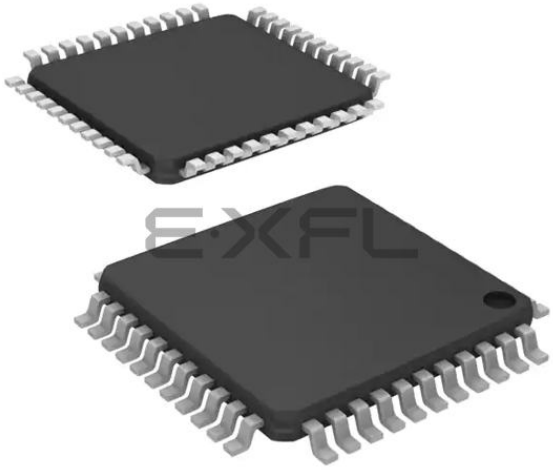


Welcome to E-XFL.COM

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, POR, PWM, WDT
Number of I/O	35
Program Memory Size	28KB (16K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 28x10b; D/A 1x5b, 1x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-TQFP
Supplier Device Package	44-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf1719-i-pt

TABLE 2: 40/44-PIN ALLOCATION TABLE (PIC16(L)F1717/9)

I/O(2)	PDIP	TQFP	UQFN	ADC	Reference	Comparator		Op Amp	DAC	Zero Cross	Timers			CCP		NCO	PWM		COG				MSSP			EUSART		CLC				Interrupt	Pullup	Basic
RA0	2	19	17	AN0		C1IN0- C2IN0-																							CLCIN0(1)				IOC	Y	
RA1	3	20	18	AN1		C1IN1- C2IN1-		OPA1OUT																					CLCIN1(1)				IOC	Y	
RA2	4	21	19	AN2	V_{REF}-	C1IN0+ C2IN0+			DAC1OUT1																								IOC	Y	
RA3	5	22	20	AN3	V_{REF}+	C1IN1+																											IOC	Y	
RA4	6	23	21					OPA1IN+			T0CKI(1)																						IOC	Y	
RA5	7	24	22	AN4				OPA1IN-	DAC2OUT1																nSS(1)								IOC	Y	
RA6	14	31	29																														IOC	Y	OSC2 CLKOUT
RA7	13	30	28																														IOC	Y	OSC1 CLKIN
RB0	33	8	8	AN12		C2IN1+				ZCD										COG1IN(1)													INT(1) IOC	Y	
RB1	34	9	9	AN10		C1IN3- C2IN3-		OPA2OUT																									IOC	Y	
RB2	35	10	10	AN8				OPA2IN-																									IOC	Y	
RB3	36	11	11	AN9		C1IN2- C2IN2-		OPA2IN+																									IOC	Y	
RB4	37	14	12	AN11																													IOC	Y	
RB5	38	15	13	AN13							T1G(1)																						IOC	Y	
RB6	39	16	14																												CLCIN2(1)		IOC	Y	ICSPCLK
RB7	40	17	15						DAC1OUT2 DAC2OUT2																						CLCIN3(1)			IOC	Y	ICSPDAT
RC0	15	32	30								T1CKI(1) SOSCO																						IOC	Y	
RC1	16	35	31								SOSCI			CCP2(1)																			IOC	Y	
RC2	17	36	32	AN14										CCP1(1)																			IOC	Y	
RC3	18	37	33	AN15																					SCL/SCK(1)								IOC	Y	

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.

TABLE 1-3: PIC16(L)F1717/9 PINOUT DESCRIPTION (CONTINUED)

Name	Function	Input Type	Output Type	Description
RB1/AN10/C1IN3-/C2IN3-/ OPA2OUT	RB1	TTL/ST	CMOS	General purpose I/O.
	AN10	AN	—	ADC Channel 10 input.
	C1IN3-	AN	—	Comparator C1 negative input.
	C2IN3-	AN	—	Comparator C2 negative input.
	OPA2OUT	—	AN	Operational Amplifier 2 output.
RB2/AN8/OPA2IN-	RB2	TTL/ST	CMOS	General purpose I/O.
	AN8	AN	—	ADC Channel 8 input.
	OPA2IN-	AN	—	Operational Amplifier 2 inverting input.
RB3/AN9/C1IN2-/C2IN2-/ OPA2IN+	RB3	TTL/ST	CMOS	General purpose I/O.
	AN9	AN	—	ADC Channel 9 input.
	C1IN2-	AN	—	Comparator C1 negative input.
	C2IN2-	AN	—	Comparator C2 negative input.
	OPA2IN+	AN	—	Operational Amplifier 2 non-inverting input.
RB4/AN11	RB4	TTL/ST	CMOS	General purpose I/O.
	AN11	AN	—	ADC Channel 11 input.
RB5/AN13/T1G(1)	RB5	TTL/ST	CMOS	General purpose I/O.
	AN13	AN	—	ADC Channel 13 input.
	T1G	TTL/ST	—	Timer1 gate input.
RB6/CLCIN2(1)/ICSPCLK	RB6	TTL/ST	CMOS	General purpose I/O.
	CLCIN2	TTL/ST	—	Configurable Logic Cell source input.
	ICSPCLK	ST	—	Serial Programming Clock.
RB7/DAC1OUT2/DAC2OUT2/ CLCIN3(1)/ICSPDAT	RB7	TTL/ST	CMOS	General purpose I/O.
	DAC1OUT2	—	AN	Digital-to-Analog Converter output.
	DAC2OUT2	—	AN	Digital-to-Analog Converter output.
	CLCIN3	TTL/ST	—	Configurable Logic Cell source input.
	ICSPDAT	ST	CMOS	ICSP™ Data I/O.
RC0/T1CKI(1)/SOSCO	RC0	TTL/ST	CMOS	General purpose I/O.
	T1CKI	ST	—	Timer1 clock input.
	SOSCO	XTAL	XTAL	Secondary Oscillator Connection.
RC1/SOSCI/CCP2(1)	RC1	TTL/ST	CMOS	General purpose I/O.
	SOSCI	XTAL	XTAL	Secondary Oscillator Connection.
	CCP2	TTL/ST	—	Capture input.
RC2/AN14/CCP1(1)	RC2	TTL/ST	CMOS	General purpose I/O.
	AN14	AN	—	ADC Channel 14 input.
	CCP1	TTL/ST	—	Capture input.
RC3/AN15/SCK(1)/SCL(1)	RC3	TTL/ST	CMOS	General purpose I/O.
	AN15	AN	—	ADC Channel 15 input.
	SCK	TTL/ST	—	SPI clock input.
	SCL	I^2C	OD	I^2C clock.

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: 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.

