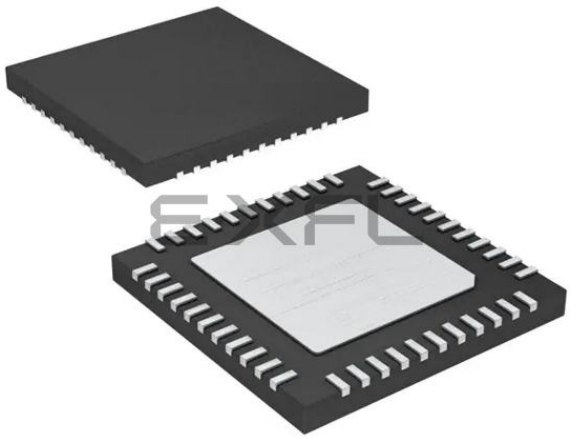


Welcome to E-XFL.COM

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	36
Program Memory Size	14KB (8K x 14)
Program Memory Type	FLASH
EEPROM Size	128 x 8
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 28x10b; D/A 4x5b, 4x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-VQFN Exposed Pad
Supplier Device Package	44-QFN (8x8)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf1777-i-ml

TABLE 1-3: PIC16(L)F1777/9 PINOUT DESCRIPTION (CONTINUED)

Name	Function	Input Type	Output Type	Description
RB2/AN8/OPA2IN0-/ DAC3OUT1/PRG4F/COG3IN/ MD4MOD	RB2	TTL/ST	CMOS	General purpose I/O.
	AN8	AN	—	ADC Channel 8 input.
	OPA2IN0-	AN	—	Operational amplifier 2 inverting input.
	DAC3OUT1	—	AN	DAC3 voltage output.
	PRG4F(1)	TTL/ST	—	Ramp generator set_falling input.
	COG3IN(1)	TTL/ST	—	Complementary output generator 3 input.
	MD4MOD(1)	TTL/ST	—	Data signal modulator modulation input.
RB3/AN9/C1IN2-/C2IN2-/ C3IN2-/OPA2IN0+/MD3CL	RB3	TTL/ST	CMOS	General purpose I/O.
	AN9	AN	—	ADC Channel 9 input.
	C1IN2-	AN	—	Comparator 1 negative input.
	C2IN2-	AN	—	Comparator 2 negative input.
	C3IN2-	AN	—	Comparator 3 negative input.
	OPA2IN0+	AN		Operational amplifier 2 non-inverting input.
	MD3CL(1)	TTL/ST	—	Data signal modulator 3 low carrier input.
RB4/AN11/C3IN1+/MD3CH	RB4	TTL/ST	CMOS	General purpose I/O.
	AN11	AN	—	ADC Channel 11 input.
	C3IN1+	AN	—	Comparator 3 positive input.
	MD3CH(1)	TTL/ST	—	Data signal modulator 3 high carrier input.
RB5/AN13/DAC5REF1-/ DAC7REF1-/C4IN2-/CCP7/ MD3MOD	RB5	TTL/ST	CMOS	General purpose I/O.
	AN13	AN	—	ADC Channel 11 input.
	DAC5REF1-	AN	—	DAC5 negative reference.
	DAC7REF1-	AN	—	DAC7 negative reference.
	C4IN2-	AN	—	Comparator 4 negative input.
	CCP7(1)	TTL/ST	—	CCP7 capture input.
	MD3MOD(1)	TTL/ST	—	Data signal modulator modulation input.
RB6/DAC5REF1+/DAC7REF1+/ C4IN1+/CLCIN2/ICSPCLK	RB6	TTL/ST	CMOS	General purpose I/O.
	DAC5REF1+	AN	—	DAC5 positive reference.
	DAC7REF1+	AN	—	DAC7 positive reference.
	C4IN1+	AN	—	Comparator 2 positive input.
	CLCIN2(1)	TTL/ST	—	CLC input 2.
	ICSPCLK	ST	—	Serial Programming 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
HP = High Power 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)F1778 MEMORY MAP, BANK 16-23

BANK 16		BANK 17		BANK 18		BANK 19		BANK 20		BANK 21		BANK 22		BANK 23	
800h	Core Registers (Table 3-2)	880h	Core Registers (Table 3-2)	900h	Core Registers (Table 3-2)	980h	Core Registers (Table 3-2)	A00h	Core Registers (Table 3-2)	A80h	Core Registers (Table 3-2)	B00h	Core Registers (Table 3-2)	B80h	Core Registers (Table 3-2)
80Bh		88Bh		90Bh		98Bh		A0Bh		A8Bh		B0Bh		B8Bh	
80Ch	—	88Ch	Unimplemented Read as '0'	90Ch	CM4CON0	98Ch	Unimplemented Read as '0'	A0Ch	Unimplemented Read as '0'	A8Ch	Unimplemented Read as '0'	B0Ch	Unimplemented Read as '0'	B8Ch	Unimplemented Read as '0'
80Dh	COG3PHR			90Dh	CM4CON1										
80Eh	COG3PHF			90Eh	CM4NSEL										
80Fh	COG3BLKR			90Fh	CM4PSEL										
810h	COG3BLKF			910h	CM5CON0										
811h	COG3DBR			911h	CM5CON1										
812h	COG3DBF			912h	CM5NSEL										
813h	COG3CON0			913h	CM5PSEL										
814h	COG3CON1			914h	CM6CON0										
815h	COG3RIS0			915h	CM6CON1										
816h	COG3RIS1			916h	CM6NSEL										
817h	COG3RSIM0			917h	CM6PSEL										
818h	COG3RSIM1			918h	Unimplemented Read as '0'										
819h	COG3FIS0														
81Ah	COG3FIS1														
81Bh	COG3FSIM0														
81Ch	COG3FSIM1														
81Dh	COG3ASD0														
81Eh	COG3ASD1														
81Fh	COG3STR	89Fh		91Fh		99Fh		A1Fh		A9Fh		B1Fh		B9Fh	
820h	General Purpose Register 48 Bytes	8A0h	General Purpose Register 48 Bytes	920h	General Purpose Register 48 Bytes	9A0h	General Purpose Register 48 Bytes	A20h	General Purpose Register 48 Bytes	AA0h	General Purpose Register 48 Bytes	B20h	General Purpose Register 48 Bytes	BA0h	General Purpose Register 48 Bytes
86Fh		8EFh		96Fh		9EFh		A6Fh		AEFh		B6Fh		BEFh	
870h	Accesses 70h – 7Fh	8F0h	Accesses 70h – 7Fh	970h	Accesses 70h – 7Fh	9F0h	Accesses 70h – 7Fh	A70h	Accesses 70h – 7Fh	AF0h	Accesses 70h – 7Fh	B70h	Accesses 70h – 7Fh	BF0h	Accesses 70h – 7Fh
87Fh		8FFh		97Fh		9FFh		A7Fh		AFFh		B7Fh		BFFh	

