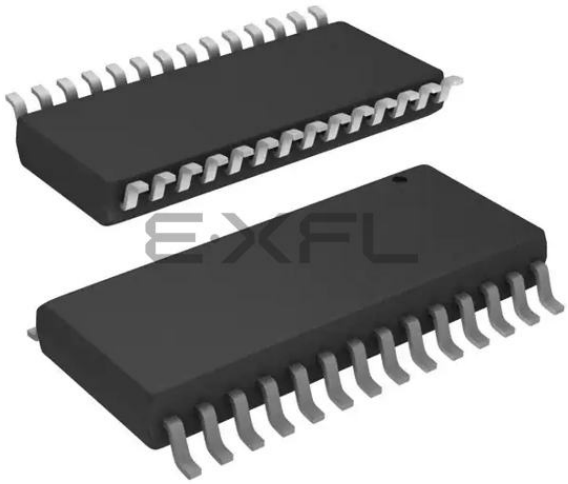


Welcome to E-XFL.COM

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	CANbus, I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	25
Program Memory Size	64KB (32K x 16)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	2.3V ~ 5.5V
Data Converters	A/D 24x12b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	28-SOIC (0.295", 7.50mm Width)
Supplier Device Package	28-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f26k83-e-so

3.1.1 PRIORITY LOCK

The System arbiter grants memory access to the peripheral selections (DMAx, Scanner) when the PRLOCKED bit (PRLOCK Register) is set.

Priority selections are locked by setting the PRLOCKED bit of the PRLOCK register. Setting and clearing this bit requires a special sequence as an extra precaution against inadvertent changes. Examples of setting and clearing the PRLOCKED bit are shown in Example 3-1 and Example 3-2.

EXAMPLE 3-1: PRIORITY LOCK SEQUENCE

```
; Disable interrupts
BCF INTCON0,GIE

; Bank to PRLOCK register
BANKSEL PRLOCK
MOVLW 55h

; Required sequence, next 4
instructions
MOVWF PRLOCK
MOVLW AAh
MOVWF PRLOCK
; Set PRLOCKED bit to grant memory
access to peripherals
BSF PRLOCK,0

; Enable Interrupts
BSF INTCON0,GIE
```

EXAMPLE 3-2: PRIORITY UNLOCK SEQUENCE

```
; Disable interrupts
BCF INTCON0,GIE

; Bank to PRLOCK register
BANKSEL PRLOCK
MOVLW 55h

; Required sequence, next 4
instructions
MOVWF PRLOCK
MOVLW AAh
MOVWF PRLOCK
; Clear PRLOCKED bit to allow changing
priority settings
BCF PRLOCK,0

; Enable Interrupts
BSF INTCON0,GIE
```

3.2 Memory Access Scheme

The user can assign priorities to both system level and peripheral selections based on which the system arbiter grants memory access. Let us consider the following priority scenarios between ISR, MAIN, and Peripherals.

Note:	It is always required that the ISR priority be higher than Main priority.

3.2.1 ISR PRIORITY > MAIN PRIORITY > PERIPHERAL PRIORITY

When the Peripheral Priority (DMAx, Scanner) is lower than ISR and MAIN Priority, and the peripheral requires:

1. Access to the Program Flash Memory, then the peripheral waits for an instruction cycle in which the CPU does not need to access the PFM (such as a branch instruction) and uses that cycle to do its own Program Flash Memory access, unless a PFM Read/Write operation is in progress.
2. Access to the SFR/GPR, then the peripheral waits for an instruction cycle in which the CPU does not need to access the SFR/GPR (such as `MOVLW`, `CALL`, `NOP`) and uses that cycle to do its own SFR/GPR access.
3. Access to the Data EEPROM, then the peripheral has access to Data EEPROM unless a Data EEPROM Read/Write operation is being performed.

This results in the lowest throughput for the peripheral to access the memory, and does so without any impact on execution times.

3.2.2 PERIPHERAL PRIORITY > ISR PRIORITY > MAIN PRIORITY

When the Peripheral Priority (DMAx, Scanner) is higher than ISR and MAIN Priority, the CPU operation is stalled when the peripheral requests memory.

The CPU is held in its current state until the peripheral completes its operation. Since the peripheral requests access to the bus, the peripheral cannot be disabled until it completes its operation.