TABLE 3-8: PIC16(L)F1718/9 MEMORY MAP, BANK 24-31

BANK 24		BANK 25		BANK 26		BANK 27		BANK 28		BANK 29		BANK 30		BANK 31	
C00h	Core Registers (Table 3-2)	C80h	Core Registers (Table 3-2)	D00h	Core Registers (Table 3-2)	D80h	Core Registers (Table 3-2)	E00h	Core Registers (Table 3-2)	E80h	Core Registers (Table 3-2)	F00h	Core Registers (Table 3-2)	F80h	Core Registers (Table 3-2)
C0Bh		C8Bh		D0Bh		D8Bh		E0Bh		E8Bh		F0Bh		F8Bh	
C0Ch	—	C8Ch	—	D0Ch	—	D8Ch	—	E0Ch	See Table 3-9 for register mapping details	E8Ch	See Table 3-9 for register mapping details	F0Ch	See Table 3-9 for register mapping details	F8Ch	See Table 3-10 for register mapping details
C0Dh	—	C8Dh	—	D0Dh	—	D8Dh	—	E0Dh		E8Dh		F0Dh		F8Dh	
C0Eh	—	C8Eh	—	D0Eh	—	D8Eh	—	E0Eh		E8Eh		F0Eh		F8Eh	
C0Fh	—	C8Fh	—	D0Fh	—	D8Fh	—	E0Fh		E8Fh		F0Fh		F8Fh	
C10h	—	C90h	—	D10h	—	D90h	—	E10h		E90h		F10h		F90h	
C11h	—	C91h	—	D11h	—	D91h	—	E11h		E91h		F11h		F91h	
C12h	—	C92h	—	D12h	—	D92h	—	E12h		E92h		F12h		F92h	
C13h	—	C93h	—	D13h	—	D93h	—	E13h		E93h		F13h		F93h	
C14h	—	C94h	—	D14h	—	D94h	—	E14h		E94h		F14h		F94h	
C15h	—	C95h	—	D15h	—	D95h	—	E15h		E95h		F15h		F95h	
C16h	—	C96h	—	D16h	—	D96h	—	E16h		E96h		F16h		F96h	
C17h	—	C97h	—	D17h	—	D97h	—	E17h		E97h		F17h		F97h	
C18h	—	C98h	—	D18h	—	D98h	—	E18h		E98h		F18h		F98h	
C19h	—	C99h	—	D19h	—	D99h	—	E19h		E99h		F19h		F99h	
C1Ah	—	C9Ah	—	D1Ah	—	D9Ah	—	E1Ah		E9Ah		F1Ah		F9Ah	
C1Bh	—	C9Bh	—	D1Bh	—	D9Bh	—	E1Bh		E9Bh		F1Bh		F9Bh	
C1Ch	—	C9Ch	—	D1Ch	—	D9Ch	—	E1Ch		E9Ch		F1Ch		F9Ch	
C1Dh	—	C9Dh	—	D1Dh	—	D9Dh	—	E1Dh		E9Dh		F1Dh		F9Dh	
C1Eh	—	C9Eh	—	D1Eh	—	D9Eh	—	E1Eh		E9Eh		F1Eh		F9Eh	
C1Fh	—	C9Fh	—	D1Fh	—	D9Fh	—	E1Fh		E9Fh		F1Fh		F9Fh	
C20h	General Purpose Register 80 Bytes	CA0h	General Purpose Register 32 Bytes	D20h	Unimplemented Read as '0'	DA0h	Unimplemented Read as '0'	E20h		EA0h		F20h		FA0h	
C6Fh		CBFh		D6Fh		DEFh		E6Fh		EEFh		F6Fh		FEFh	
C70h	Accesses 70h – 7Fh	CC0h	Unimplemented Read as '0'	D70h	Accesses 70h – 7Fh	DF0h	Accesses 70h – 7Fh	E70h	Accesses 70h – 7Fh	EF0h	Accesses 70h – 7Fh	F70h	Accesses 70h – 7Fh	FF0h	Accesses 70h – 7Fh
CFFh		CFFh		D7Fh		DFFh		E7Fh		EFFh		F7Fh		FFFh	

Legend: = Unimplemented data memory locations, read as '0'.

TABLE 3-12: 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 7											
38Ch	INLVLA	INLVLA7	INLVLA6	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	1111 1111	1111 1111
38Dh	INLVLB	INLVLB7	INLVLB6	INLVLB5	INLVLB4	INLVLB3	INLVLB2	INLVLB1	INLVLB0	1111 1111	1111 1111
38Eh	INLVLC	INLVLC7	INLVLC6	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	1111 1111	1111 1111
38Fh	INLVLD(1)	INLVLD7	INLVLD6	INLVLD5	INLVLD4	INLVLD3	INLVLD2	INLVLD1	INLVLD0	1111 1111	1111 1111
390h	INLVLE					INLVLE3	INLVLE2(1)	INLVLE1(1)	INLVLE0(1)	---- 1111	---- 1111
391h	IOCAP	IOCAP7	IOCAP6	IOCAP5	IOCAP4	IOCAP3	IOCAP2	IOCAP1	IOCAP0	0000 0000	--00 0000
392h	IOCAN	IOCAN7	IOCAN6	IOCAN5	IOCAN4	IOCAN3	IOCAN2	IOCAN1	IOCAN0	0000 0000	--00 0000
393h	IOCAF	IOCAF7	IOCAF6	IOCAF5	IOCAF4	IOCAF3	IOCAF2	IOCAF1	IOCAF0	0000 0000	--00 0000
394h	IOCBP	IOCBP7	IOCBP6	IOCBP5	IOCBP4	IOCBP3	IOCBP2	IOCBP1	IOCBP0	0000 0000	0000 ----
395h	IOCBN	IOCBN7	IOCBN6	IOCBN5	IOCBN4	IOCBN3	IOCBN2	IOCBN1	IOCBN0	0000 0000	0000 ----
396h	IOCBF	IOCBF7	IOCBF6	IOCBF5	IOCBF4	IOCBF3	IOCBF2	IOCBF1	IOCBF0	0000 0000	0000 ----
397h	IOCCP	IOCCP7	IOCCP6	IOCCP5	IOCCP4	IOCCP3	IOCCP2	IOCCP1	IOCCP0	0000 0000	0000 0000
398h	IOCCN	IOCCN7	IOCCN6	IOCCN5	IOCCN4	IOCCN3	IOCCN2	IOCCN1	IOCCN0	0000 0000	0000 0000
399h	IOCCF	IOCCF7	IOCCF6	IOCCF5	IOCCF4	IOCCF3	IOCCF2	IOCCF1	IOCCF0	0000 0000	0000 0000
39Ah — 39Ch	—	Unimplemented								—	—
39Dh	IOCEP	—	—	—	—	IOCEP3	—	—	—	---- 0---	---- 0---
39Eh	IOCEN	—	—	—	—	IOCEN3	—	—	—	---- 0---	---- 0---
39Fh	IOCEF	—	—	—	—	IOCEF3	—	—	—	---- 0---	---- 0---
Bank 8											
40Ch — 414h	—	Unimplemented								—	—
415h	TMR4	Holding Register for the 8-bit TMR4 Register								0000 0000	uuuu uuuu
416h	PR4	Timer4 Period Register								1111 1111	uuuu uuuu
417h	T4CON	—	T4OUTPS<3:0>				TMR4ON	T4CKPS<1:0>		-000 0000	-000 0000
418h — 41Bh	—	Unimplemented								—	—
41Ch	TMR6	Holding Register for the 8-bit TMR6 Register								0000 0000	uuuu uuuu
41Dh	PR6	Timer6 Period Register								1111 1111	uuuu uuuu
41Eh	T6CON	—	T6OUTPS<3:0>				TMR6ON	T6CKPS<1:0>		-000 0000	-000 0000
41Fh	—	Unimplemented								—	—
Bank 9											
48Ch to 497h	—	Unimplemented								—	—
498h	NCO1ACCL	NCO1ACC								0000 0000	0000 0000
499h	NCO1ACCH	NCO1ACC								0000 0000	0000 0000
49Ah	NCO1ACCU	NCO1ACC								---- 0000	---- 0000
49Bh	NCO1INCL	NCO1INC								0000 0001	0000 0001
49Ch	NCO1INCH	NCO1INC								0000 0000	0000 0000
49Dh	NCO1INCU	NCO1INC								---- 0000	---- 0000
49Eh	NCO1CON	N1EN	—	N1OUT	N1POL	—	—	—	N1PFM	0-00 ---0	0-00 ---0
49Fh	NCO1CLK	N1PWS<2:0>			—	—	—	N1CKS<1:0>		000- --00	000- --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 on PIC16(L)F1718.
2: Unimplemented on PIC16LF1717/8/9

