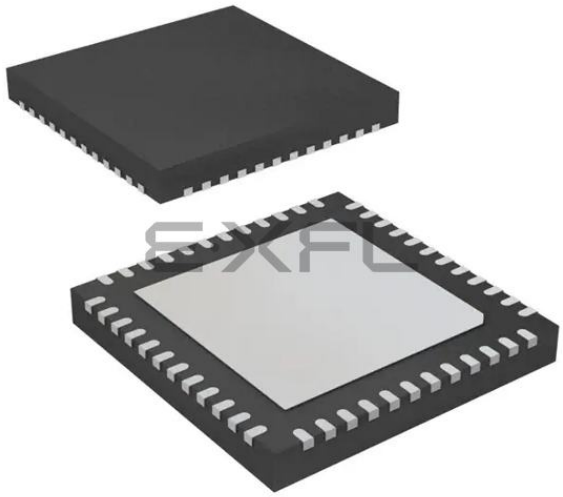


Welcome to E-XFL.COM

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	64MHz
Connectivity	I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LVD, POR, PWM, WDT
Number of I/O	44
Program Memory Size	32KB (16K x 16)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 43x12b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	48-UFQFN Exposed Pad
Supplier Device Package	48-UQFN (6x6)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18lf55k42-e-mv

FIGURE 4-4: DATA MEMORY MAP FOR PIC18(L)F26/27/45/46/47/55/56/57K42 DEVICES

Bank	BSR<5:0>	Address addr<7:0>	PIC18(L)F45K42 PIC18(L)F55K42	PIC18(L)F26K42 PIC18(L)F46K42 PIC18(L)F56K42	PIC18(L)F27K42 PIC18(L)F47K42 PIC18(L)F57K42	Address addr<13:0>
Bank 0	00 0000	00h	Access RAM	Access RAM	Access RAM	0000h 005Fh
		FFh	GPR	GPR	GPR	0060h 00FFh
Bank 1	00 0001	00h FFh	GPR	GPR	GPR	0100h 03FFh
Bank 2	00 0010	00h FFh				
Bank 3	00 0011	00h FFh				
Banks 4 to 7	00 0100 – 00 0111	00h FFh	GPR	GPR	GPR	0400h 07FFh
Banks 8 to 15	00 1000 – 00 1111	00h FFh	Unimplemented	GPR	GPR	0800h 0FFFh
Banks 16 to 31	01 0000 – 01 1111	00h FFh		Unimplemented		1000h 1FFFh
Banks 32 to 55	10 0000 – 11 0111	00h FFh			Unimplemented	2000h 37FFh
Banks 56 to 62	11 1000 – 11 1110	00h FFh	SFR	SFR	SFR	3800h 3EFFh
Bank 63	11 1111	00h FFh	SFR	SFR	SFR	3800h 3EFFh 3F60h 3FFFh

TABLE 4-7: SPECIAL FUNCTION REGISTER MAP FOR PIC18(L)F26/27/45/46/47/55/56/57K42 DEVICES BANK 60

3CFFh	—	3CDFh	—	3CBFh	—	3C9Fh	—	3C7Fh	—	3C5Fh	CLC4GLS3	3C3Fh	—	3C1Fh	—
3CFEh	MD1CARH	3CDEh	—	3CBEh	—	3C9Eh	—	3C7Eh	CLCDATA0	3C5Eh	CLC4GLS2	3C3Eh	—	3C1Eh	—
3CFDh	MD1CARL	3CDDh	—	3CBDh	—	3C9Dh	—	3C7Dh	CLC1GLS3	3C5Dh	CLC4GLS1	3C3Dh	—	3C1Dh	—
3CFCh	MD1SRC	3CDCh	—	3CBCh	—	3C9Ch	—	3C7Ch	CLC1GLS2	3C5Ch	CLC4GLS0	3C3Ch	—	3C1Ch	—
3CFBh	MD1CON1	3CDBh	—	3CBBh	—	3C9Bh	—	3C7Bh	CLC1GLS1	3C5Bh	CLC4SEL3	3C3Bh	—	3C1Bh	—
3CFAh	MD1CON0	3CDAh	—	3CBAh	—	3C9Ah	—	3C7Ah	CLC1GLS0	3C5Ah	CLC4SEL2	3C3Ah	—	3C1Ah	—
3CF9h	—	3CD9h	—	3CB9h	—	3C99h	—	3C79h	CLC1SEL3	3C59h	CLC4SEL1	3C39h	—	3C19h	—
3CF8h	—	3CD8h	—	3CB8h	—	3C98h	—	3C78h	CLC1SEL2	3C58h	CLC4SEL0	3C38h	—	3C18h	—
3CF7h	—	3CD7h	—	3CB7h	—	3C97h	—	3C77h	CLC1SEL1	3C57h	CLC4POL	3C37h	—	3C17h	—
3CF6h	—	3CD6h	—	3CB6h	—	3C96h	—	3C76h	CLC1SEL0	3C56h	CLC4CON	3C36h	—	3C16h	—
3CF5h	—	3CD5h	—	3CB5h	—	3C95h	—	3C75h	CLC1POL	3C55h	—	3C35h	—	3C15h	—
3CF4h	—	3CD4h	—	3CB4h	—	3C94h	—	3C74h	CLC1CON	3C54h	—	3C34h	—	3C14h	—
3CF3h	—	3CD3h	—	3CB3h	—	3C93h	—	3C73h	CLC2GLS3	3C53h	—	3C33h	—	3C13h	—
3CF2h	—	3CD2h	—	3CB2h	—	3C92h	—	3C72h	CLC2GLS2	3C52h	—	3C32h	—	3C12h	—
3CF1h	—	3CD1h	—	3CB1h	—	3C91h	—	3C71h	CLC2GLS1	3C51h	—	3C31h	—	3C11h	—
3CF0h	—	3CD0h	—	3CB0h	—	3C90h	—	3C70h	CLC2GLS0	3C50h	—	3C30h	—	3C10h	—
3CEFh	—	3CCFh	—	3CAFh	—	3C8Fh	—	3C6Fh	CLC2SEL3	3C4Fh	—	3C2Fh	—	3C0Fh	—
3CEEh	—	3CCEh	—	3CAEh	—	3C8Eh	—	3C6Eh	CLC2SEL2	3C4Eh	—	3C2Eh	—	3C0Eh	—
3CEDh	—	3CCDh	—	3CADh	—	3C8Dh	—	3C6Dh	CLC2SEL1	3C4Dh	—	3C2Dh	—	3C0Dh	—
3CECh	—	3CCCh	—	3CACh	—	3C8Ch	—	3C6Ch	CLC2SEL0	3C4Ch	—	3C2Ch	—	3C0Ch	—
3CEBh	—	3CCBh	—	3CABh	—	3C8Bh	—	3C6Bh	CLC2POL	3C4Bh	—	3C2Bh	—	3C0Bh	—
3CEAh	—	3CCAh	—	3CAAh	—	3C8Ah	—	3C6Ah	CLC2CON	3C4Ah	—	3C2Ah	—	3C0Ah	—
3CE9h	—	3CC9h	—	3CA9h	—	3C89h	—	3C69h	CLC3GLS3	3C49h	—	3C29h	—	3C09h	—
3CE8h	—	3CC8h	—	3CA8h	—	3C88h	—	3C68h	CLC3GLS2	3C48h	—	3C28h	—	3C08h	—
3CE7h	—	3CC7h	—	3CA7h	—	3C87h	—	3C67h	CLC3GLS1	3C47h	—	3C27h	—	3C07h	—
3CE6h	CLKRCLK	3CC6h	—	3CA6h	—	3C86h	—	3C66h	CLC3GLS0	3C46h	—	3C26h	—	3C06h	—
3CE5h	CLKRCON	3CC5h	—	3CA5h	—	3C85h	—	3C65h	CLC3SEL3	3C45h	—	3C25h	—	3C05h	—
3CE4h	—	3CC4h	—	3CA4h	—	3C84h	—	3C64h	CLC3SEL2	3C44h	—	3C24h	—	3C04h	—
3CE3h	—	3CC3h	—	3CA3h	—	3C83h	—	3C63h	CLC3SEL1	3C43h	—	3C23h	—	3C03h	—
3CE2h	—	3CC2h	—	3CA2h	—	3C82h	—	3C62h	CLC3SEL0	3C42h	—	3C22h	—	3C02h	—
3CE1h	—	3CC1h	—	3CA1h	—	3C81h	—	3C61h	CLC3POL	3C41h	—	3C21h	—	3C01h	—
3CE0h	—	3CC0h	—	3CA0h	—	3C80h	—	3C60h	CLC3CON	3C40h	—	3C20h	—	3C00h	—