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

TABLE 3-18: 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 15											
78Ch — 78Dh	—	Unimplemented								—	—
78Eh	PRG1RTSS	—	—	—	—	RTSS<3:0>				---- 0000	---- 0000
78Fh	PRG1FTSS	—	—	—	—	FTSS<3:0>				---- 0000	---- 0000
790h	PRG1INS	—	—	—	—	INS<3:0>				---- 0000	---- 0000
791h	PRG1CON0	EN	—	FEDG	REDG	MODE<1:0>		OS	GO	0-000 0000	0-00 0000
792h	PRG1CON1	—	—	—	—	—	RDY	FPOL	RPOL	---- -000	---- -000
793h	PRG1CON2	—	—	—	ISET<4:0>					---0 0000	---0 0000
794h	PRG2RTSS	—	—	—	—	RTSS<3:0>				---- 0000	---- 0000
795h	PRG2FTSS	—	—	—	—	FTSS<3:0>				---- 0000	---- 0000
796h	PRG2INS	—	—	—	—	INS<3:0>				---- 0000	---- 0000
797h	PRG2CON0	EN	—	FEDG	REDG	MODE<1:0>		OS	GO	0-000 0000	0-00 0000
798h	PRG2CON1	—	—	—	—	—	RDY	FPOL	RPOL	---- -000	---- -000
799h	PRG2CON2	—	—	—	ISET<4:0>					---0 0000	---0 0000
79Ah	PRG3RTSS	—	—	—	—	RTSS<3:0>				---- 0000	---- 0000
79Bh	PRG3FTSS	—	—	—	—	FTSS<3:0>				---- 0000	---- 0000
79Ch	PRG3INS	—	—	—	—	INS<3:0>				---- 0000	---- 0000
79Dh	PRG3CON0	EN	—	FEDG	REDG	MODE<1:0>		OS	GO	0-000 0000	0-00 0000
79Eh	PRG3CON1	—	—	—	—	—	RDY	FPOL	RPOL	---- -000	---- -000
79Fh	PRG3CON2	—	—	—	ISET<4:0>					---0 0000	---0 0000

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 PIC16LF1777/8/9.
3: Unimplemented on PIC16(L)F1778.

TABLE 3-18: 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 16											
80Ch	—	Unimplemented								—	—
80Dh	COG3PHR	—	—	COG Rising Edge Phase Delay Count Register						--00 0000	--00 0000
80Eh	COG3PHF	—	—	COG Falling Edge Phase Delay Count Register						--00 0000	--00 0000
80Fh	COG3BLKR	—	—	COG Rising Edge Blanking Count Register						--00 0000	--00 0000
810h	COG3BLKF	—	—	COG Falling Edge Blanking Count Register						--00 0000	--00 0000
811h	COG3DBR	—	—	COG Rising Edge Dead-band Count Register						--00 0000	--00 0000
812h	COG3DBF	—	—	COG Falling Edge Dead-band Count Register						--00 0000	--00 0000
813h	COG3CON0	EN	LD	—	CS<1:0>		MD<2:0>			00-0 0000	00-0 0000
814h	COG3CON1	RDBS	FDBS	—	—	POLD	POLC	POLB	POLA	00-- 0000	00-- 0000
815h	COG3RIS0	RIS7	RIS6	RIS5	RIS4	RIS3	RIS2	RIS1	RIS0	0000 0000	0000 0000
816h	COG3RIS1	RIS15	RIS14	RIS13	RIS12	RIS11	RIS10	RIS9	RIS8	0000 0000	0000 0000
817h	COG3RSIM0	RSIM7	RSIM6	RSIM5	RSIM4	RSIM3	RSIM2	RSIM1	RSIM0	0000 0000	0000 0000
818h	COG3RSIM1	RSIM15	RSIM14	RSIM13	RSIM12	RSIM11	RSIM10	RSIM9	RSIM8	0000 0000	0000 0000
819h	COG3FIS0	FIS7	FIS6	FIS5	FIS4	FIS3	FIS2	FIS1	FIS0	0000 0000	0000 0000
81Ah	COG3FIS1	FIS15	FIS14	FIS13	FIS12	FIS11	FIS10	FIS9	FIS8	0000 0000	0000 0000
81Bh	COG3FSIM0	FSIM7	FSIM6	FSIM5	FSIM4	FSIM3	FSIM2	FSIM1	FSIM0	0000 0000	0000 0000
81Ch	COG3FSIM1	FSIM15	FSIM14	FSIM13	FSIM12	FSIM11	FSIM10	FSIM9	FSIM8	0000 0000	0000 0000
81Dh	COG3ASD0	ASE	ARSEN	ASDBD<1:0>		ASDAC<1:0>		—	—	0001 01--	0001 01--
81Eh	COG3ASD1	AS7E	AS6E	AS5E	AS4E	AS3E	AS2E	AS1E	AS0E	0000 0000	0000 0000
81Fh	COG3STR	SDATD	SDATC	SDATB	SDATA	STRD	STRC	STRB	STRA	0000 0000	0000 0000

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 PIC16LF1777/8/9.
3: Unimplemented on PIC16(L)F1778.

3.6 Stack

All devices have a 16-level x 15-bit wide hardware stack (refer to Figures 3-1 and 3-2). The stack space is not part of either program or data space. The PC is PUSHed onto the stack when CALL or CALLW instructions are executed or an interrupt causes a branch. The stack is POPed in the event of a RETURN, RETLW or a RETFIE instruction execution. PCLATH is not affected by a PUSH or POP operation.

The stack operates as a circular buffer if the STVREN bit is programmed to '0' (Configuration Words). This means that after the stack has been PUSHed sixteen times, the seventeenth PUSH overwrites the value that was stored from the first PUSH. The eighteenth PUSH overwrites the second PUSH (and so on). The STKOVF and STKUNF flag bits will be set on an Overflow/Underflow, regardless of whether the Reset is enabled.

Note:	There are no instructions/mnemonics called PUSH or POP. These are actions that occur from the execution of the CALL, CALLW, RETURN, RETLW and RETFIE instructions or the vectoring to an interrupt address.