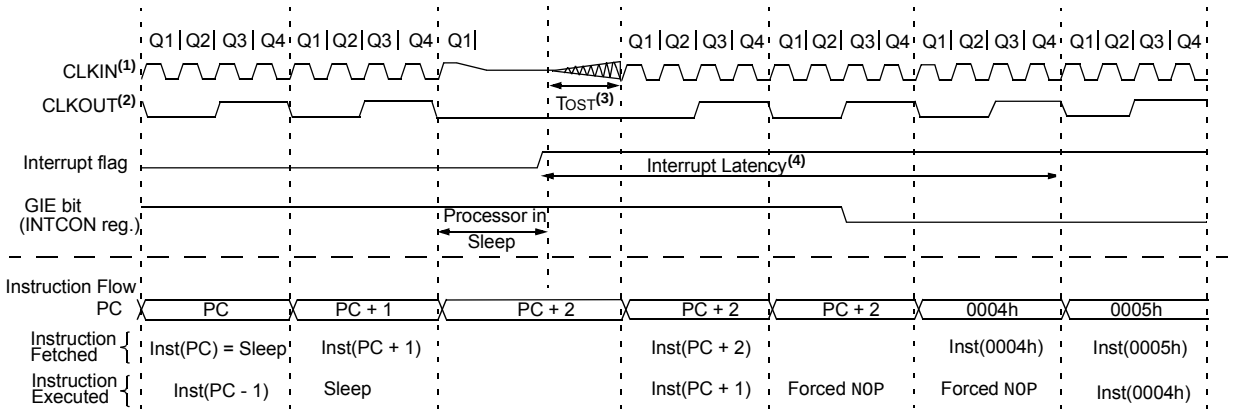
8.1.1 WAKE-UP USING INTERRUPTS

When global interrupts are disabled (GIE cleared) and any interrupt source has both its interrupt enable bit and interrupt flag bit set, one of the following will occur:

- If the interrupt occurs **before** the execution of a SLEEP instruction
  - SLEEP instruction will execute as a NOP
  - WDT and WDT prescaler will not be cleared
  - $\overline{TO}$ bit of the STATUS register will not be set
  - $\overline{PD}$ bit of the STATUS register will not be cleared
- If the interrupt occurs **during or after** the execution of a SLEEP instruction
  - SLEEP instruction will be completely executed
  - Device will immediately wake-up from Sleep
  - WDT and WDT prescaler will be cleared
  - $\overline{TO}$ bit of the STATUS register will be set
  - $\overline{PD}$ bit of the STATUS register will be cleared

Even if the flag bits were checked before executing a SLEEP instruction, it may be possible for flag bits to become set before the SLEEP instruction completes. To determine whether a SLEEP instruction executed, test the $\overline{PD}$ bit. If the $\overline{PD}$ bit is set, the SLEEP instruction was executed as a NOP.

FIGURE 8-1: WAKE-UP FROM SLEEP THROUGH INTERRUPT



Note 1: External clock. High, Medium, Low mode assumed.
2: CLKOUT is shown here for timing reference.
3: TOST = 1024 TOSC. This delay does not apply to EC, RC and INTOSC Oscillator modes or Two-Speed Start-up (see **Section 6.4 "Two-Speed Clock Start-up Mode"**.
4: GIE = 1 assumed. In this case after wake-up, the processor calls the ISR at 0004h. If GIE = 0, execution will continue in-line.

9.6 Register Definitions: Watchdog Control

REGISTER 9-1: WDTCON: WATCHDOG TIMER CONTROL REGISTER

U-0	U-0	R/W-0/0	R/W-1/1	R/W-0/0	R/W-1/1	R/W-1/1	R/W-0/0
—	—	WDTPS<4:0>(1)					SWDTEN
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	-m/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

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

bit 5-1 **WDTPS<4:0>:** Watchdog Timer Period Select bits(1)

Bit Value = Prescale Rate

11111 = Reserved. Results in minimum interval (1:32)

•

•

•

10011 = Reserved. Results in minimum interval (1:32)

10010 = 1:8388608 (2^{23}) (Interval 256s nominal)
10001 = 1:4194304 (2^{22}) (Interval 128s nominal)
10000 = 1:2097152 (2^{21}) (Interval 64s nominal)
01111 = 1:1048576 (2^{20}) (Interval 32s nominal)
01110 = 1:524288 (2^{19}) (Interval 16s nominal)
01101 = 1:262144 (2^{18}) (Interval 8s nominal)
01100 = 1:131072 (2^{17}) (Interval 4s nominal)
01011 = 1:65536 (Interval 2s nominal) (Reset value)
01010 = 1:32768 (Interval 1s nominal)
01001 = 1:16384 (Interval 512 ms nominal)
01000 = 1:8192 (Interval 256 ms nominal)
00111 = 1:4096 (Interval 128 ms nominal)
00110 = 1:2048 (Interval 64 ms nominal)
00101 = 1:1024 (Interval 32 ms nominal)
00100 = 1:512 (Interval 16 ms nominal)
00011 = 1:256 (Interval 8 ms nominal)
00010 = 1:128 (Interval 4 ms nominal)
00001 = 1:64 (Interval 2 ms nominal)
00000 = 1:32 (Interval 1 ms nominal)

bit 0 **SWDTEN:** Software Enable/Disable for Watchdog Timer bit

If WDTE<1:0> = 1x:
This bit is ignored.
If WDTE<1:0> = 01:
1 = WDT is turned on
0 = WDT is turned off
If WDTE<1:0> = 00:
This bit is ignored.

Note 1: Times are approximate. WDT time is based on 31 kHz LFINTOSC.