This results in the highest throughput for the peripheral to access the memory, but has the cost of stalling other execution while it occurs.

TABLE 4-3: SPECIAL FUNCTION REGISTER MAP FOR PIC18(L)F25/26K83 DEVICES BANK 63

3FFFh	TOSU	3FDFh	INDF2	3FBFh	—	3F9Fh	T4PR	3F7Fh	CCP1CAP	3F5Fh	CCPTMRS1	3F3Fh	NCO1CLK	3F1Fh	SMT1CON1
3FFEh	TOSH	3FDEh	POSTINC2	3FBEh	—	3F9Eh	T4TMR	3F7Eh	CCP1CON	3F5Eh	CCPTMRS0	3F3Eh	NCO1CON	3F1Eh	SMT1CON0
3FFDh	TOSL	3FDDh	POSTDEC2	3FBDh	—	3F9Dh	T5CLK	3F7Dh	CCPR1H	3F5Dh	—	3F3Dh	NCO1INCU	3F1Dh	SMT1PRU
3FFCh	STKPTR	3FDCh	PRECIN2	3FBCh	LATC	3F9Ch	T5GATE	3F7Ch	CCPR1L	3F5Ch	—	3F3Ch	NCO1INCH	3F1Ch	SMT1PRH
3FFBh	PCLATU	3FDBh	PLUSW2	3FBBh	LATB	3F9Bh	T5GCON	3F7Bh	CCP2CAP	3F5Bh	—	3F3Bh	NCO1INCL	3F1Bh	SMT1PRL
3FFAh	PCLATH	3FDAh	FSR2H	3FBAh	LATA	3F9Ah	T5CON	3F7Ah	CCP2CON	3F5Ah	CWG1STR	3F3Ah	NCO1ACCU	3F1Ah	SMT1CPWU
3FF9h	PCL	3FD9h	FSR2L	3FB9h	T0CON1	3F99h	TMR5H	3F79h	CCPR2H	3F59h	CWG1AS1	3F39h	NCO1ACCH	3F19h	SMT1CPWH
3FF8h	TBLPRTU	3FD8h	STATUS	3FB8h	T0CON0	3F98h	TMR5L	3F78h	CCPR2L	3F58h	CWG1AS0	3F38h	NCO1ACCL	3F18h	SMT1CPWL
3FF7h	TBLPTRH	3FD7h	IVTBASEU	3FB7h	TMR0H	3F97h	T6RST	3F77h	CCP3CAP	3F57h	CWG1CON1	3F37h	—	3F17h	SMT1CPRU
3FF6h	TBLPTRL	3FD6h	IVTBASEH	3FB6h	TMR0L	3F96h	T6CLK	3F76h	CCP3CON	3F56h	CWG1CON0	3F36h	—	3F16h	SMT1CPRH
3FF5h	TABLAT	3FD5h	IVTBASEL	3FB5h	T1CLK	3F95h	T6HLT	3F75h	CCPR3H	3F55h	CWG1DBF	3F35h	—	3F15h	SMT1CPRL
3FF4h	PRODH	3FD4h	IVTLOCK	3FB4h	T1GATE	3F94h	T6CON	3F74h	CCPR3L	3F54h	CWG1DBR	3F34h	—	3F14h	SMT1TMRU
3FF3h	PRODL	3FD3h	INTCON1	3FB3h	T1GCON	3F93h	T6PR	3F73h	CCP4CAP	3F53h	CWG1ISM	3F33h	—	3F13h	SMT1TMRH
3FF2h	—	3FD2h	INTCON0	3FB2h	T1CON	3F92h	T6TMR	3F72h	CCP4CON	3F52h	CWG1CLK	3F32h	—	3F12h	SMT1TMRL
3FF1h	PCON1	3FD1h	—	3FB1h	TMR1H	3F91h	ECANCON	3F71h	CCPR4H	3F51h	CWG2STR	3F31h	—	3F11h	SMT2WIN