3.6.1 ACCESSING THE STACK

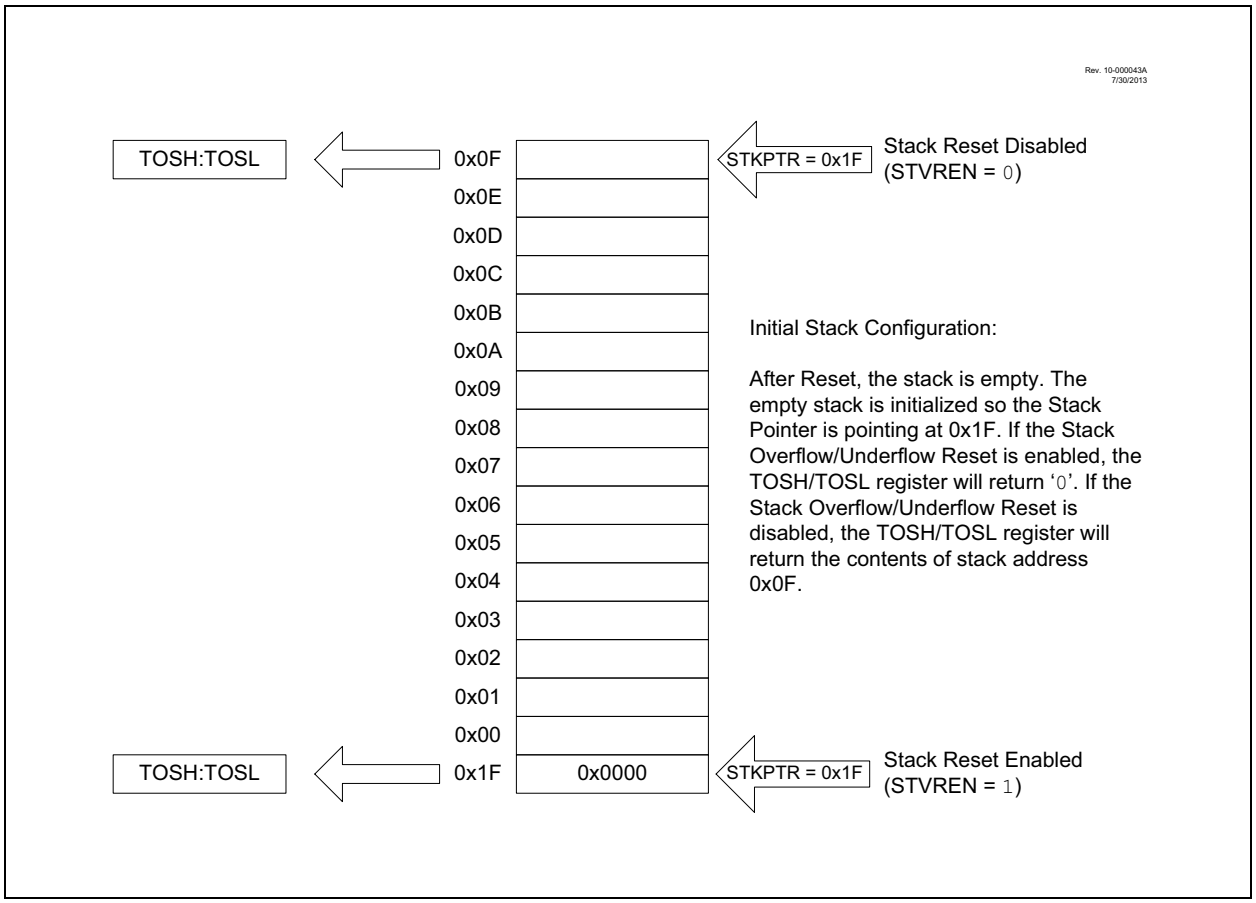
The stack is available through the TOSH, TOSL and STKPTR registers. STKPTR is the current value of the Stack Pointer. The TOSH:TOSL register pair points to the TOP of the stack. Both registers are read/writable. TOS is split into TOSH and TOSL due to the 15-bit size of the PC. To access the stack, adjust the value of STKPTR, which will position TOSH:TOSL, then read/write to TOSH:TOSL. STKPTR is five bits to allow detection of overflow and underflow.

Note:	Care should be taken when modifying the STKPTR while interrupts are enabled.

During normal program operation, CALL, CALLW and Interrupts will increment STKPTR while RETLW, RETURN, and RETFIE will decrement STKPTR. At any time, STKPTR can be inspected to see how much stack is left. The STKPTR always points at the currently used place on the stack. Therefore, a CALL or CALLW will increment the STKPTR and then write the PC, and a return will unload the PC and then decrement the STKPTR.

Reference Figure 3-5 through Figure 3-8 for examples of accessing the stack.

FIGURE 3-5: ACCESSING THE STACK EXAMPLE 1



4.3 Code Protection

Code protection allows the device to be protected from unauthorized access. Program memory protection is controlled independently. Internal access to the program memory is unaffected by any code protection setting.

4.3.1 PROGRAM MEMORY PROTECTION

The entire program memory space is protected from external reads and writes by the $\overline{CP}$ bit in Configuration Words. When $\overline{CP}$ = 0, external reads and writes of program memory are inhibited and a read will return all '0's. The CPU can continue to read program memory, regardless of the protection bit settings. Writing the program memory is dependent upon the write protection setting. See **Section 4.4 "Write Protection"** for more information.

4.4 Write Protection

Write protection allows the device to be protected from unintended self-writes. Applications, such as boot loader software, can be protected while allowing other regions of the program memory to be modified.

The WRT<1:0> bits in Configuration Words define the size of the program memory block that is protected.

4.5 User ID

Four memory locations (8000h-8003h) are designated as ID locations where the user can store checksum or other code identification numbers. These locations are readable and writable during normal execution. See **Section 10.4 "User ID, Device ID and Configuration Word Access"** for more information on accessing these memory locations. For more information on checksum calculation, see the *"PIC16(L)F170X Memory Programming Specification"* (DS41683).

5.2.2.7 Internal Oscillator Clock Switch Timing

When switching between the HFINTOSC, MFINTOSC and the LFINTOSC, the new oscillator may already be shut down to save power (see Figure 5-7). If this is the case, there is a delay after the IRCF<3:0> bits of the OSCCON register are modified before the frequency selection takes place. The OSCSTAT register will reflect the current active status of the HFINTOSC, MFINTOSC and LFINTOSC oscillators. The sequence of a frequency selection is as follows:

1. IRCF<3:0> bits of the OSCCON register are modified.
2. If the new clock is shut down, a clock start-up delay is started.
3. Clock switch circuitry waits for a falling edge of the current clock.
4. The current clock is held low and the clock switch circuitry waits for a rising edge in the new clock.
5. The new clock is now active.
6. The OSCSTAT register is updated as required.
7. Clock switch is complete.

See Figure 5-7 for more details.

If the internal oscillator speed is switched between two clocks of the same source, there is no start-up delay before the new frequency is selected. Clock switching time delays are shown in Table 5-1.

Start-up delay specifications are located in the oscillator tables of **Section 36.0 "Electrical Specifications"**.