EXAMPLE 10-3: WRITING TO FLASH PROGRAM MEMORY

```
; This write routine assumes the following:
; 1. 64 bytes of data are loaded, starting at the address in DATA_ADDR
; 2. Each word of data to be written is made up of two adjacent bytes in DATA_ADDR,
; stored in little endian format
; 3. A valid starting address (the least significant bits = 00000) is loaded in ADDRH:ADDRL
; 4. ADDRH and ADDRL are located in shared data memory 0x70 - 0x7F (common RAM)
;
        BCF         INTCON,GIE      ; Disable ints so required sequences will execute properly
        BANKSEL     PMADRH          ; Bank 3
        MOVF        ADDRH,W         ; Load initial address
        MOVWF       PMADRH          ;
        MOVF        ADDRL,W         ;
        MOVWF       PMADRL          ;
        MOVLW       LOW DATA_ADDR   ; Load initial data address
        MOVWF       FSR0L           ;
        MOVLW       HIGH DATA_ADDR  ; Load initial data address
        MOVWF       FSR0H           ;
        BCF         PMCON1,CFGS     ; Not configuration space
        BSF         PMCON1,WREN     ; Enable writes
        BSF         PMCON1,LWLO     ; Only Load Write Latches
LOOP
        MOVIW       FSR0++          ; Load first data byte into lower
        MOVWF       PMDATL          ;
        MOVIW       FSR0++          ; Load second data byte into upper
        MOVWF       PMDATH          ;

        MOVF        PMADRL,W        ; Check if lower bits of address are '00000'
        XORLW       0x1F            ; Check if we're on the last of 32 addresses
        ANDLW       0x1F            ;
        BTFSC       STATUS,Z        ; Exit if last of 32 words,
        GOTO        START_WRITE     ;

  Required Sequence
        MOVLW       55h             ; Start of required write sequence:
        MOVWF       PMCON2          ; Write 55h
        MOVLW       0AAh            ;
        MOVWF       PMCON2          ; Write AAh
        BSF         PMCON1,WR       ; Set WR bit to begin write
        NOP                         ; NOP instructions are forced as processor
                                    ; loads program memory write latches
        NOP                         ;

        INCF        PMADRL,F        ; Still loading latches Increment address
        GOTO        LOOP            ; Write next latches

START_WRITE
        BCF         PMCON1,LWLO     ; No more loading latches - Actually start Flash program
                                    ; memory write

  Required Sequence
        MOVLW       55h             ; Start of required write sequence:
        MOVWF       PMCON2          ; Write 55h
        MOVLW       0AAh            ;
        MOVWF       PMCON2          ; Write AAh
        BSF         PMCON1,WR       ; Set WR bit to begin write
        NOP                         ; NOP instructions are forced as processor writes
                                    ; all the program memory write latches simultaneously
        NOP                         ; to program memory.
                                    ; After NOPs, the processor
                                    ; stalls until the self-write process in complete
                                    ; after write processor continues with 3rd instruction
        BCF         PMCON1,WREN     ; Disable writes
        BSF         INTCON,GIE      ; Enable interrupts
```

REGISTER 11-3: LATA: PORTA 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
LATA7	LATA6	LATA5	LATA4	LATA3	LATA2	LATA1	LATA0
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 **LATA<7:0>:** RA<7:0> Output Latch Value bits(1)

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

REGISTER 11-4: ANSELA: PORTA ANALOG SELECT REGISTER

U-0	U-0	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
—	—	ANSA5	ANSA4	ANSA3	ANSA2	ANSA1	ANSA0
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 **Unimplemented:** Read as ‘0’

bit 5-0 **ANSA<5:0>**: Analog Select between Analog or Digital Function on Pins RA<5:0>, respectively
1 = Analog input. Pin is assigned as analog input(1). Digital input buffer disabled.
0 = Digital I/O. Pin is assigned to port or digital special function.

Note 1: When setting a pin to an analog input, the corresponding TRIS bit must be set to Input mode in order to allow external control of the voltage on the pin.

15.4 ADC Acquisition Time

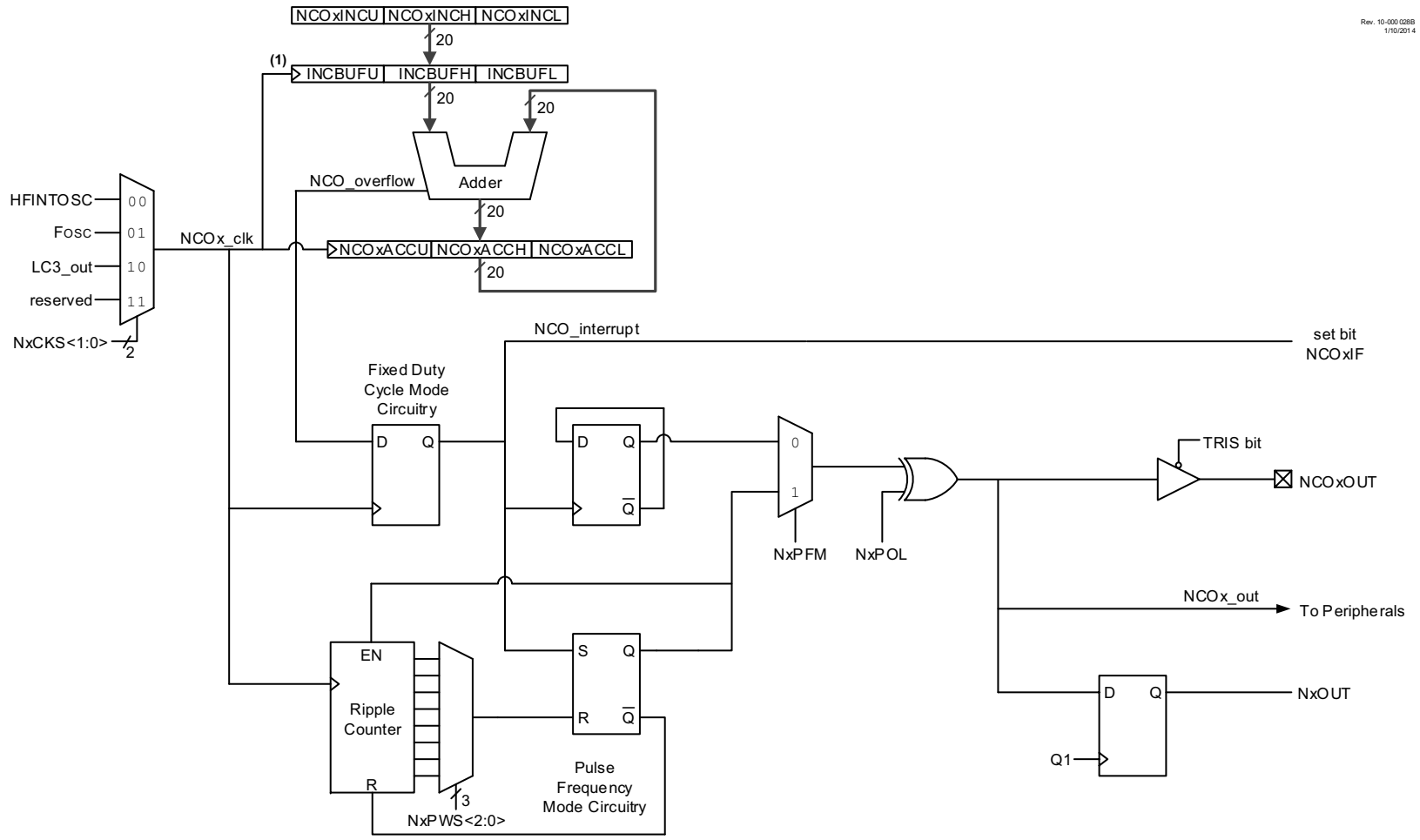
To ensure accurate temperature measurements, the user must wait at least 200 μs after the ADC input multiplexer is connected to the temperature indicator output before the conversion is performed. In addition, the user must wait 200 μs between sequential conversions of the temperature indicator output.

TABLE 15-2: SUMMARY OF REGISTERS ASSOCIATED WITH THE TEMPERATURE INDICATOR

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDFVR<1:0>		ADFVR<1:0>		165

Legend: Shaded cells are unused by the temperature indicator module.