3FF0h	PCON0	3FD0h	—	3FB0h	TMR1L	3F90h	COMSTAT	3F70h	CCPR4L	3F50h	CWG2AS1	3F30h	—	3F10h	SMT2SIG
3FEFh	INDF0	3FCFh	—	3FAFh	T2RST	3F8Fh	CANCON	3F6Fh	—	3F4Fh	CWG2AS0	3F2Fh	—	3F0Fh	SMT2CLK
3FEEh	POSTINC0	3FCEh	PORTE	3FAEh	T2CLK	3F8Eh	CANSTAT	3F6Eh	PWM5CON	3F4Eh	CWG2CON1	3F2Eh	—	3F0Eh	SMT2STAT
3FEDh	POSTDEC0	3FCDh	—	3FADh	T2HLT	3F8Dh	RXB0D7	3F6Dh	PWM5DCH	3F4Dh	CWG2CON0	3F2Dh	—	3F0Dh	SMT2CON1
3FECh	PRECIN0	3FCCh	PORTC	3FACh	T2CON	3F8Ch	RXB0D6	3F6Ch	PWM5DCL	3F4Ch	CWG2DBF	3F2Ch	—	3F0Ch	SMT2CON0
3FEBh	PLUSW0	3FCBh	PORTB	3FABh	T2PR	3F8Bh	RXB0D5	3F6Bh	—	3F4Bh	CWG2DBR	3F2Bh	—	3F0Bh	SMT2PRU
3FEAh	FSR0H	3FCAh	PORTA	3FAAh	T2TMR	3F8Ah	RXB0D4	3F6Ah	PWM6CON	3F4Ah	CWG2ISM	3F2Ah	—	3F0Ah	SMT2PRH
3FE9h	FSR0L	3FC9h	—	3FA9h	T3CLK	3F89h	RXB0D3	3F69h	PWM6DCH	3F49h	CWG2CLK	3F29h	—	3F09h	SMT2PRL
3FE8h	WREG	3FC8h	—	3FA8h	T3GATE	3F88h	RXB0D2	3F68h	PWM6DCL	3F48h	CWG3STR	3F28h	—	3F08h	SMT2CPWU
3FE7h	INDF1	3FC7h	—	3FA7h	T3GCON	3F87h	RXB0D1	3F67h	—	3F47h	CWG3AS1	3F27h	—	3F07h	SMT2CPWH
3FE6h	POSTINC1	3FC6h	—	3FA6h	T3CON	3F86h	RXB0D0	3F66h	PWM7CON	3F46h	CWG3AS0	3F26h	—	3F06h	SMT2CPWL
3FE5h	POSTDEC1	3FC5h	—	3FA5h	TMR3H	3F85h	RXB0DLC	3F65h	PWM7DCH	3F45h	CWG3CON1	3F25h	—	3F05h	SMT2CPRU
3FE4h	PRECIN1	3FC4h	TRISC	3FA4h	TMR3L	3F84h	RXB0EIDL	3F64h	PWM7DCL	3F44h	CWG3CON0	3F24h	—	3F04h	SMT2CPRH
3FE3h	PLUSW1	3FC3h	TRISB	3FA3h	T4RST	3F83h	RXB0EIDH	3F63h	—	3F43h	CWG3DBF	3F23h	SMT1WIN	3F03h	SMT2CPRL
3FE2h	FSR1H	3FC2h	TRISA	3FA2h	T4CLK	3F82h	RXB0SIDL	3F62h	PWM8CON	3F42h	CWG3DBR	3F22h	SMT1SIG	3F02h	SMT2TMRU
3FE1h	FSR1L	3FC1h	—	3FA1h	T4HLT	3F81h	RXB0SIDH	3F61h	PWM8DCH	3F41h	CWG3ISM	3F21h	SMT1CLK	3F01h	SMT2TMRH
3FE0h	BSR	3FC0h	—	3FA0h	T4CON	3F80h	RXB0CON	3F60h	PWM8DCL	3F40h	CWG3CLK	3F20h	SMT1STAT	3F00h	SMT2TMRL