TABLE 6-5: SUMMARY OF REGISTERS ASSOCIATED WITH RESETS

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
BORCON	SBOREN	BORFS	—	—	—	—	—	BORRDY	121
PCON	STKOVF	STKUNF	—	$\overline{\text{RWDT}}$	$\overline{\text{RMCLR}}$	$\overline{\text{RI}}$	$\overline{\text{POR}}$	$\overline{\text{BOR}}$	125
STATUS	—	—	—	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C	40
WDTCON	—	—	WDTPS<4:0>					SWDTEN	154

Legend: — = unimplemented location, read as '0'. Shaded cells are not used by Resets.

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

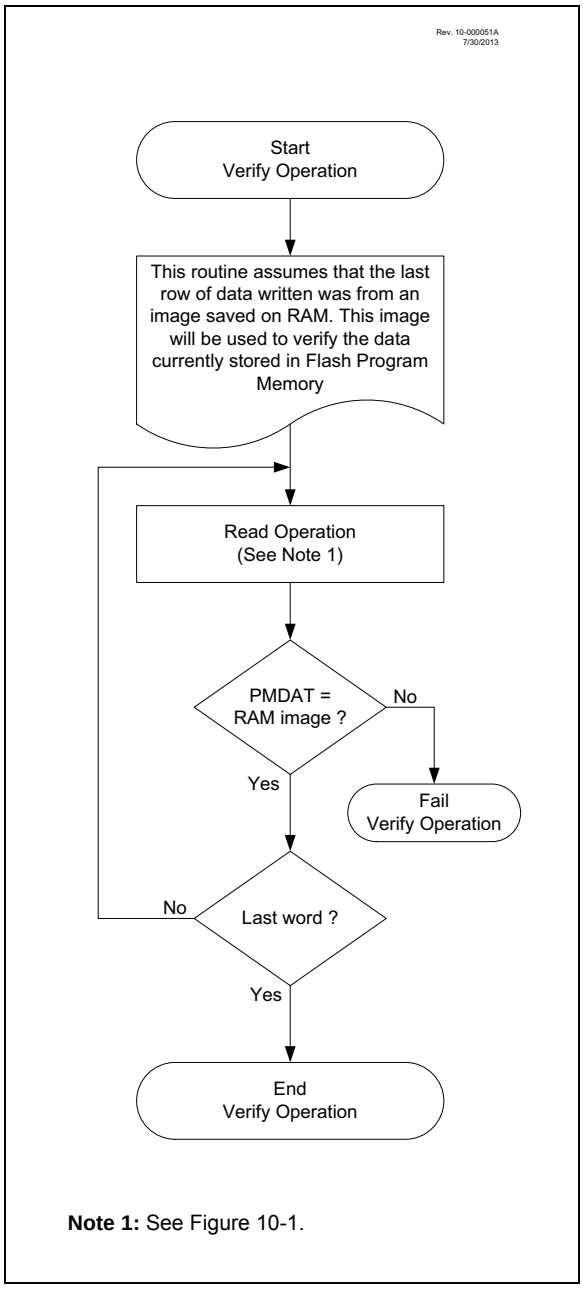
bit 0 **Reserved:** Read as ‘1’. Maintain this bit set.

Note 1: PIC16F1777/8/9 only.
2: See **Section 36.0 “Electrical Specifications”**.

10.5 Write/Verify

It is considered good programming practice to verify that program memory writes agree with the intended value. Since program memory is stored as a full page then the stored program memory contents are compared with the intended data stored in RAM after the last write is complete.

FIGURE 10-8: FLASH PROGRAM MEMORY VERIFY FLOWCHART



Note 1: See Figure 10-1.

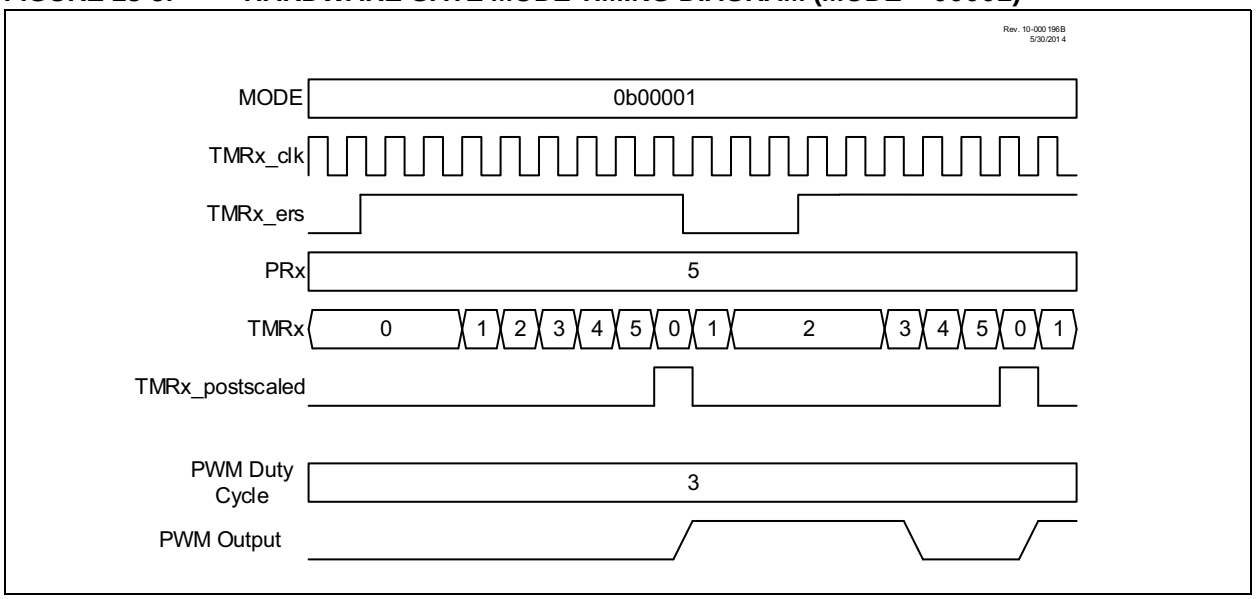
23.6.2 HARDWARE GATE MODE

The Hardware Gate modes operate the same as the Software Gate mode except the TMRx_ers external signal can also gate the timer. When used with the CCP the gating extends the PWM period. If the timer is stopped when the PWM output is high then the duty cycle is also extended.

When MODE<4:0> = 00001 then the timer is stopped when the external signal is high. When MODE<4:0> = 00010 then the timer is stopped when the external signal is low.

Figure 23-5 illustrates the Hardware Gating mode for MODE<4:0> = 00001 in which a high input level starts the counter.