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

Note 1: Unimplemented in LF devices.

2: Unimplemented in PIC18(L)F26/27K42.

3: Unimplemented in PIC18(L)F26/27/45/46/47K42.

REGISTER 5-9: CONFIGURATION WORD 5L (30 0008h)

U-1	U-1	U-1	U-1	U-1	U-1	U-1	R/W-1
—	—	—	—	—	—	—	$\overline{CP}$
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '1'	
-n = Value for blank device	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-1 **Unimplemented:** Read as '1'

bit 0 **$\overline{CP}$:** User Program Flash Memory and Data EEPROM Code Protection bit
1 = User Program Flash Memory and Data EEPROM code protection is disabled
0 = User Program Flash Memory and Data EEPROM code protection is enabled

REGISTER 5-10: CONFIGURATION WORD 5H (30 0009h)

U-1	U-1	U-1	U-1	U-1	U-1	U-1	U-1
—	—	—	—	—	—	—	—
bit 7							bit 0

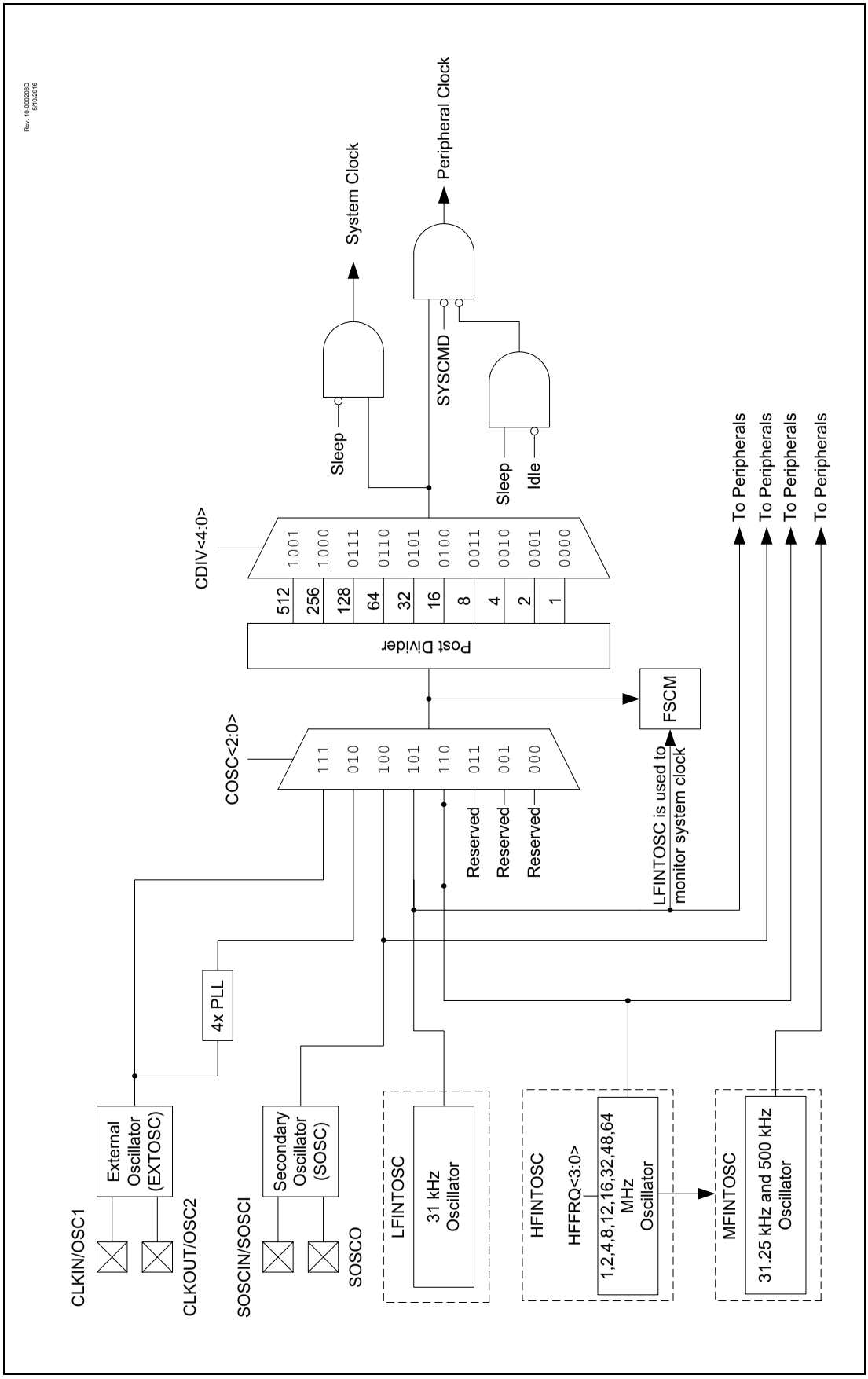
Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '1'	
-n = Value for blank device	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-0 **Unimplemented:** Read as '1'

TABLE 5-2: SUMMARY OF CONFIGURATION WORDS

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Default/ Unprogrammed Value
30 0000h	CONFIG1L	—	RSTOSC<2:0>			—	FEXTOSC<2:0>			1111 1111
30 0001h	CONFIG1H	—	—	FCMEN	—	CSWEN	—	PR1WAY	$\overline{CLKOUTEN}$	1111 1111
30 0002h	CONFIG2L	BOREN<1:0>		$\overline{LPBOREN}$	IVT1WAY	MVECEN	PWRTS<1:0>		MCLRE	1111 1111
30 0003h	CONFIG2H	$\overline{XINST}$	—	$\overline{DEBUG}$	STVREN	PPS1WAY	$\overline{ZCD}$	BORV<1:0>		1111 1111
30 0004h	CONFIG3L	—	WDTE<1:0>		WDTCPS<4:0>					1111 1111
30 0005h	CONFIG3H	—	—	WDTCCS<2:0>			WDTCWS<2:0>			1111 1111
30 0006h	CONFIG4L	$\overline{WRTAPP}$	—	—	$\overline{SAFEN}$	$\overline{BBEN}$	BBSIZE<2:0>			1111 1111
30 0007h	CONFIG4H	—	—	LVP	—	$\overline{WRTSAF}$	$\overline{WRTD}$	$\overline{WRTC}$	$\overline{WRTB}$	1111 1111
30 0008h	CONFIG5L	—	—	—	—	—	—	—	$\overline{CP}$	1111 1111
30 0009h	CONFIG5H	—	—	—	—	—	—	—	—	1111 1111