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

REGISTER 9-19: PIE6: PERIPHERAL INTERRUPT ENABLE REGISTER 6

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
DMA2AIE	DMA2ORIE	DMA2DCNTIE	DMA2SCNTIE	SMT2PWAIE	SMT2PRAIE	SMT2IE	C2IE
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 **DMA2AIE:** DMA Abort Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 6 **DMA2ORIE:** DMA2 Overrun Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 5 **DMA2DCNTIE:** DMA2 Destination Count Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 4 **DMA2SCNTIE:** DMA2 Source Count Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 3 **SMT2PWAIE:** SMT2 Pulse-Width Acquisition Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 2 **SMT2PRAIE:** SMT2 Period Acquisition Receive Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 1 **SMT2IE:** SMT2 Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 0 **C2IE:** C2 Interrupt Enable bit
1 = Enabled
0 = Disabled

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 13-3).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.

REGISTER 13-5: NVMDAT: DATA EEPROM MEMORY DATA

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
DAT<7:0>							
bit 7							bit 0

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

bit 7-0 **DAT<7:0>:** The value of the data memory word returned from NVMADR after a Read command, or the data written by a Write command.

TABLE 13-5: SUMMARY OF REGISTERS ASSOCIATED WITH NONVOLATILE MEMORY CONTROL

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
NVMCON1	REG<1:0>		—	FREE	WRERR	WREN	WR	RD	200
NVMCON2	Unlock Pattern								201
NVMADRL	NVMADR<7:0>								201
NVMADRH	—	—	—	—	—	—	NVMADR<9:8>		201
NVMDAT	NVMDAT<7:0>								202

Legend: — = unimplemented, read as '0'. Shaded bits are not used during EEPROM access.

*Page provides register information.

REGISTER 14-9: CRCXORH: CRC XOR HIGH BYTE REGISTER

R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x
X<15:8>							
bit 7							bit 0

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

bit 7-0 **X<15:8>**: XOR of Polynomial Term X^n Enable bits

REGISTER 14-10: CRCXORL: CRC XOR LOW BYTE REGISTER

R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x	U-1
X<7:1>							—
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit
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 7-1 **X<7:1>**: XOR of Polynomial Term X^n Enable bits

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

REGISTER 15-3: DMAxBUF: DMAx DATA BUFFER REGISTER

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
BUF7	BUF6	BUF5	BUF4	BUF3	BUF2	BUF1	BUF0
bit 7							bit 0

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

bit 7-0 **BUF<7:0>:** DMA Internal Data Buffer bits
DMABUF<7:0>
These bits reflect the content of the internal data buffer the DMA peripheral uses to hold the data being moved from the source to destination.

REGISTER 15-4: DMAxSSAL: DMAx SOURCE START ADDRESS LOW REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
SSA<7:0>							
bit 7							bit 0

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

bit 7-0 **SSA<7:0>:** Source Start Address bits

REGISTER 15-5: DMAxSSAH: DMAx SOURCE START ADDRESS HIGH REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
SSA<15:8>							
bit 7							bit 0

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

bit 7-0 **SSA<15:8>:** Source Start Address bits

20.3 Programmable Prescaler

A software programmable prescaler is available for exclusive use with Timer0. There are 16 prescaler options for Timer0 ranging in powers of two from 1:1 to 1:32768. The prescaler values are selected using the CKPS<3:0> bits of the T0CON1 register.

The prescaler is not directly readable or writable. Clearing the prescaler register can be done by writing to the TMR0L register or to the T0CON0/T0CON1 register or by any Reset.

20.4 Programmable Postscaler

A software programmable postscaler (output divider) is available for exclusive use with Timer0. There are 16 postscaler options for Timer0 ranging from 1:1 to 1:16. The postscaler values are selected using the OUTPS bits of the T0CON0 register.

The postscaler is not directly readable or writable. Clearing the postscaler register can be done by writing to the TMR0L register or to the T0CON0/T0CON1 register or by any Reset.

20.5 Operation During Sleep

When operating synchronously, Timer0 will halt. When operating asynchronously, Timer0 will continue to increment and wake the device from Sleep (if Timer0 interrupts are enabled) provided that the input clock source is active.

20.6 Timer0 Interrupts

The Timer0 interrupt flag bit (TMR0IF) is set when either of the following conditions occur:

- 8-bit TMR0L matches the TMR0H value
- 16-bit TMR0 rolls over from ‘FFFFh’

When the postscaler bits (OUTPS) are set to 1:1 operation (no division), the T0IF flag bit will be set with every TMR0 match or rollover. In general, the TMR0IF flag bit will be set every OUTPS +1 matches or rollovers.