FIGURE 23-5: HARDWARE GATE MODE TIMING DIAGRAM (MODE = 00001)



23.6.10 LEVEL-TRIGGERED HARDWARE LIMIT ONE-SHOT MODES

The Level-Triggered Hardware Limit One-Shot modes hold the timer in Reset on an external Reset level and start counting when both the ON bit is set and the external signal is not at the Reset level. If one of either the external signal is not in Reset or the ON bit is set then the other signal being set/made active will start the timer. Reset levels are selected as follows:

- Low Reset level (MODE<4:0> = `10110`)
- High Reset level (MODE<4:0> = `10111`)

When the timer count matches the PRx period count, the timer is reset and the ON bit is cleared. When the ON bit is cleared by either a PRx match or by software control the timer will stay in Reset until both the ON bit is set and the external signal is not at the Reset level.

When Level-Triggered Hardware Limit One-Shot modes are used in conjunction with the CCP PWM operation the PWM drive goes active with either the external signal edge or the setting of the ON bit, whichever of the two starts the timer.

REGISTER 23-4: TxRST: TIMERx EXTERNAL RESET SIGNAL SELECTION REGISTER

U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—	RSEL<4:0>				
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 7-5 **Unimplemented:** Read as '0'

bit 4-0 **RSEL<4:0>:** TimerX External Reset Signal Source Selection bits
See Table 23-4.

TABLE 23-4: EXTERNAL RESET SOURCES

RSEL<4:0>	Timer2	Timer4	Timer6	Timer8
11111	Reserved	Reserved	Reserved	Reserved
11110	Reserved	Reserved	Reserved	Reserved
11101	LC4_out	LC4_out	LC4_out	LC4_out
11100	LC3_out	LC3_out	LC3_out	LC3_out
11011	LC2_out	LC2_out	LC2_out	LC2_out
11010	LC1_out	LC1_out	LC1_out	LC1_out
11001	ZCD_out	ZCD_out	ZCD_out	ZCD_out
11000(1)	sync_C8OUT	sync_C8OUT	sync_C8OUT	sync_C8OUT
10111(1)	sync_C7OUT	sync_C7OUT	sync_C7OUT	sync_C7OUT
10110	sync_C6OUT	sync_C6OUT	sync_C6OUT	sync_C6OUT
10101	sync_C5OUT	sync_C5OUT	sync_C5OUT	sync_C5OUT
10100	sync_C4OUT	sync_C4OUT	sync_C4OUT	sync_C4OUT
10011	sync_C3OUT	sync_C3OUT	sync_C3OUT	sync_C3OUT
10010	sync_C2OUT	sync_C2OUT	sync_C2OUT	sync_C2OUT
10001	sync_C1OUT	sync_C1OUT	sync_C1OUT	sync_C1OUT
10000(1)	PWM12_out	PWM12_out	PWM12_out	PWM12_out
01111	PWM11_out	PWM11_out	PWM11_out	PWM11_out
01110	PWM6_out	PWM6_out	PWM6_out	PWM6_out
01101	PWM5_out	PWM5_out	PWM5_out	PWM5_out
01100(1)	PWM10_out	PWM10_out	PWM10_out	PWM10_out
01011	PWM9_out	PWM9_out	PWM9_out	PWM9_out
01010	PWM4_out	PWM4_out	PWM4_out	PWM4_out
01001	PWM3_out	PWM3_out	PWM3_out	PWM3_out
01000(1)	CCP8_out	CCP8_out	CCP8_out	CCP8_out
00111	CCP7_out	CCP7_out	CCP7_out	CCP7_out
00110	CCP2_out	CCP2_out	CCP2_out	CCP2_out
00101	CCP1_out	CCP1_out	CCP1_out	CCP1_out
00100	TMR8_postscaled	TMR8_postscaled	TMR8_postscaled	Reserved
00011	TMR6_postscaled	TMR6_postscaled	Reserved	TMR6_postscaled
00010	TMR4_postscaled	Reserved	TMR4_postscaled	TMR4_postscaled
00001	Reserved	TMR2_postscaled	TMR2_postscaled	TMR2_postscaled
00000	Pin selected byT2INPPS	Pin selected by T4INPPS	Pin selected by T6INPPS	Pin selected by T6INPPS

Note 1: PIC16LF1777/9 only.

24.5 Register Definitions: CCP Control

REGISTER 24-1: CCPxCON: CCPx CONTROL REGISTER

R/W-0/0	U-0	R-x	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
EN	—	OUT	FMT	MODE<3:0>			
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 Reset
'1' = Bit is set	'0' = Bit is cleared	

bit 7 **EN:** CCPx Module Enable bit
1 = CCPx is enabled
0 = CCPx is disabled

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

bit 5 **OUT:** CCPx Output Data bit (read-only)

bit 4 **FMT:** CCPW (Pulse-Width) Alignment bit
If MODE = PWM Mode
1 = Left-aligned format, CCPRxH <7> is the MSB of the PWM duty cycle
0 = Right-aligned format, CCPRxL<0> is the LSB of the PWM duty cycle

bit 3-0 **MODE<3:0>:** CCPx Mode Selection bits
11xx = PWM mode

1011 = Compare mode: Pulse output, clear TMR1
1010 = Compare mode: Pulse output (0 - 1 - 0)
1001 = Compare mode: clear output on compare match. Output is set upon selection of this mode.
1000 = Compare mode: set output on compare match. Output is set upon selection of this mode.

0111 = Capture mode: every 16th rising edge
0110 = Capture mode: every 4th rising edge
0101 = Capture mode: every rising edge
0100 = Capture mode: every falling edge

0011 = Capture mode: every rising or falling edge
0010 = Compare mode: toggle output on match
0001 = Compare mode: Toggle output and clear TMR1 on match
0000 = Capture/Compare/PWM off (resets CCPx module) (reserved for backwards compatibility)

27.8.1 FALLING EVENT BLANKING OF RISING EVENT INPUTS

The falling event blanking counter inhibits rising event inputs from triggering a rising event. The falling event blanking time starts when the rising_event output drive goes false.

The falling event blanking time is set by the value contained in the COGxBLKF register (Register 27-17). Blanking times are calculated using the formula shown in Equation 27-1.

When the COGxBLKF value is zero, falling event blanking is disabled and the blanking counter output is true, thereby, allowing the event signal to pass straight through to the event trigger circuit.

27.8.2 RISING EVENT BLANKING OF FALLING EVENT INPUTS

The rising event blanking counter inhibits falling event inputs from triggering a falling event. The rising event blanking time starts when the falling_event output drive goes false.

The rising event blanking time is set by the value contained in the COGxBLKR register (Register 27-16).

When the COGxBLKR value is zero, rising event blanking is disabled and the blanking counter output is true, thereby, allowing the event signal to pass straight through to the event trigger circuit.