FIGURE 7-1: SIMPLIFIED PIC® MCU CLOCK SOURCE BLOCK DIAGRAM



REGISTER 9-7: PIR4: PERIPHERAL INTERRUPT REGISTER 4(1)

R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	U-0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0
CLC1IF	CWG1IF	NCO1IF	—	CCP1IF	TMR2IF	TMR1GIF	TMR1IF
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 **CLC1IF:** CLC1 Interrupt Flag bit
- 1 = Interrupt has occurred (must be cleared by software)
- 0 = Interrupt event has not occurred

bit 6 **CWG1IF:** CWG1 Interrupt Flag bit
- 1 = Interrupt has occurred (must be cleared by software)
- 0 = Interrupt event has not occurred

bit 5 **NCO1IF:** NCO1 Interrupt Flag bit
- 1 = Interrupt has occurred (must be cleared by software)
- 0 = Interrupt event has not occurred

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

bit 3 **CCP1IF:** CCP1 Interrupt Flag bit
- 1 = Interrupt has occurred (must be cleared by software)
- 0 = Interrupt event has not occurred

bit 2 **TMR2IF:** TMR2 Interrupt Flag bit
- 1 = Interrupt has occurred (must be cleared by software)
- 0 = Interrupt event has not occurred

bit 1 **TMR1GIF:** TMR1 Gate Interrupt Flag bit
- 1 = Interrupt has occurred (must be cleared by software)
- 0 = Interrupt event has not occurred

bit 0 **TMR1IF:** TMR1 Interrupt Flag bit
- 1 = Interrupt has occurred (must be cleared by software)
- 0 = Interrupt event has not occurred

Note 1: Interrupt flag bits get set when an interrupt condition occurs, regardless of the state of its corresponding enable bit, or the global enable bit. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 9-27: IPR2: PERIPHERAL INTERRUPT PRIORITY REGISTER 2

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
I2C1RXIP	SPI1IP	SPI1TXIP	SPI1RXIP	DMA1AIP	DMA1ORIP	DMA1DCNTIP	DMA1SCNTIP
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 **I2C1RXIP:** I^2C1 Receive Interrupt Priority bit
1 = High priority
0 = Low priority

bit 6 **SPI1IP:** SPI1 Transmit Interrupt Priority bit
1 = High priority
0 = Low priority

bit 5 **SPI1TXIP:** I^2C1 Transmit Interrupt Priority bit
1 = High priority
0 = Low priority

bit 4 **SPI1RXIP:** SPI1 Receive Interrupt Priority bit
1 = High priority
0 = Low priority

bit 3 **DMA1AIP:** DMA1 Abort Transmit Interrupt Priority bit
1 = High priority
0 = Low priority

bit 2 **DMA1ORIP:** DMA1 Overrun Interrupt Priority bit
1 = High priority
0 = Low priority

bit 1 **DMA1DCNTIP:** DMA1 Destination Count Interrupt Priority bit
1 = High priority
0 = Low priority

bit 0 **DMA1SCNTIP:** DMA1 Source Count Interrupt Priority bit
1 = High priority
0 = Low priority

13.1.2 CONTROL REGISTERS

Several control registers are used in conjunction with the TBLRD and TBLWT instructions. These include the following registers:

- NVMCON1 register
- NVMCON2 register
- TABLAT register
- TBLPTR registers

13.1.2.1 NVMCON1 and NVMCON2 Registers

The NVMCON1 register (Register 13-1) is the control register for memory accesses. The NVMCON2 register is not a physical register; it is used exclusively in the memory write and erase sequences. Reading NVMCON2 will read all '0's.

The REG<1:0> control bits determine if the access will be to Data EEPROM Memory locations. PFM locations or User IDs, Configuration bits, Rev ID and Device ID. When REG<1:0> = 00, any subsequent operations will operate on the Data EEPROM Memory. When REG<1:0> = 10, any subsequent operations will operate on the program memory. When REG<1:0> = x1, any subsequent operations will operate on the Configuration bits, User IDs, Rev ID and Device ID.

The FREE bit allows the program memory erase operation. When the FREE bit is set, an erase operation is initiated on the next WR command. When FREE is clear, only writes are enabled. This bit is applicable only to the PFM and not to data EEPROM.

When set, the WREN bit will allow a program/erase operation. The WREN bit is cleared on power-up.

The WRERR bit is set by hardware when the WR bit is set and is cleared when the internal programming timer expires and the write operation is successfully complete.

The WR control bit initiates erase/write cycle operation when the REG<1:0> bits point to the Data EEPROM Memory location, and it initiates a write operation when the REG<1:0> bits point to the PFM location. The WR bit cannot be cleared by firmware; it can only be set by firmware. Then the WR bit is cleared by hardware at the completion of the write operation.

The NVMIF Interrupt Flag bit is set when the write is complete. The NVMIF flag stays set until cleared by firmware.

13.1.2.2 TABLAT – Table Latch Register

The Table Latch (TABLAT) is an 8-bit register mapped into the SFR space. The Table Latch register is used to hold 8-bit data during data transfers between program memory and data RAM.

13.1.2.3 TBLPTR – Table Pointer Register

The Table Pointer (TBLPTR) register addresses a byte within the program memory. The TBLPTR is comprised of three SFR registers: Table Pointer Upper Byte, Table Pointer High Byte and Table Pointer Low Byte (TBLPTRU:TBLPTRH:TBLPTRL). These three registers join to form a 22-bit wide pointer. The low-order 21 bits allow the device to address up to 2 Mbytes of program memory space. The 22nd bit allows access to the Device ID, the User ID and the Configuration bits.

The Table Pointer register, TBLPTR, is used by the TBLRD and TBLWT instructions. These instructions can update the TBLPTR in one of four ways based on the table operation. These operations on the TBLPTR affect only the low-order 21 bits.

13.1.2.4 Table Pointer Boundaries

TBLPTR is used in reads, writes and erases of the Program Flash Memory.

When a TBLRD is executed, all 22 bits of the TBLPTR determine which byte is read from program memory directly into the TABLAT register.

When a TBLWT is executed the byte in the TABLAT register is written, not to memory but, to a holding register in preparation for a program memory write. The holding registers constitute a write block which varies depending on the device (see Table 5-4).The 3, 4, or 5 LSbs of the TBLPTRL register determine which specific address within the holding register block is written to. The MSBs of the Table Pointer have no effect during TBLWT operations.

When a program memory write is executed the entire holding register block is written to the memory at the address determined by the MSbs of the TBLPTR. The 3, 4, or 5 LSBs are ignored during memory writes. For more detail, see **Section 13.1.6 "Writing to Program Flash Memory"**.

Figure 13-3 describes the relevant boundaries of TBLPTR based on Program Flash Memory operations.