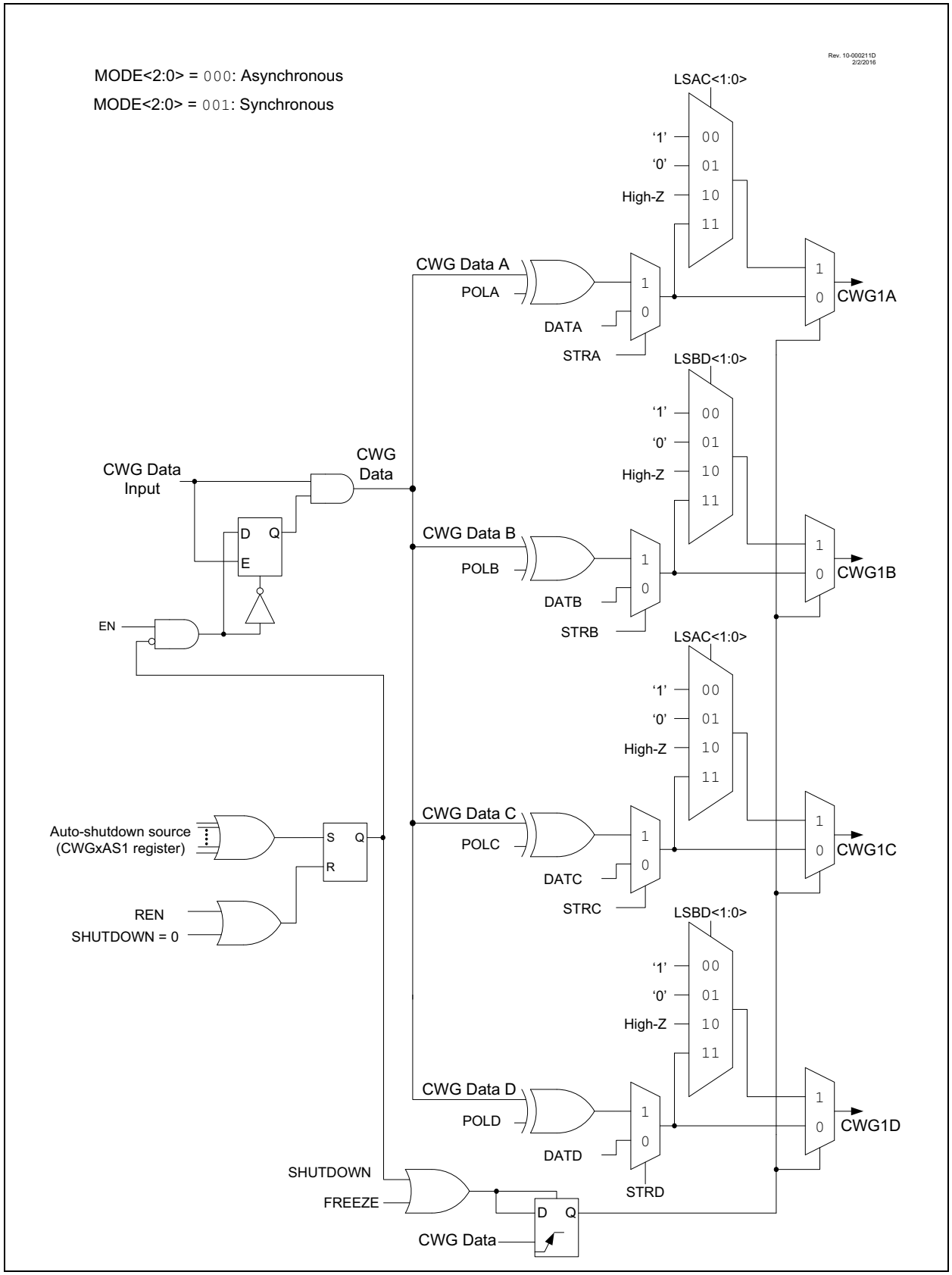
If Timer0 interrupts are enabled (TMR0IE bit of the PIE3 register = ‘1’), the CPU will be interrupted and the device may wake from Sleep (see **Section 20.2 “Clock Source Selection”** for more details).

20.7 Timer0 Output

The Timer0 output can be routed to any I/O pin via the RxyPPS output selection register (see **Section 17.0 “Peripheral Pin Select (PPS) Module”** for additional information). The Timer0 output can also be used by other peripherals, such as the auto-conversion trigger of the Analog-to-Digital Converter. Finally, the Timer0 output can be monitored through software via the Timer0 output bit (OUT) of the T0CON0 register (Register 20-1).