27.8.3 BLANKING TIME UNCERTAINTY

When the rising and falling sources that trigger the blanking counters are asynchronous to the COG_clock, it creates uncertainty in the blanking time. The maximum uncertainty is equal to one COG_clock period. Refer to Equation 27-1 and Example 27-1 for more detail.

27.9 Phase Delay

It is possible to delay the assertion of either or both the rising event and falling events. This is accomplished by placing a non-zero value in COGxPHR or COGxPHF phase-delay count registers, respectively (Register 27-18 and Register 27-19). Refer to Figure 27-10 for COG operation with CCP1 and phase delay. The delay from the input rising event signal switching to the actual assertion of the events is calculated the same as the dead-band and blanking delays. Refer to Equation 27-1.

When the phase-delay count value is zero, phase delay is disabled and the phase-delay counter output is true, thereby, allowing the event signal to pass straight through to the complementary output driver flop.

27.9.1 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 27-1: PHASE, DEAD-BAND, AND BLANKING TIME CALCULATION

$$T_{\min} = \frac{\text{Count}}{F_{COG\_\text{clock}}}$$

$$T_{\max} = \frac{\text{Count} + 1}{F_{COG\_\text{clock}}}$$

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

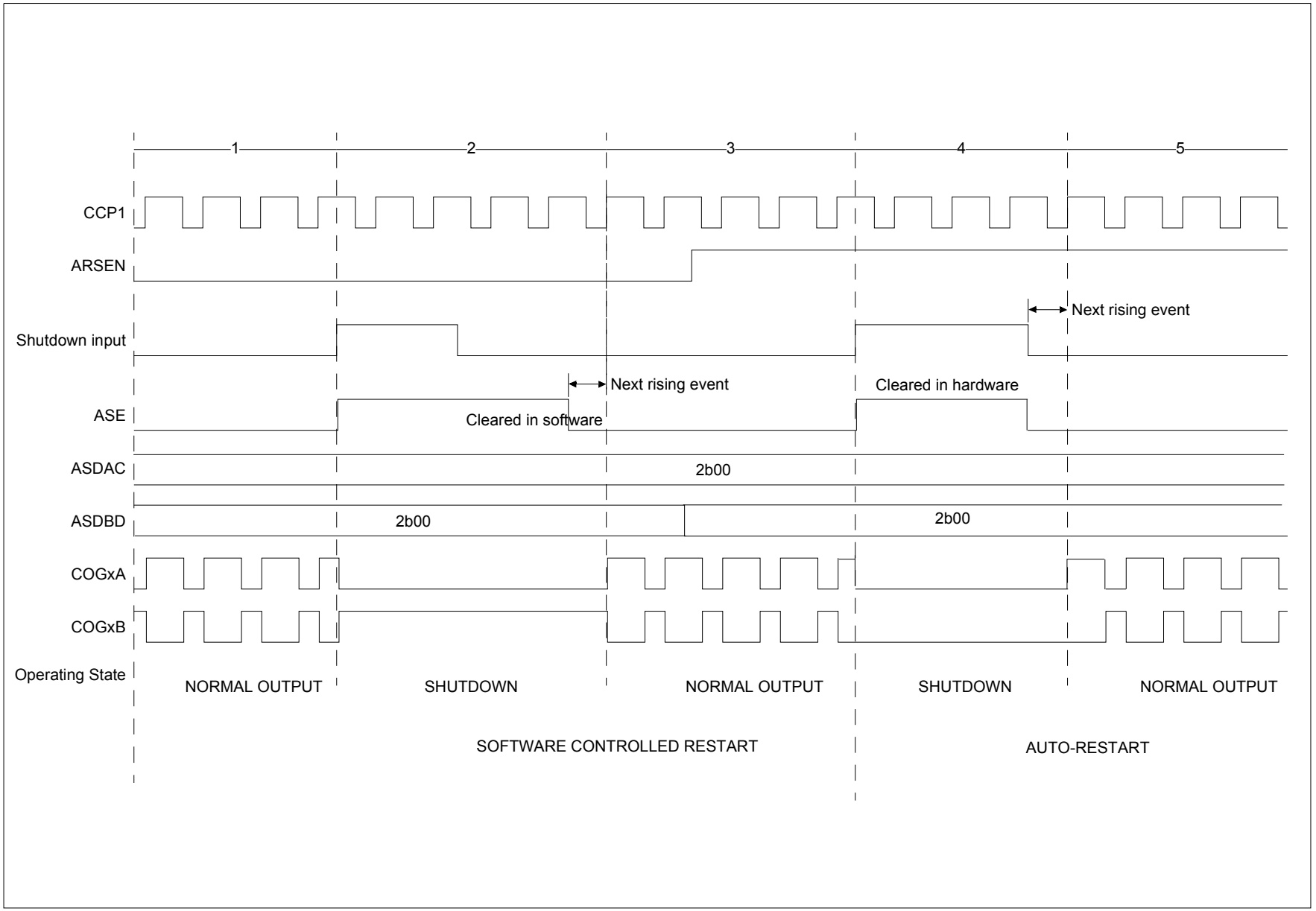
Also:

$$T_{\text{uncertainty}} = \frac{1}{F_{COG\_\text{clock}}}$$

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

FIGURE 27-15: AUTO-SHUTDOWN WAVEFORM – CCP1 AS RISING AND FALLING EVENT INPUT SOURCE



REGISTER 27-13: COGxSTR: COG STEERING CONTROL REGISTER 1(1)

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
SDATD	SDATC	SDATB	SDATA	STRD	STRC	STRB	STRA
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 **SDATD:** COGxD Static Output Data bit
- 1 = COGxD static data is high
- 0 = COGxD static data is low

bit 6 **SDATC:** COGxC Static Output Data bit
- 1 = COGxC static data is high
- 0 = COGxC static data is low

bit 5 **SDATB:** COGxB Static Output Data bit
- 1 = COGxB static data is high
- 0 = COGxB static data is low

bit 4 **SDATA:** COGxA Static Output Data bit
- 1 = COGxA static data is high
- 0 = COGxA static data is low

bit 3 **STRD:** COGxD Steering Control bit
- 1 = COGxD output has the COGxD waveform with polarity control from POLD bit
- 0 = COGxD output is the static data level determined by the SDATD bit

bit 2 **STRC:** COGxC Steering Control bit
- 1 = COGxC output has the COGxC waveform with polarity control from POLC bit
- 0 = COGxC output is the static data level determined by the SDATC bit

bit 1 **STRB:** COGxB Steering Control bit
- 1 = COGxB output has the COGxB waveform with polarity control from POLB bit
- 0 = COGxB output is the static data level determined by the SDATB bit

bit 0 **STRA:** COGxA Steering Control bit
- 1 = COGxA output has the COGxA waveform with polarity control from POLA bit
- 0 = COGxA output is the static data level determined by the SDATA bit

Note 1: Steering is active only when the MD<1:0> bits of the COGxCON0 register = 00x. (See Register 27-1).