FIGURE 20-1: NUMERICALLY CONTROLLED OSCILLATOR (NCOx) MODULE SIMPLIFIED BLOCK DIAGRAM



Note 1: The increment registers are double-buffered to allow for value changes to be made without first disabling the NCO module. The full increment value is loaded into the buffer registers on the second rising edge of the NCOx_clk signal that occurs immediately after a write to NCOxINCL register. The buffers are not user-accessible and are shown here for reference.

REGISTER 21-5: ADRESL: ADC RESULT REGISTER LOW (ADRESL) ADFM = 0

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
ADRES<1:0>		—	—	—	—	—	—
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 7-6 **ADRES<1:0>**: ADC Result Register bits
Lower two bits of 10-bit conversion result

bit 5-0 **Reserved**: Do not use.

REGISTER 21-6: ADRESH: ADC RESULT REGISTER HIGH (ADRESH) ADFM = 1

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
—	—	—	—	—	—	ADRES<9:8>	
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 7-2 **Reserved**: Do not use.

bit 1-0 **ADRES<9:8>**: ADC Result Register bits
Upper two bits of 10-bit conversion result

REGISTER 21-7: ADRESL: ADC RESULT REGISTER LOW (ADRESL) ADFM = 1

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
ADRES<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 **ADRES<7:0>**: ADC Result Register bits
Lower eight bits of 10-bit conversion result

26.1.3 SOFTWARE PROGRAMMABLE PRESCALER

A software programmable prescaler is available for exclusive use with Timer0. The prescaler is enabled by clearing the PSA bit of the OPTION_REG register.

Note:	The Watchdog Timer (WDT) uses its own independent prescaler.

There are eight prescaler options for the Timer0 module ranging from 1:2 to 1:256. The prescale values are selectable via the PS<2:0> bits of the OPTION_REG register. In order to have a 1:1 prescaler value for the Timer0 module, the prescaler must be disabled by setting the PSA bit of the OPTION_REG register.

The prescaler is not readable or writable. All instructions writing to the TMR0 register will clear the prescaler.

26.1.4 TIMER0 INTERRUPT

Timer0 will generate an interrupt when the TMR0 register overflows from FFh to 00h. The TMR0IF interrupt flag bit of the INTCON register is set every time the TMR0 register overflows, regardless of whether or not the Timer0 interrupt is enabled. The TMR0IF bit can only be cleared in software. The Timer0 interrupt enable is the TMR0IE bit of the INTCON register.

Note:	The Timer0 interrupt cannot wake the processor from Sleep since the timer is frozen during Sleep.

26.1.5 8-BIT COUNTER MODE SYNCHRONIZATION

When in 8-Bit Counter mode, the incrementing edge on the T0CKI pin must be synchronized to the instruction clock. Synchronization can be accomplished by sampling the prescaler output on the Q2 and Q4 cycles of the instruction clock. The high and low periods of the external clocking source must meet the timing requirements as shown in Table 34-12: Timer0 and Timer1 External Clock Requirements.

26.1.6 OPERATION DURING SLEEP

Timer0 cannot operate while the processor is in Sleep mode. The contents of the TMR0 register will remain unchanged while the processor is in Sleep mode.

FIGURE 27-6: TIMER1 GATE SINGLE-PULSE AND TOGGLE COMBINED MODE

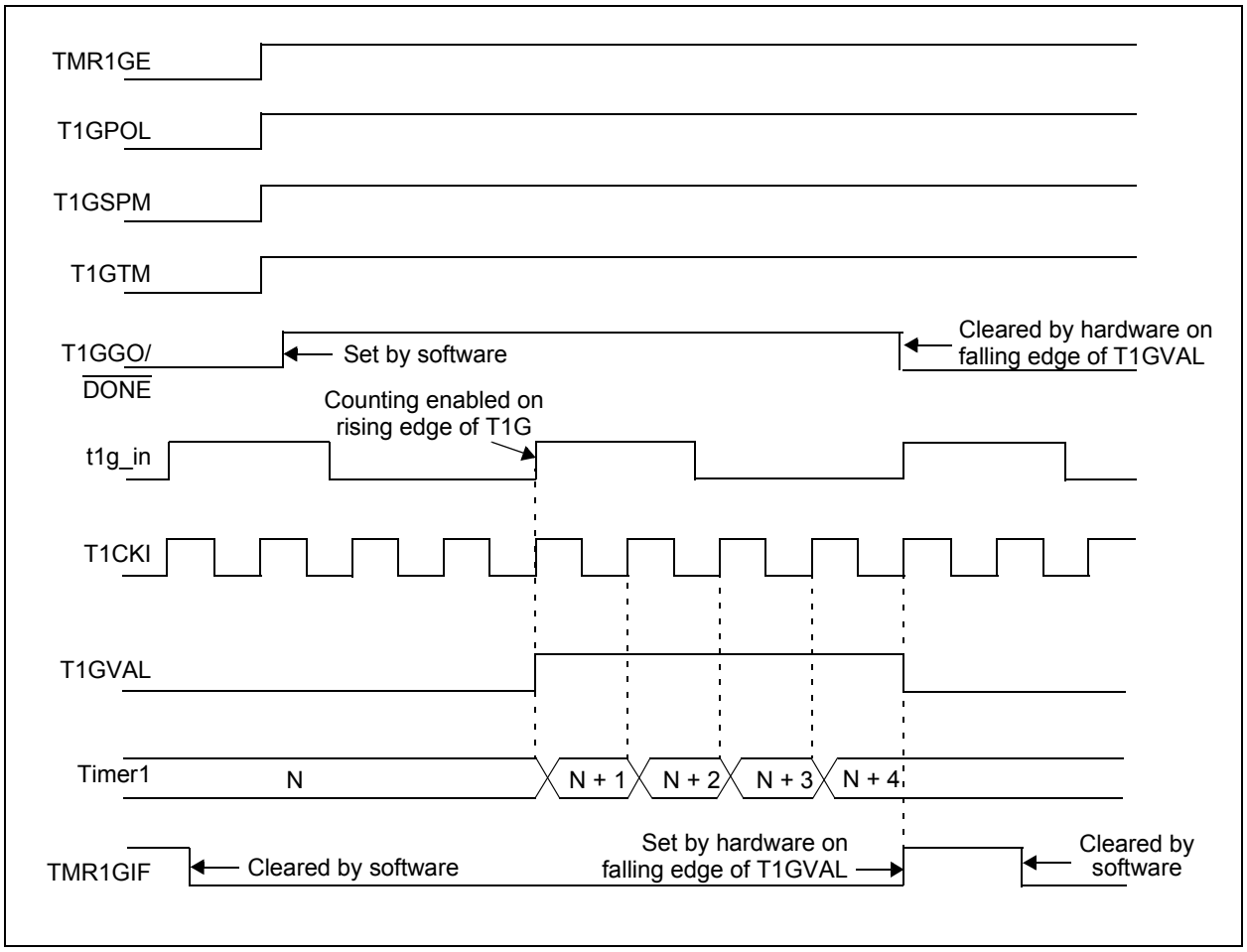


TABLE 28-1: SUMMARY OF REGISTERS ASSOCIATED WITH TIMER2

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
CCP2CON	—	—	DC2B<1:0>		CCP2M<3:0>				294
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	90
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	91
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	94
PR2	Timer2 Module Period Register								282*
T2CON	—	T2OUTPS<3:0>				TMR2ON	T2CKPS<1:0>		284
TMR2	Holding Register for the 8-bit TMR2 Register								282*

Legend: — = unimplemented location, read as '0'. Shaded cells are not used for Timer2 module.