13.3 Data EEPROM Memory

The data EEPROM is a nonvolatile memory array, separate from the data RAM and program memory, which is used for long-term storage of program data. It is not directly mapped in either the register file or program memory space but is indirectly addressed through the Special Function Registers (SFRs). The EEPROM is readable and writable during normal operation over the entire VDD range.

Five SFRs are used to read and write to the data EEPROM as well as the program memory. They are:

- NVMCON1
- NVMCON2
- NVMDAT
- NVMADRL
- NVMADRH(1)

Note 1: NVMADRH register is not implemented on PIC18(L)F45/55K42.

The data EEPROM allows byte read and write. When interfacing to the data memory block, NVMDAT holds the 8-bit data for read/write and the NVMADRH:NVMADRL register pair holds the address of the EEPROM location being accessed.

The EEPROM data memory is rated for high erase/write cycle endurance. A byte write automatically erases the location and writes the new data (erase-before-write). The write time is controlled by an internal programming timer; it will vary with voltage and temperature as well as from chip-to-chip. Refer to the Data EEPROM Memory parameters in **Section 44.0 "Electrical Specifications"** for limits.

13.3.1 NVMADRL AND NVMADRH REGISTERS

The NVMADRH:NVMADRL registers are used to address the data EEPROM for read and write operations.

13.3.2 NVMCON1 AND NVMCON2 REGISTERS

Access to the data EEPROM is controlled by two registers: NVMCON1 and NVMCON2. These are the same registers which control access to the program memory and are used in a similar manner for the data EEPROM.

The NVMCON1 register (Register 13-1) is the control register for data and program memory access. Control bits REG<1:0> determine if the access will be to program, Data EEPROM Memory or the User IDs, Configuration bits, Revision ID and Device ID.

The WREN bit, when set, will allow a write operation. On power-up, the WREN bit is clear.

The WRERR bit is set by hardware when the WR bit is set and cleared when the internal programming timer expires and the write operation is complete.

The WR control bit initiates write operations. The bit can be set but not cleared by software. It is cleared only by hardware at the completion of the write operation.

The NVMIF Interrupt Flag bit of the PIR0 register is set when the write is complete. It must be cleared by software.

Control bits, RD and WR, start read and erase/write operations, respectively. These bits are set by firmware and cleared by hardware at the completion of the operation.

The RD bit cannot be set when accessing program memory (REG<1:0> = 0x10). Program memory is read using table read instructions. See **Section 13.1.1 "Table Reads and Table Writes"** regarding table reads.

13.3.5 WRITE VERIFY

Depending on the application, good programming practice may dictate that the value written to the memory should be verified against the original value. This should be used in applications where excessive writes can stress bits near the specification limit.

EXAMPLE 13-5: DATA EEPROM READ

```
; Data Memory Address to read
            CLRF      NVMCON1          ; Setup Data EEPROM Access
            MOVF      EE_ADDRL, W      ;
            MOVWF     NVMADRL          ; Setup Address
            BSF       NVMCON1, RD      ; Issue EE Read
            MOVF      NVMDAT, W        ; W = EE_DATA
```

EXAMPLE 13-6: DATA EEPROM WRITE

```
; Data Memory Address to write
            CLRF      NVMCON1          ; Setup Data EEPROM Access
            MOVF      EE_ADDRL, W      ;
            MOVWF     NVMADRL          ; Setup Address
; Data Memory Value to write
            MOVF      EE_DATA, W       ;
            MOVWF     NVMDAT           ;
; Enable writes
            BSF       NVMCON1, WREN    ;
; Disable interrupts
            BCF       INTCON0, GIE     ;
; Required unlock sequence
            MOVLW     55h              ;
            MOVWF     NVMCON2          ;
            MOVLW     AAh              ;
            MOVWF     NVMCON2          ;
; Set WR bit to begin write
            BSF       NVMCON1, WR      ;
; Enable INT
            BSF       INTCON0, GIE     ;
; Wait for interrupt, write done
            SLEEP                      ;
; Disable writes
            BCF       NVMCON1, WREN    ;
```

13.3.6 OPERATION DURING CODE-PROTECT

Data EEPROM Memory has its own code-protect bits in Configuration Words. External read and write operations are disabled if code protection is enabled.

If the Data EEPROM is write-protected or if NVMADR points an invalid address location, the WR bit is cleared without any effect. WRERR is signaled in this scenario.

13.3.7 PROTECTION AGAINST SPURIOUS WRITE

There are conditions when the user may not want to write to the Data EEPROM Memory. To protect against spurious EEPROM writes, various mechanisms have been implemented. On power-up, the WREN bit is cleared. In addition, writes to the EEPROM are blocked during the Power-up Timer period (TPWRT).

The unlock sequence and the WREN bit together help prevent an accidental write during brown-out, power glitch or software malfunction.

22.5.10 LEVEL-TRIGGERED HARDWARE LIMIT ONE-SHOT MODES

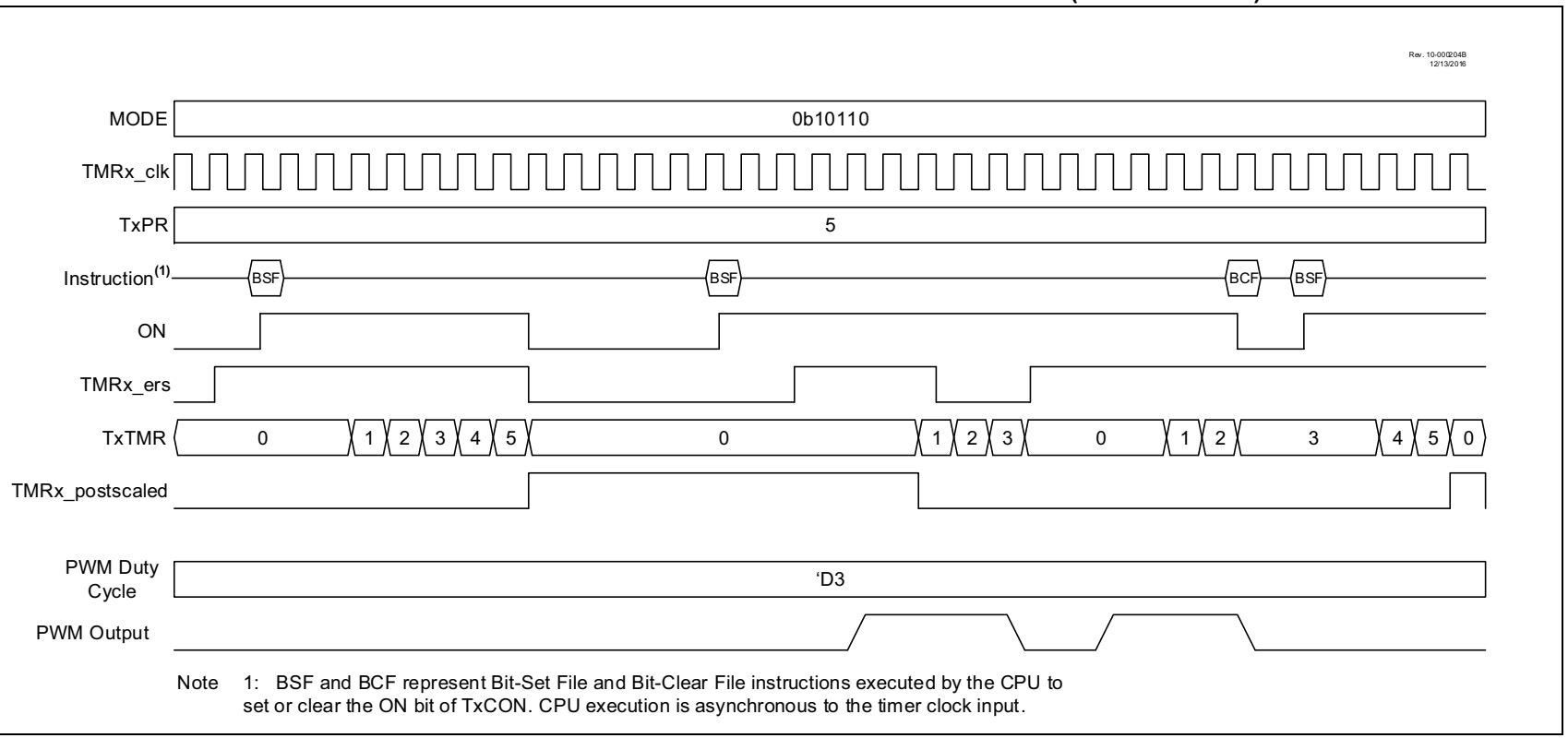
The Level Triggered Hardware Limit One-Shot modes hold the timer in Reset on an external Reset level and start counting when both the ON bit is set and the external signal is not at the Reset level. If one of either the external signal is not in reset or the ON bit is set then the other signal being 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 T2PR period count, the timer is reset and the ON bit is cleared. When the ON bit is cleared by either a T2PR 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.