TMR0_out will be a pulse of one postscaled clock period when a match occurs between TMR0L and PR0 (Period register for TMR0) in 8-bit mode, or when TMR0 rolls over in 16-bit mode. The Timer0 output is a 50% duty cycle that toggles on each TMR0_out rising clock edge.

FIGURE 26-11: SIMPLIFIED CWG BLOCK DIAGRAM (OUTPUT STEERING MODES)



26.10 Auto-Shutdown

Auto-shutdown is a method to immediately override the CWG output levels with specific overrides that allow for safe shutdown of the circuit. The shutdown state can be either cleared automatically or held until cleared by software. The auto-shutdown circuit is illustrated in Figure 26-14.

26.10.1 SHUTDOWN

The shutdown state can be entered by either of the following two methods:

- Software generated
- External Input

26.10.1.1 Software Generated Shutdown

Setting the SHUTDOWN bit of the CWGxAS0 register will force the CWG into the shutdown state.

When the auto-restart is disabled, the shutdown state will persist as long as the SHUTDOWN bit is set.

When auto-restart is enabled, the SHUTDOWN bit will clear automatically and resume operation on the next rising edge event. The SHUTDOWN bit indicates when a shutdown condition exists. The bit may be set or cleared in software or by hardware.

26.10.1.2 External Input Source

External shutdown inputs provide the fastest way to safely suspend CWG operation in the event of a Fault condition. When any of the selected shutdown inputs goes active, the CWG outputs will immediately go to the specified override levels without software delay. The override levels are selected by the LSBD<1:0> and LSAC<1:0> bits of the CWGxAS0 register (Register 26-6). Several input sources can be selected to cause a shutdown condition. All input sources are active-low. The sources are:

- Pin selected by CWGxPPS
- Timer2 postscaled output
- Timer4 postscaled output
- Timer6 postscaled output
- Comparator 1 output
- Comparator 2 output
- CLC2 output

Shutdown input sources are individually enabled by the ASxE bits of the CWGxAS1 register (Register 26-7).

Note:	Shutdown inputs are level sensitive, not edge sensitive. The shutdown state cannot be cleared, except by disabling auto-shutdown, as long as the shutdown input level persists.

26.10.1.3 Pin Override Levels

The levels driven to the CWG outputs during an auto-shutdown event are controlled by the LSBD<1:0> and LSAC<1:0> bits of the CWGxAS0 register (Register 26-6). The LSBD<1:0> bits control CWGxB/D output levels, while the LSAC<1:0> bits control the CWGxA/C output levels.

26.10.1.4 Auto-Shutdown Interrupts

When an auto-shutdown event occurs, either by software or hardware setting SHUTDOWN, the CWGxIF flag bit of the respective PIR register is set.

26.11 Auto-Shutdown Restart

After an auto-shutdown event has occurred, there are two ways to resume operation:

- Software controlled
- Auto-restart

In either case, the shutdown source must be cleared before the restart can take place. That is, either the shutdown condition must be removed, or the corresponding ASxE bit must be cleared.

26.11.1 SOFTWARE-CONTROLLED RESTART

If the REN bit of the CWGxAS0 register is clear (REN = 0), the CWG module must be restarted after an auto-shutdown event through software.

Once all auto-shutdown sources are removed, the software must clear SHUTDOWN. Once SHUTDOWN is cleared, the CWG module will resume operation upon the first rising edge of the CWG data input.

Note:	The SHUTDOWN bit cannot be cleared in software if the auto-shutdown condition is still present.

26.11.2 AUTO-RESTART

If the REN bit of the CWGxAS0 register is set (REN = 1), the CWG module will restart from the shutdown state automatically.

Once all auto-shutdown conditions are removed, the hardware will automatically clear SHUTDOWN. Once SHUTDOWN is cleared, the CWG module will resume operation upon the first rising edge of the CWG data input.

Note:	The SHUTDOWN bit cannot be cleared in software if the auto-shutdown condition is still present.