* Page provides register information.

FIGURE 30-34: BUS COLLISION DURING START CONDITION (SCL = 0)

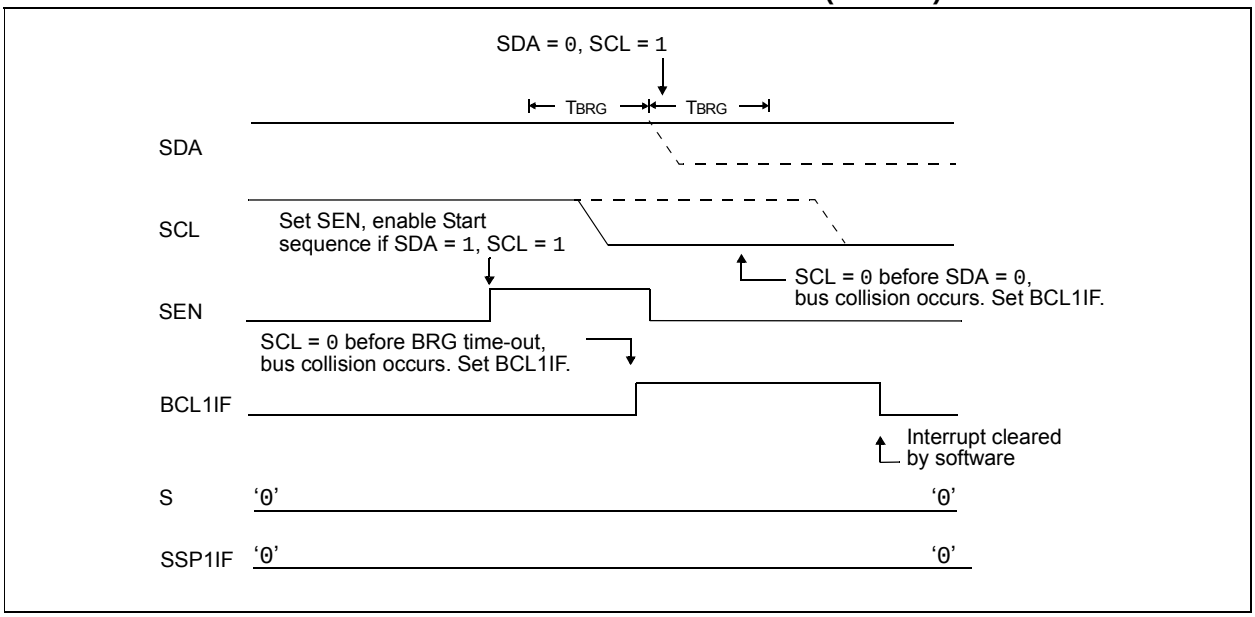
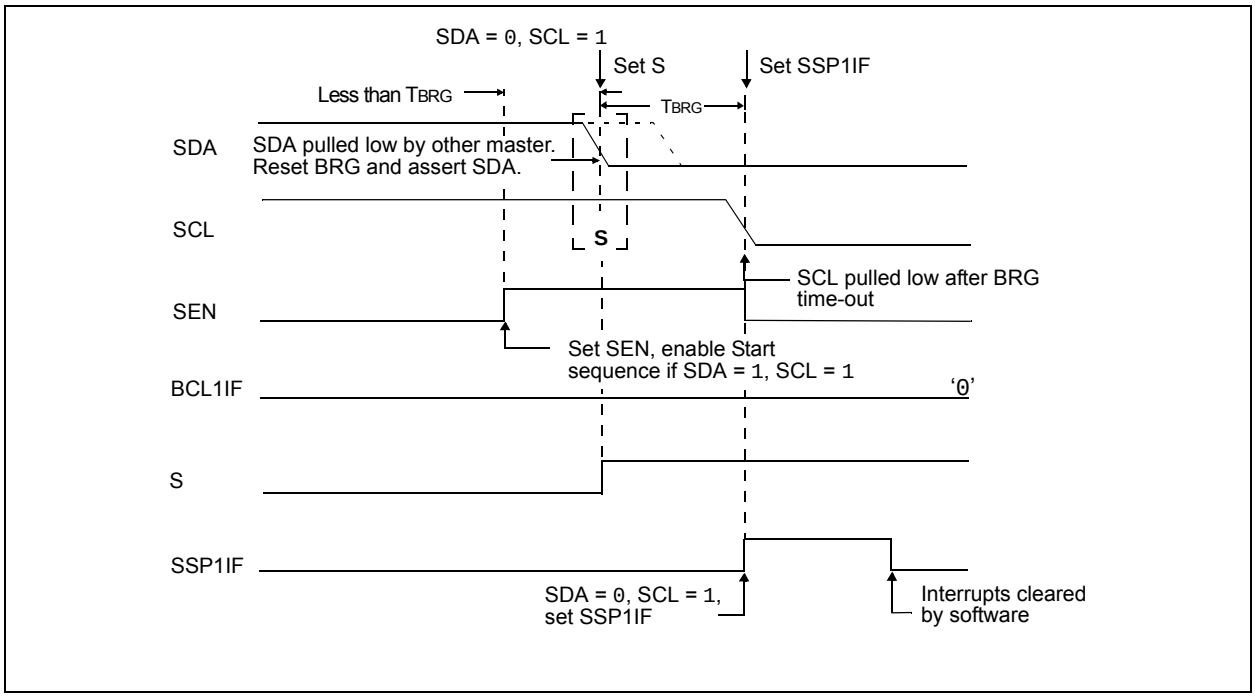


FIGURE 30-35: BRG RESET DUE TO SDA ARBITRATION DURING START CONDITION



31.4.2 AUTO-BAUD OVERFLOW

During the course of automatic baud detection, the ABDOVF bit of the BAUD1CON register will be set if the baud rate counter overflows before the fifth rising edge is detected on the RX pin. The ABDOVF bit indicates that the counter has exceeded the maximum count that can fit in the 16 bits of the SPxBRGH:SPxBRGL register pair. The overflow condition will set the RCIF flag. The counter continues to count until the fifth rising edge is detected on the RX pin. The RCIDL bit will remain false ('0') until the fifth rising edge at which time the RCIDL bit will be set. If the RC1REG is read after the overflow occurs, but before the fifth rising edge, the fifth rising edge will set the RCIF again.

Terminating the auto-baud process early to clear an overflow condition will prevent proper detection of the sync character's fifth rising edge. If any falling edges of the sync character have not yet occurred when the ABDEN bit is cleared, those will be falsely detected as start bits. The following steps are recommended to clear the overflow condition:

1. Read RCREG to clear RCIF.
2. If RCIDL is zero then wait for RCIF and repeat step 1.
3. Clear the ABDOVF bit.

31.4.3 AUTO-WAKE-UP ON BREAK

During Sleep mode, all clocks to the EUSART are suspended. Because of this, the Baud Rate Generator is inactive and a proper character reception cannot be performed. The Auto-Wake-up feature allows the controller to wake-up due to activity on the RX/DT line. This feature is available only in Asynchronous mode.