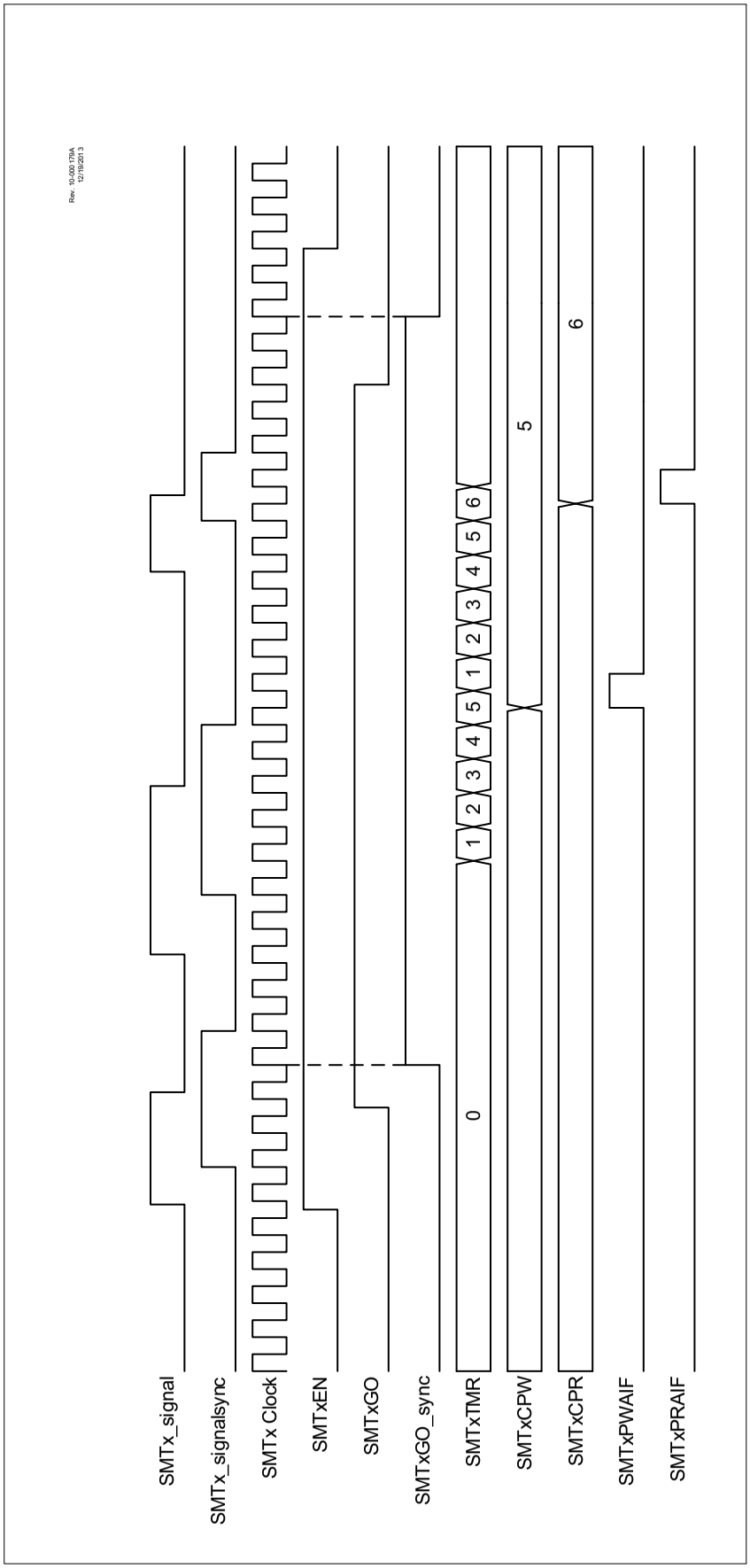
When Level Triggered Hardware Limit One-Shot modes are used in conjunction with the CCP PWM operation, the PWM drive goes active with either the external signal edge or the setting of the ON bit, whichever of the two starts the timer.

FIGURE 22-13: LEVEL-TRIGGERED HARDWARE LIMIT ONE-SHOT MODE TIMING DIAGRAM (MODE = 10110)



Note 1: BSF and BCF represent Bit-Set File and Bit-Clear File instructions executed by the CPU to set or clear the ON bit of TxCON. CPU execution is asynchronous to the timer clock input.