32.4.9 ACKNOWLEDGE SEQUENCE

The ninth SCL pulse for any transferred byte in I^2C is dedicated as an Acknowledge. It allows receiving devices to respond back to the transmitter by pulling the SDA line low. The transmitter must release control of the line during this time to shift in the response. The Acknowledge ($\overline{ACK}$) is an active-low signal, pulling the SDA line low indicates to the transmitter that the device has received the transmitted data and is ready to receive more.

The result of an $\overline{ACK}$ is placed in the ACKSTAT bit of the SSPxCON2 register.

Slave software, when the AHEN and DHEN bits are set, allow the user to set the $\overline{ACK}$ value sent back to the transmitter. The ACKDT bit of the SSPxCON2 register is set/cleared to determine the response.

Slave hardware will generate an $\overline{ACK}$ response if the AHEN and DHEN bits of the SSPxCON3 register are clear.

There are certain conditions where an $\overline{ACK}$ will not be sent by the slave. If the BF bit of the SSPxSTAT register or the SSPOV bit of the SSPxCON1 register are set when a byte is received.

When the module is addressed, after the eighth falling edge of SCL on the bus, the ACKTIM bit of the SSPxCON3 register is set. The ACKTIM bit indicates the acknowledge time of the active bus. The ACKTIM Status bit is only active when the AHEN bit or DHEN bit is enabled.

32.5 I^2C SLAVE MODE OPERATION

The MSSP Slave mode operates in one of four modes selected by the SSPM bits of SSPxCON1 register. The modes can be divided into 7-bit and 10-bit Addressing mode. 10-bit Addressing modes operate the same as 7-bit with some additional overhead for handling the larger addresses.

Modes with Start and Stop bit interrupts operate the same as the other modes with SSPxIF additionally getting set upon detection of a Start, Restart, or Stop condition.

32.5.1 SLAVE MODE ADDRESSES

The SSPxADD register (Register 32-6) contains the Slave mode address. The first byte received after a Start or Restart condition is compared against the value stored in this register. If the byte matches, the value is loaded into the SSPxBUF register and an interrupt is generated. If the value does not match, the module goes idle and no indication is given to the software that anything happened.

The SSP Mask register (Register 32-5) affects the address matching process. See **Section 32.5.8 "SSP Mask Register"** for more information.

32.5.1.1 I^2C Slave 7-bit Addressing Mode

In 7-bit Addressing mode, the LSb of the received data byte is ignored when determining if there is an address match.

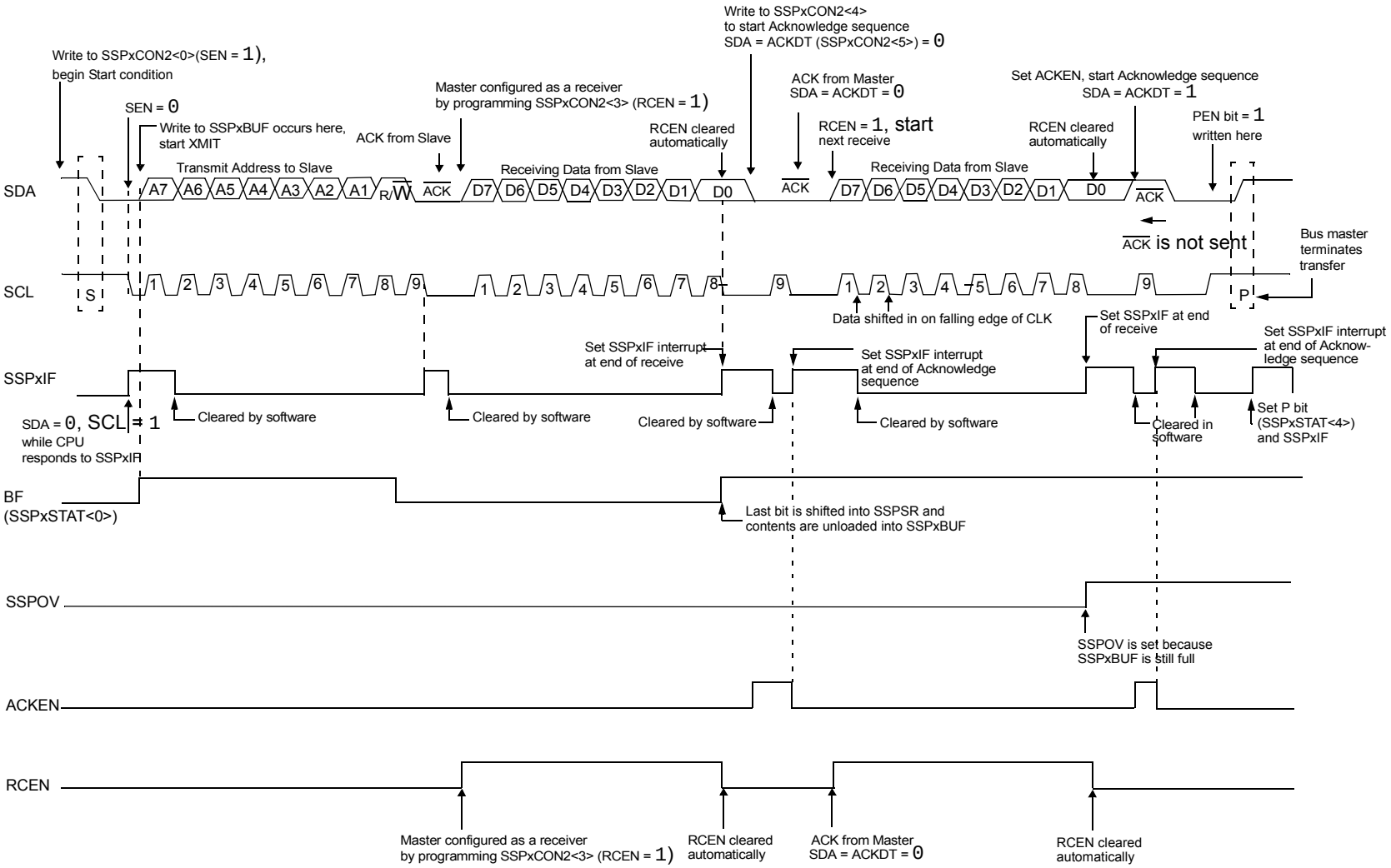
32.5.1.2 I^2C Slave 10-bit Addressing Mode

In 10-bit Addressing mode, the first received byte is compared to the binary value of '1 1 1 1 0 A9 A8 0'. A9 and A8 are the two MSb's of the 10-bit address and stored in bits 2 and 1 of the SSPxADD register.