REGISTER 27-8: CLCxGLS1: GATE 1 LOGIC SELECT REGISTER

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
G2D4T	G2D4N	G2D3T	G2D3N	G2D2T	G2D2N	G2D1T	G2D1N
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 **G2D4T:** Gate 1 Data 4 True (noninverted) bit
1 = CLCIN3 (true) is gated into CLCx Gate 1
0 = CLCIN3 (true) is not gated into CLCx Gate 1

bit 6 **G2D4N:** Gate 1 Data 4 Negated (inverted) bit
1 = CLCIN3 (inverted) is gated into CLCx Gate 1
0 = CLCIN3 (inverted) is not gated into CLCx Gate 1

bit 5 **G2D3T:** Gate 1 Data 3 True (noninverted) bit
1 = CLCIN2 (true) is gated into CLCx Gate 1
0 = CLCIN2 (true) is not gated into CLCx Gate 1

bit 4 **G2D3N:** Gate 1 Data 3 Negated (inverted) bit
1 = CLCIN2 (inverted) is gated into CLCx Gate 1
0 = CLCIN2 (inverted) is not gated into CLCx Gate 1

bit 3 **G2D2T:** Gate 1 Data 2 True (noninverted) bit
1 = CLCIN1 (true) is gated into CLCx Gate 1
0 = CLCIN1 (true) is not gated into CLCx Gate 1

bit 2 **G2D2N:** Gate 1 Data 2 Negated (inverted) bit
1 = CLCIN1 (inverted) is gated into CLCx Gate 1
0 = CLCIN1 (inverted) is not gated into CLCx Gate 1

bit 1 **G2D1T:** Gate 1 Data 1 True (noninverted) bit
1 = CLCIN0 (true) is gated into CLCx Gate 1
0 = CLCIN0 (true) is not gated into CLCx Gate1

bit 0 **G2D1N:** Gate 1 Data 1 Negated (inverted) bit
1 = CLCIN0 (inverted) is gated into CLCx Gate 1
0 = CLCIN0 (inverted) is not gated into CLCx Gate 1

30.1 DSM Operation

The DSM module can be enabled by setting the EN bit in the MD1CON0 register. Clearing the EN bit in the MD1CON0 register, disables the DSM module output and switches the carrier high and carrier low signals to the default option of MD1CARHPPS and MD1CARLPPS, respectively. The modulator signal source is also switched to the BIT in the MD1CON0 register.

The values used to select the carrier high, carrier low, and modulator sources held by the Modulation Source, Modulation High Carrier, and Modulation Low Carrier control registers are not affected when the EN bit is cleared and the DSM module is disabled. The values inside these registers remain unchanged while the DSM is inactive. The sources for the carrier high, carrier low and modulator signals will once again be selected when the EN bit is set and the DSM module is again enabled and active.

30.2 Modulator Signal Sources

The modulator signal can be supplied from the sources specified in Table 30-3.

The modulator signal is selected by configuring the MS<4:0> bits in the MD1SRC register.

30.3 Carrier Signal Sources

The carrier high signal and carrier low signal can be supplied from the sources specified in Table 30-1.

The carrier high signal is selected by configuring the CH<4:0> bits in the MD1CARH register. The carrier low signal is selected by configuring the CL<4:0> bits in the MD1CARL register.

30.4 Carrier Synchronization

During the time when the DSM switches between carrier high and carrier low signal sources, the carrier data in the modulated output signal can become truncated. To prevent this, the carrier signal can be synchronized to the modulator signal. When synchronization is enabled, the carrier pulse that is being mixed at the time of the transition is allowed to transition low before the DSM switches over to the next carrier source.

Synchronization is enabled separately for the carrier high and carrier low signal sources. Synchronization for the carrier high signal is enabled by setting the CHSYNC bit in the MD1CON1 register. Synchronization for the carrier low signal is enabled by setting the CLSYNC bit in the MD1CON1 register.

Figure 30-2 through Figure 30-6 show timing diagrams of using various synchronization methods.