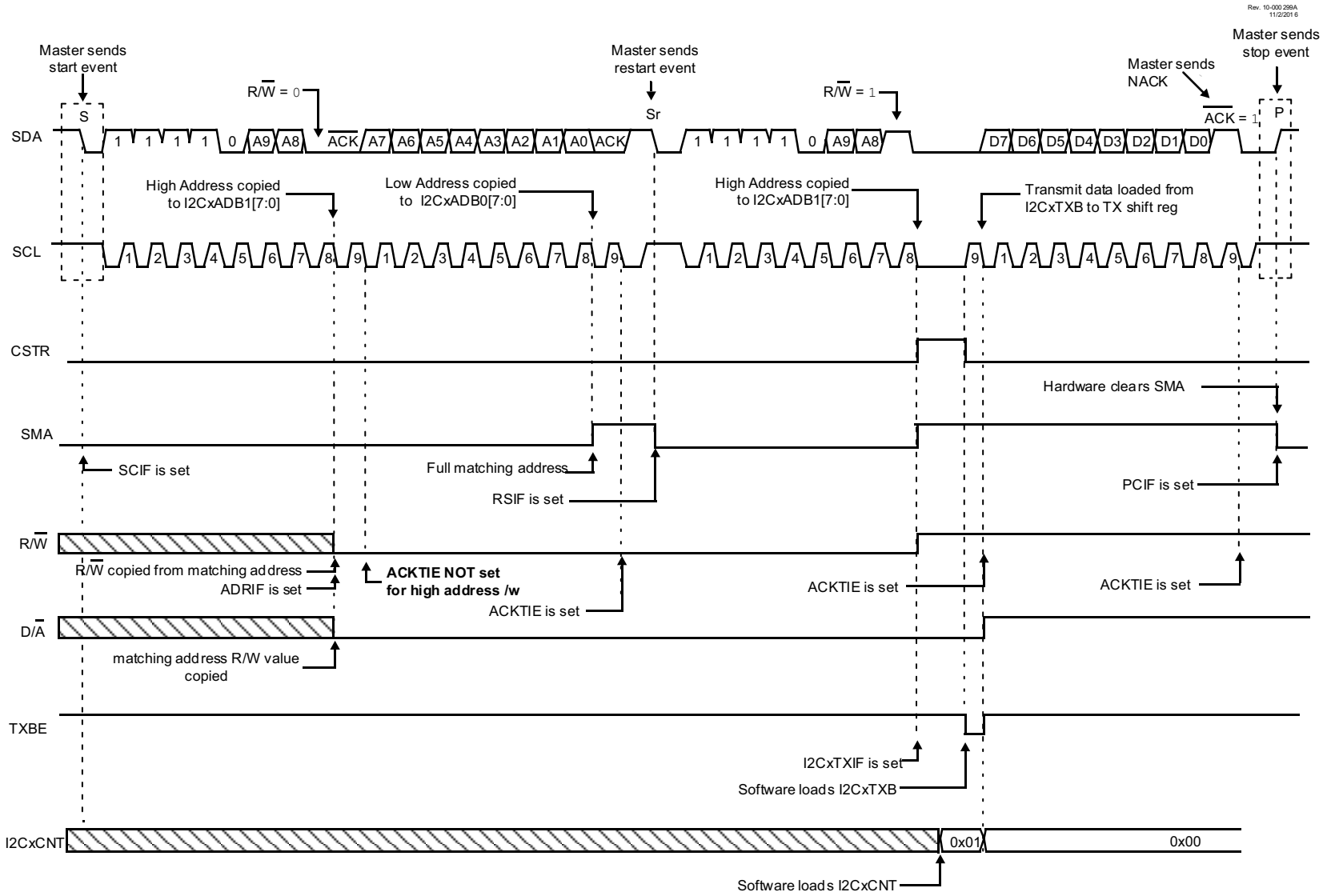
FIGURE 25-9: HIGH AND LOW MEASURE MODE SINGLE ACQUISITION TIMING DIAGRAM



25.6.5 WINDOWED MEASURE MODE

This mode measures the window duration of the SMTWINx input of the SMT. It begins incrementing the timer on a rising edge of the SMTWINx input and updates the SMT1CPR register with the value of the timer and resets the timer on a second rising edge. See Figure 25-10 and Figure 25-11.

FIGURE 33-12: I²C SLAVE, 10-BIT ADDRESS, TRANSMISSION



35.0 TEMPERATURE INDICATOR MODULE

This family of devices is equipped with a temperature circuit designed to measure the operating temperature of the silicon die.

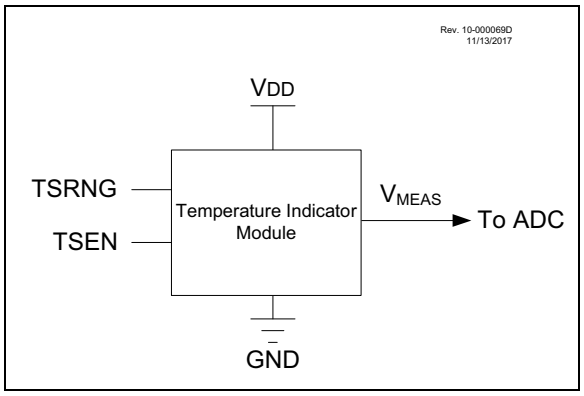
The circuit's range of operating temperature falls between -40°C and +125°C. A one-point calibration allows the circuit to indicate a temperature closely surrounding that point. A two-point calibration allows the circuit to sense the entire range of temperature more accurately.

35.1 Module Operation

The temperature indicator module consists of a temperature-sensing circuit that provides a voltage to the device ADC. The analog voltage output, VMEAS, varies inversely to the device temperature. The output of the temperature indicator is referred to as VMEAS.

Figure 35-1 shows a simplified block diagram of the temperature indicator module.

FIGURE 35-1: TEMPERATURE INDICATOR MODULE BLOCK DIAGRAM



The output of the circuit is measured using the internal Analog-to-Digital Converter. A channel is reserved for the temperature circuit output. Refer to **Section 36.0 "Analog-to-Digital Converter with Computation (ADC2) Module"** for detailed information.

The ON/OFF bit for the module is located in the FVRCON register. See **Section 34.0 "Fixed Voltage Reference (FVR)"** for more information. The circuit is enabled by setting the TSEN bit of the FVRCON register. When the module is disabled, the circuit draws no current.

The circuit operates in either High or Low range. Refer to the next section for more details on the range settings.

35.1.1 TEMPERATURE INDICATOR RANGE

The temperature indicator circuit operates in either high or low range. The high range, selected by setting the TSRNG bit of the FVRCON register, provides a wider output voltage. This provides more resolution over the temperature range. High range requires a higher-bias voltage to operate and thus, a higher VDD is needed. The low range is selected by clearing the TSRNG bit of the FVRCON register. The low range generates a lower sensor voltage and thus, a lower VDD voltage is needed to operate the circuit.

The output voltage of the sensor is the highest value at -40°C and the lowest value at +125°C.

High Range: The High range is used in applications with the reference for the ADC, VREF = 2.048V. This range may not be suitable for battery-powered applications. The ADC reading (in counts) at 90°C for the high range setting is stored in the DIA Table (Table 5-3) as parameter TSHR2.

Low Range: This mode is useful in applications in which the VDD is too low for high-range operation. The VDD in this mode can be as low as 1.8V. VDD must, however, be at least 0.5V higher than the maximum sensor voltage depending on the expected low operating temperature. The ADC reading (in counts) at 90°C for the Low range setting is stored in the DIA Table (Table 5-3) as parameter TSLR2.

35.1.2 MINIMUM OPERATING VDD

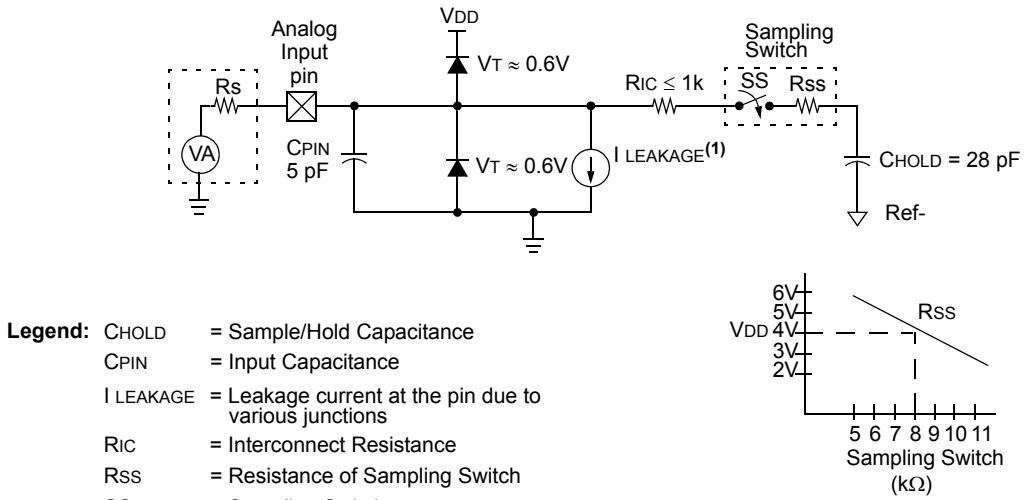
When the temperature circuit is operated in low range, the device may be operated at any operating voltage that is within specifications. When the temperature circuit is operated in high range, the device operating voltage, VDD, must be high enough to ensure that the temperature circuit is correctly biased.