The Auto-Wake-up feature is enabled by setting the WUE bit of the BAUD1CON register. Once set, the normal receive sequence on RX/DT is disabled, and the EUSART remains in an Idle state, monitoring for a wake-up event independent of the CPU mode. A wake-up event consists of a high-to-low transition on the RX/DT line. (This coincides with the start of a Sync Break or a wake-up signal character for the LIN protocol.)

The EUSART module generates an RCIF interrupt coincident with the wake-up event. The interrupt is generated synchronously to the Q clocks in normal CPU operating modes (Figure 31-7), and asynchronously if the device is in Sleep mode (Figure 31-8). The interrupt condition is cleared by reading the RC1REG register.

The WUE bit is automatically cleared by the low-to-high transition on the RX line at the end of the Break. This signals to the user that the Break event is over. At this point, the EUSART module is in Idle mode waiting to receive the next character.

31.4.3.1 Special Considerations

Break Character

To avoid character errors or character fragments during a wake-up event, the wake-up character must be all zeros.

When the wake-up is enabled the function works independent of the low time on the data stream. If the WUE bit is set and a valid non-zero character is received, the low time from the Start bit to the first rising edge will be interpreted as the wake-up event. The remaining bits in the character will be received as a fragmented character and subsequent characters can result in framing or overrun errors.

Therefore, the initial character in the transmission must be all '0's. This must be ten or more bit times, 13-bit times recommended for LIN bus, or any number of bit times for standard RS-232 devices.

Oscillator Start-up Time

Oscillator start-up time must be considered, especially in applications using oscillators with longer start-up intervals (i.e., LP, XT or HS/PLL mode). The Sync Break (or wake-up signal) character must be of sufficient length, and be followed by a sufficient interval, to allow enough time for the selected oscillator to start and provide proper initialization of the EUSART.

WUE Bit

The wake-up event causes a receive interrupt by setting the RCIF bit. The WUE bit is cleared in hardware by a rising edge on RX/DT. The interrupt condition is then cleared in software by reading the RC1REG register and discarding its contents.

To ensure that no actual data is lost, check the RCIDL bit to verify that a receive operation is not in process before setting the WUE bit. If a receive operation is not occurring, the WUE bit may then be set just prior to entering the Sleep mode.

CALL **Call Subroutine**

Syntax:	[*label*] CALL k
Operands:	0 ≤ k ≤ 2047
Operation:	(PC)+ 1→ TOS, k → PC<10:0>, (PCLATH<6:3>) → 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.

CALLW **Subroutine Call With W**

Syntax:	[*label*] CALLW
Operands:	None
Operation:	(PC) +1 → TOS, (W) → PC<7:0>, (PCLATH<6:0>) → 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.

CLRF **Clear f**

Syntax:	[*label*] CLRF f
Operands:	0 ≤ f ≤ 127
Operation:	00h → (f) 1 → Z
Status Affected:	Z
Description:	The contents of register ‘f’ are cleared and the Z bit is set.

CLRW **Clear W**

Syntax:	[*label*] CLRW
Operands:	None
Operation:	00h → (W) 1 → Z
Status Affected:	Z
Description:	W register is cleared. Zero bit (Z) is set.

CLRWDT **Clear Watchdog Timer**

Syntax:	[*label*] CLRWDT
Operands:	None
Operation:	00h → WDT 0 → WDT prescaler, 1 → $\overline{TO}$ 1 → $\overline{PD}$
Status Affected:	$\overline{TO}$, $\overline{PD}$
Description:	CLRWDT instruction resets the Watchdog Timer. It also resets the prescaler of the WDT. Status bits $\overline{TO}$ and $\overline{PD}$ are set.

COMF **Complement f**

Syntax:	[*label*] COMF f,d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	($\bar{f}$) → (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’.

DECF **Decrement f**

Syntax:	[*label*] DECF f,d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	(f) - 1 → (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’.

TABLE 34-9: PLL CLOCK TIMING SPECIFICATIONS

Standard Operating Conditions (unless otherwise stated)							
Param. No.	**Sym.**	**Characteristic**	**Min.**	**Typ.†**	**Max.**	**Units**	**Conditions**
F10	FOSC	Oscillator Frequency Range	4	—	8	MHz	
F11	FSYS	On-Chip VCO System Frequency	16	—	32	MHz	
F12	TRC	PLL Start-up Time (Lock Time)	—	—	2	ms	
F13*	ΔCLK	CLKOUT Stability (Jitter)	-0.25%	—	+0.25%	%	

* These parameters are characterized but not tested.