After the acknowledge of the high byte the UA bit is set and SCL is held low until the user updates SSPxADD with the low address. The low address byte is clocked in and all eight bits are compared to the low address value in SSPxADD. Even if there is not an address match; SSPxIF and UA are set, and SCL is held low until SSPxADD is updated to receive a high byte again. When SSPxADD is updated the UA bit is cleared. This ensures the module is ready to receive the high address byte on the next communication.

A high and low address match as a write request is required at the start of all 10-bit addressing communication. A transmission can be initiated by issuing a Restart once the slave is addressed, and clocking in the high address with the R/$\overline{W}$ bit set. The slave hardware will then acknowledge the read request and prepare to clock out data. This is only valid for a slave after it has received a complete high and low address byte match.

FIGURE 32-29: I²C MASTER MODE WAVEFORM (RECEPTION, 7-BIT ADDRESS)



Note: Unless otherwise noted, VIN = 5V, FOSC = 300 kHz, CIN = 0.1 µF, TA = 25°C.

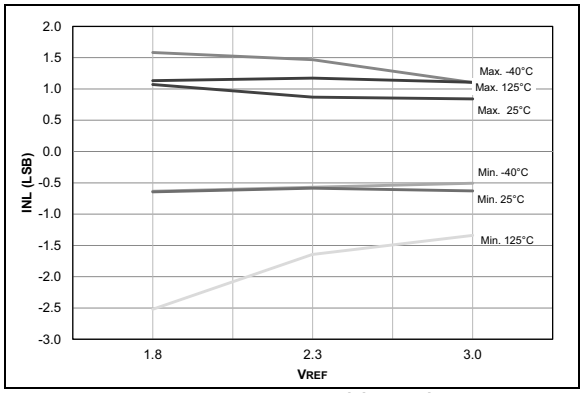


FIGURE 37-85: *ADC 10-bit Mode, Single-Ended INL, VDD = 3.0V, TAD = 1 µS.*

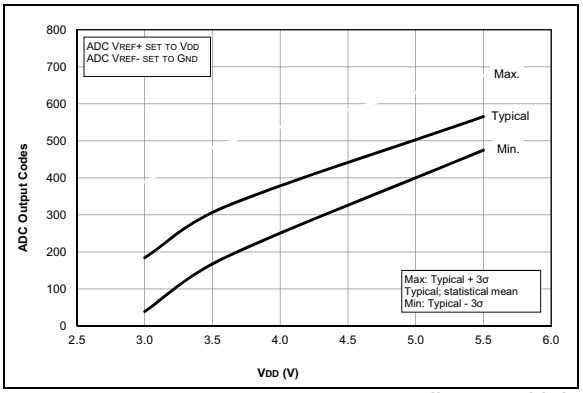


FIGURE 37-86: *Temperature Indicator Initial Offset, High Range, Temp. = 20°C, PIC16F1777/8/9 only.*

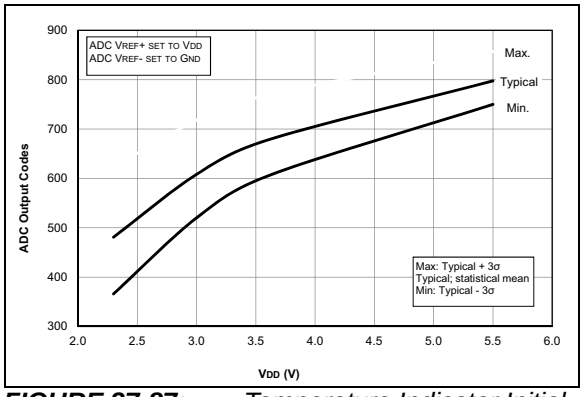


FIGURE 37-87: *Temperature Indicator Initial Offset, Low Range, Temp. = 20°C, PIC16F1777/8/9 only.*

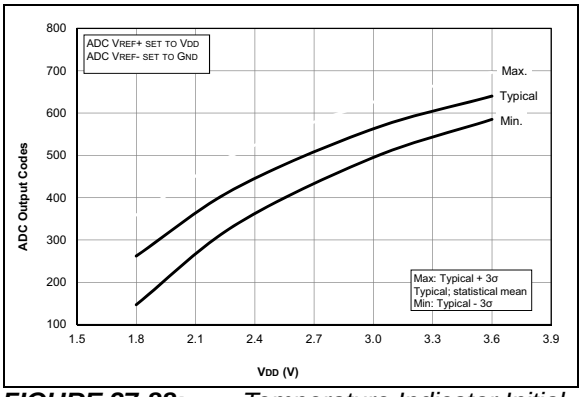


FIGURE 37-88: *Temperature Indicator Initial Offset, Low Range, Temp. = 20°C, PIC16LF1773/6 only.*

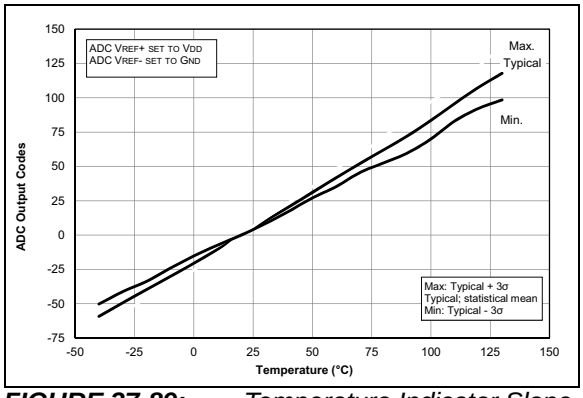


FIGURE 37-89: *Temperature Indicator Slope Normalized to 20°C, PIC16F1777/8/9 only.*

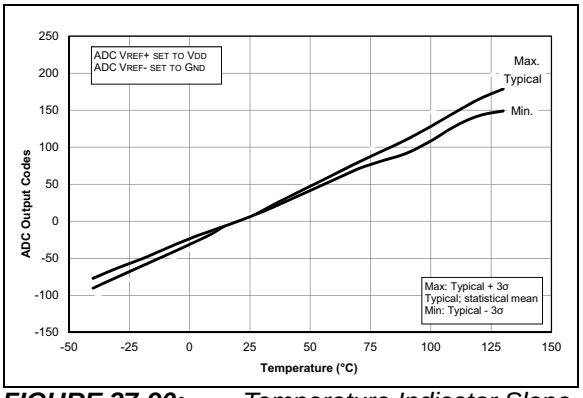


FIGURE 37-90: *Temperature Indicator Slope Normalized to 20°C, High Range, VDD = 3.6V, PIC16F1777/8/9 only.*