Table 35-1 shows the recommended minimum VDD vs. Range setting.

TABLE 35-1: RECOMMENDED VDD vs. RANGE

Min.VDD, TSRNG = 1 (High Range)	Min. VDD, TSRNG = 0 (Low Range)
≥ 2.5	≥ 1.8

FIGURE 36-4: ANALOG INPUT MODEL

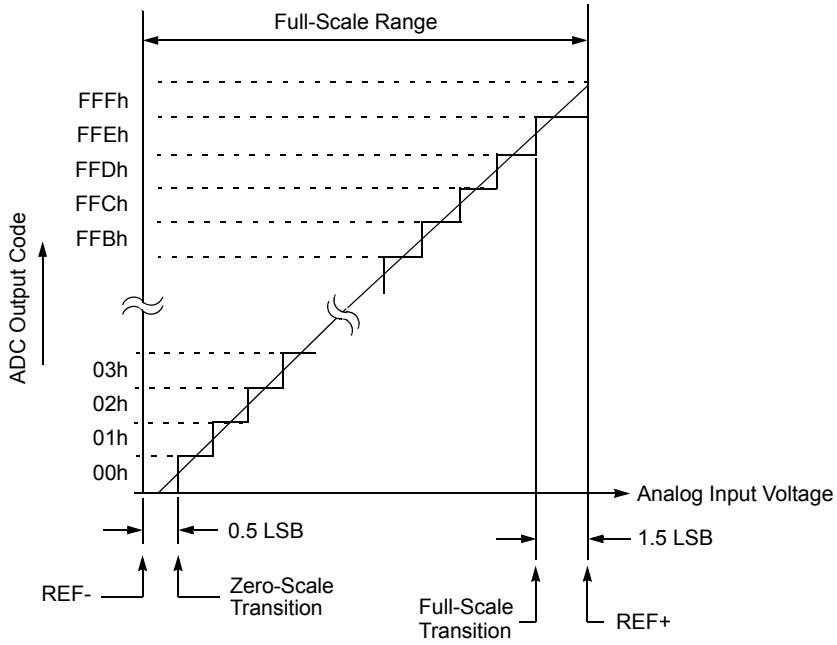


Legend:

CHOLD	= Sample/Hold Capacitance
CPIN	= Input Capacitance
I LEAKAGE	= Leakage current at the pin due to various junctions
RIC	= Interconnect Resistance
RSS	= Resistance of Sampling Switch
SS	= Sampling Switch
VT	= Threshold Voltage

Note 1: Refer to Table 44-6 (parameter D340 and D341).

FIGURE 36-5: ADC TRANSFER FUNCTION



REGISTER 36-11: ADACQL: ADC ACQUISITION TIME CONTROL REGISTER (LOW BYTE)

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
ACQ<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 ACQ<7:0>: Acquisition (charge share time) Select bits
See Table 36-5.

REGISTER 36-12: ADACQH: ADC ACQUISITION TIME CONTROL REGISTER (HIGH BYTE)

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
—	—	—	ACQ<12:8>				
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 7-5 **Unimplemented**: Read as ‘0’

bit 4-0 ACQ<12:8>: Acquisition (charge share time) Select bits
See Table 36-5.

TABLE 36-5: ACQUISITION TIME

ADACQ	Acquisition time
1 1111 1111 1111	8191 clocks of the selected ADC clock
1 1111 1111 1110	8190 clocks of the selected ADC clock
1 1111 1111 1101	8189 clocks of the selected ADC clock
...	...
0 0000 0000 0010	2 clocks of the selected ADC clock
0 0000 0000 0001	1 clock of the selected ADC clock
0 0000 0000 0000	Not included in the data conversion cycle(1)

Note 1: If ADPRE is not equal to ‘0’, then ADACQ = 0b0_0000_0000_0000 means Acquisition time is 8192 clocks of the selected ADC clock.

REGISTER 36-18: ADRESH: ADC RESULT REGISTER HIGH, FM = 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<11:4>							
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<11:4>**: ADC Result Register bits
Upper eight bits of 12-bit conversion result.

REGISTER 36-19: ADRESL: ADC RESULT REGISTER LOW, FM = 0

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	U-0	U-0	U-0	U-0
ADRES<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 Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7-4 **ADRES<3:0>**: ADC Result Register bits. Lower four bits of 12-bit conversion result.
bit 3-0 **Reserved**

38.2 Comparator Control

Each comparator has two control registers: CMxCON0 and CMxCON1.

The CMxCON0 register (see Register 38-1) contains Control and Status bits for the following:

- Enable
- Output
- Output polarity
- Hysteresis enable
- Timer1 output synchronization

The CMxCON1 register (see Register 38-2) contains Control bits for the following:

- Interrupt on positive/negative edge enables

The CMxPCH and CMxNCH registers are used to select the positive and negative input channels, respectively.

38.2.1 COMPARATOR ENABLE

Setting the EN bit of the CMxCON0 register enables the comparator for operation. Clearing the EN bit disables the comparator resulting in minimum current consumption.

38.2.2 COMPARATOR OUTPUT

The output of the comparator can be monitored by reading either the CxOUT bit of the CMxCON0 register or the CxOUT bit of the CMOUT register.

The comparator output can also be routed to an external pin through the RxyPPS register (Register 17-2). The corresponding TRIS bit must be clear to enable the pin as an output.

Note 1: The internal output of the comparator is latched with each instruction cycle. Unless otherwise specified, external outputs are not latched.

38.2.3 COMPARATOR OUTPUT POLARITY

Inverting the output of the comparator is functionally equivalent to swapping the comparator inputs. The polarity of the comparator output can be inverted by setting the POL bit of the CMxCON0 register. Clearing the POL bit results in a noninverted output.

Table 38-1 shows the output state versus input conditions, including polarity control.

TABLE 38-1: COMPARATOR OUTPUT STATE VS. INPUT CONDITIONS

Input Condition	POL	CxOUT
CxVN > CxVP	0	0
CxVN < CxVP	0	1
CxVN > CxVP	1	1
CxVN < CxVP	1	0

CALLW **Subroutine Call Using WREG**

Syntax: CALLW

Operands: None

Operation: (PC + 2) → TOS,
(W) → PCL,
(PCLATH) → PCH,
(PCLATU) → PCU

Status Affected: None

Encoding:

0000	0000	0001	0100

Description: First, the return address (PC + 2) is pushed onto the return stack. Next, the contents of W are written to PCL; the existing value is discarded. Then, the contents of PCLATH and PCLATU are latched into PCH and PCU, respectively. The second cycle is executed as a NOP instruction while the new next instruction is fetched. Unlike CALL, there is no option to update W, Status or BSR.