† Data in “Typ” column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

FIGURE 34-7: CLKOUT AND I/O TIMING

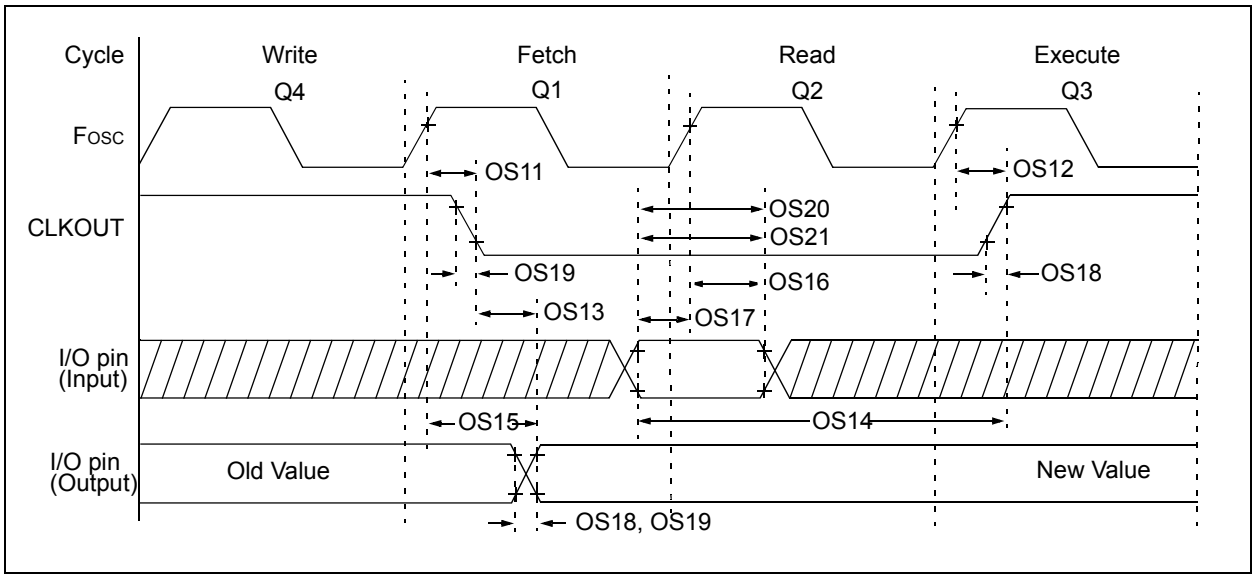


TABLE 34-10: CLKOUT AND I/O TIMING PARAMETERS

Standard Operating Conditions (unless otherwise stated)							
Param. No.	**Sym.**	**Characteristic**	**Min.**	**Typ.†**	**Max.**	**Units**	**Conditions**
OS11	TOSH2CKL	FOSC↑ to CLKOUT↓ (1)	—	—	70	ns	3.3V ≤ VDD ≤ 5.0V
OS12	TOSH2CKH	FOSC↑ to CLKOUT↑ (1)	—	—	72	ns	3.3V ≤ VDD ≤ 5.0V
OS13	TCKL2IOV	CLKOUT↓ to Port out valid(1)	—	—	20	ns	
OS14	TIOV2CKH	Port input valid before CLKOUT↑(1)	TOSC + 200 ns	—	—	ns	
OS15	TOSH2IOV	FOSC↑ (Q1 cycle) to Port out valid	—	50	70*	ns	3.3V ≤ VDD ≤ 5.0V
OS16	TOSH2IOI	FOSC↑ (Q2 cycle) to Port input invalid (I/O in hold time)	50	—	—	ns	3.3V ≤ VDD ≤ 5.0V
OS17	TIOV2OSH	Port input valid to FOSC↑ (Q2 cycle) (I/O in setup time)	20	—	—	ns	
OS18*	TIOR	Port output rise time(2)	— —	40 15	72 32	ns	VDD = 1.8V 3.3V ≤ VDD ≤ 5.0V
OS19*	TIOF	Port output fall time(2)	— —	28 15	55 30	ns	VDD = 1.8V 3.3V ≤ VDD ≤ 5.0V
OS20*	TINP	INT pin input high or low time	25	—	—	ns	
OS21*	TIOC	Interrupt-on-Change new input level time	25	—	—	ns	

* These parameters are characterized but not tested.

† Data in "Typ" column is at 3.0V, 25°C unless otherwise stated.

Note 1: Measurements are taken in EXTRC mode where CLKOUT output is 4 x TOSC.

2: Slew rate limited.

Note: Unless otherwise noted, VIN = 5V, FOSC = 500 kHz, CIN = 0.1 µF, TA = 25°C.

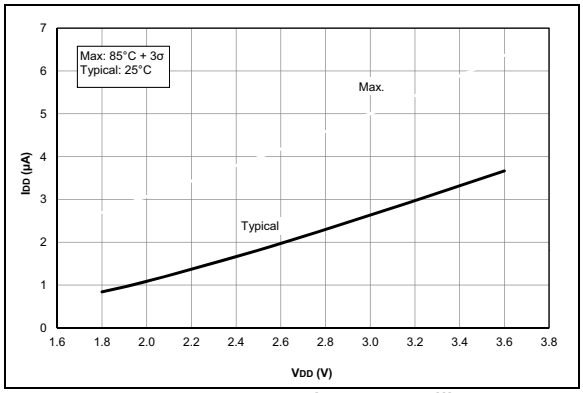


FIGURE 35-43: *IPD, Timer1 Oscillator, FOSC = 32 kHz, PIC16LF1717/8/9 Only.*

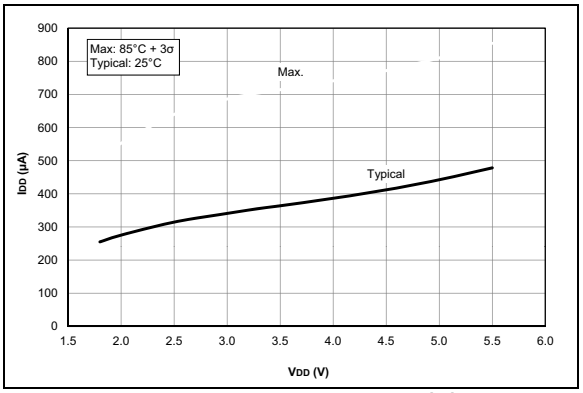


FIGURE 35-46: *IPD, Op Amp, High GBWP Mode (OPAxSP = 1), PIC16F1717/8/9 Only.*

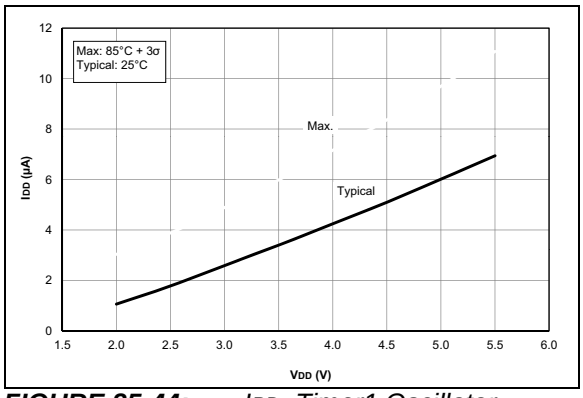


FIGURE 35-44: *IPD, Timer1 Oscillator, FOSC = 32 kHz, PIC16F1717/8/9 Only.*

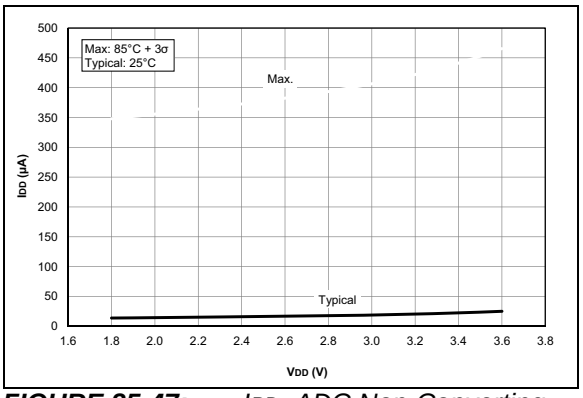


FIGURE 35-47: *IPD, ADC Non-Converting, PIC16LF1717/8/9 Only.*

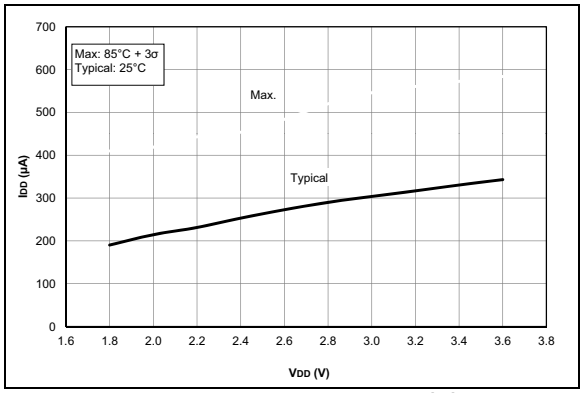


FIGURE 35-45: *IPD, Op Amp, High GBWP Mode (OPAxSP = 1), PIC16LF1717/8/9 Only.*

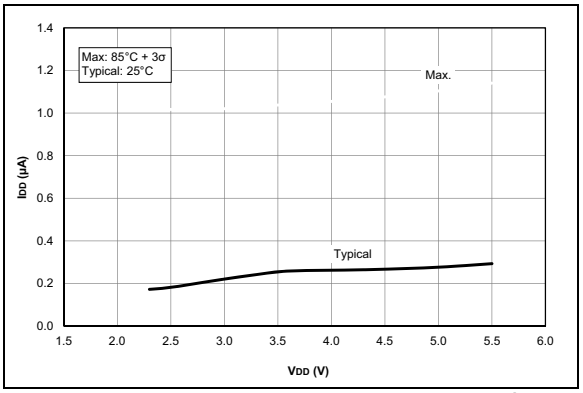


FIGURE 35-48: *IPD, ADC Non-Converting, PIC16F1717/8/9 Only.*