Words: 1

Cycles: 2

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read WREG	PUSH PC to stack	No operation
No operation	No operation	No operation	No operation

Example:
```
HERE    CALLW
```
Before Instruction
PC = address (HERE)
PCLATH = 10h
PCLATU = 00h
W = 06h
After Instruction
PC = 001006h
TOS = address (HERE + 2)
PCLATH = 10h
PCLATU = 00h
W = 06h

CLRF **Clear f**

Syntax: CLRF f {,a}

Operands: 0 ≤ f ≤ 255
a ∈ [0,1]

Operation: 000h → f
1 → Z

Status Affected: Z

Encoding:

0110	101a	ffff	ffff

Description: Clears the contents of the specified register.
If 'a' is '0', the Access Bank is selected.
If 'a' is '1', the BSR is used to select the GPR bank.
If 'a' is '0' and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever f ≤ 95 (5Fh). See **Section 41.2.3 "Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode"** for details.

Words: 1

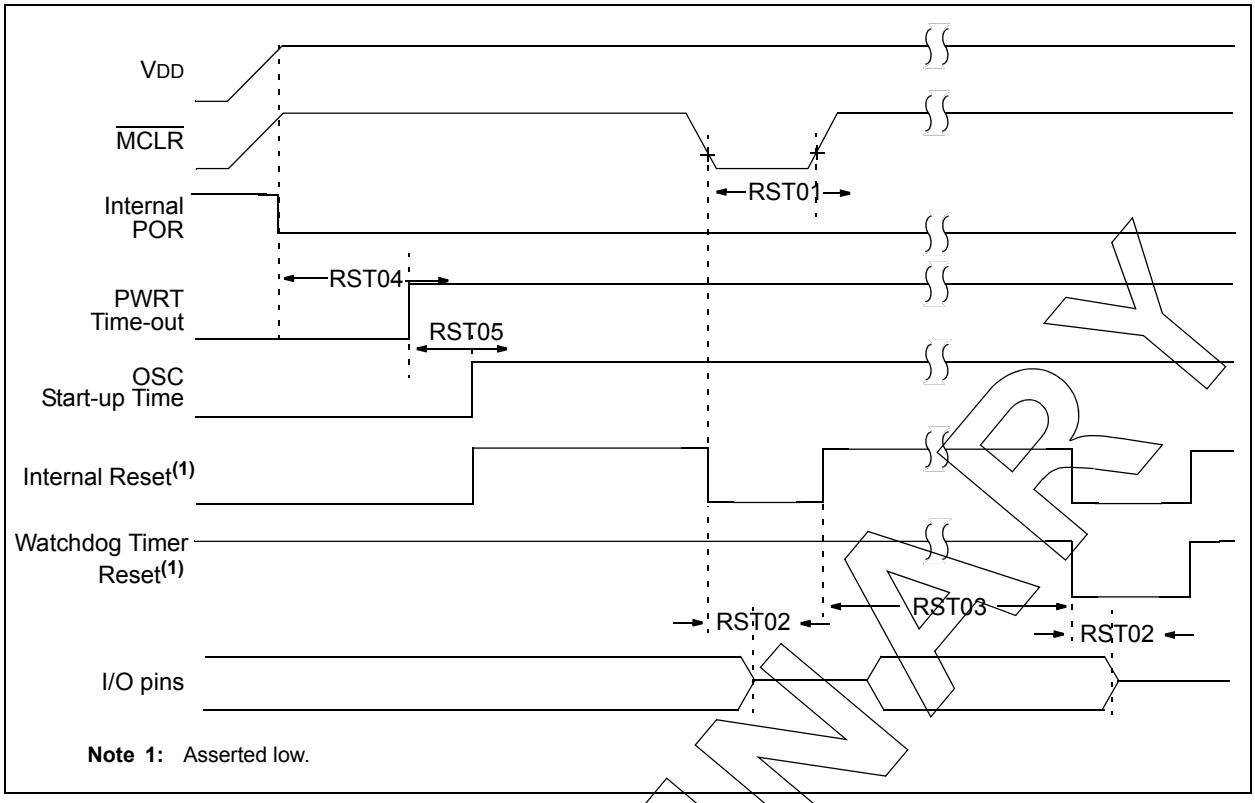
Cycles: 1

Q Cycle Activity:

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

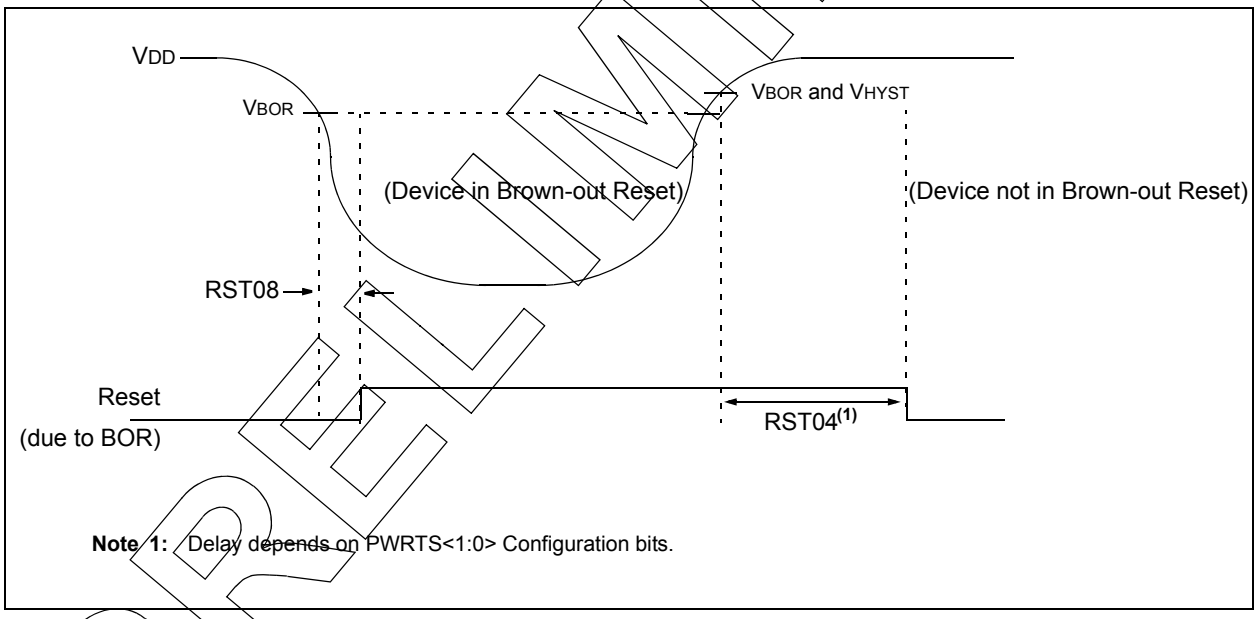
Example:
```
CLRF    FLAG_REG, 1
```
Before Instruction
FLAG_REG = 5Ah
After Instruction
FLAG_REG = 00h

FIGURE 44-8: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING



Note 1: Asserted low.

FIGURE 44-9: BROWN-OUT RESET TIMING AND CHARACTERISTICS



Note 1: Delay depends on PWRTS<1:0> Configuration bits.