TABLE 30-2: MD1SRC SELECTION MUX CONNECTIONS

MS<4:0>		Connection
1 1111 - 1 1000	31 - 24	Reserved
1 0111	23	CAN_tx0
1 0110	22	SPI1 SDO
1 0101	21	Reserved
1 0100	20	UART2 TX
1 0011	19	UART1 TX
1 0010	18	CLC4 OUT
1 0001	17	CLC3 OUT
1 0000	16	CLC2 OUT
0 1111	15	CLC1 OUT
0 1110	14	CMP2 OUT
0 1101	13	CMP1 OUT
0 1100	12	NCO1 OUT
0 1011	11	Reserved
0 1010	10	Reserved
0 1001	9	PWM8 OUT
0 1000	8	PWM7 OUT
0 0111	7	PWM6 OUT
0 0110	6	PWM5 OUT
0 0101	5	CCP4 OUT
0 0100	4	CCP3 OUT
0 0011	3	CCP2 OUT
0 0010	2	CCP1 OUT
0 0001	1	DSM1 BIT
0 0000	0	Pin selected by MDSRCPPS

TABLE 30-3: SUMMARY OF REGISTERS ASSOCIATED WITH DATA SIGNAL MODULATOR MODE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
MD1CON0	EN	—	OUT	OPOL	—	—	—	BIT	455
MD1CON1	—	—	CHPOL	CHSYNC	—	—	CLPOL	CLSYNC	456
MD1CARH	—	—	—	—	—	CHS<2:0>			457
MD1CARL	—	—	—	—	—	CLS<2:0>			457
MDSRC	—	—	—	—	SRCS<3:0>				458

Legend: — = unimplemented, read as '0'. Shaded cells are not used in the Data Signal Modulator mode.

31.21 Register Definitions: UART Control

Long bit name prefixes for the UART peripherals are shown below. Refer to **Section 1.3 "Register and Bit naming conventions"**for more information.

Peripheral	Bit Name Prefix
UART 1	U1
UART 2	U2

REGISTER 31-1: UxCON0: UART CONTROL REGISTER 0

R/W-0/0	R/W/HS/HC-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
BRGS	ABDEN	TXEN	RXEN	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 Resets
'1' = Bit is set	'0' = Bit is cleared	HC = Hardware clear

bit 7 **BRGS:** Baud rate Generator Speed Select bit
- 1 = Baud rate generator is high speed with 4 baud clocks per bit
- 0 = Baud rate generator is normal speed with 16 baud clocks per bit

bit 6 **ABDEN:** Auto-baud Detect Enable bit(3)
- 1 = Auto-baud is enabled. Receiver is waiting for Sync character (0x55)
- 0 = Auto-baud is not enabled or auto-baud is complete

bit 5 **TXEN:** Transmit Enable Control bit(2)
- 1 = Transmit is enabled. TX output pin drive is forced on when transmission is active, and controlled by PORT TRIS control when transmission is idle.
- 0 = Transmit is disabled. TX output pin drive is controlled by PORT TRIS control

bit 4 **RXEN:** Receive Enable Control bit(2)
- 1 = Receiver is enabled
- 0 = Receiver is disabled

bit 3-0 **MODE<3:0>:** UART Mode Select bits(1)
- 1111 = Reserved
- 1110 = Reserved
- 1101 = Reserved
- 1100 = LIN Master/Slave mode
- 1011 = LIN Slave-Only mode
- 1010 = DMX mode
- 1001 = DALI Control Gear mode
- 1000 = DALI Control Device mode
- 0111 = Reserved
- 0110 = Reserved
- 0101 = Reserved
- 0100 = Asynchronous 9-bit UART Address mode. 9th bit: 1 = address, 0 = data
- 0011 = Asynchronous 8-bit UART mode with 9th bit even parity
- 0010 = Asynchronous 8-bit UART mode with 9th bit odd parity
- 0001 = Asynchronous 7-bit UART mode
- 0000 = Asynchronous 8-bit UART mode

Note 1: Changing the UART MODE while ON = 1 may cause unexpected results.
2: Clearing TXEN or RXEN will not clear the corresponding buffers. Use TXBE or RXBE to clear the buffers.
3: ABDEN is read-only when MODE = 1001. When MODE = 100x and ABDEN = 1, then auto-baud is determined from Start bit.

REGISTER 31-14: UxP2H: UART PARAMETER 2 HIGH REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0/0
—	—	—	—	—	—	—	P2<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 7-6 **Unimplemented:** Read as '0'

bit 0 **P2<8>:** Most Significant Bit of Parameter 2
DMX mode:
Most Significant bit of first address of receive block
DALI mode:
Most Significant bit of number of half-bit periods of idle time in Forward Frame detection threshold
Other modes:
Not used

REGISTER 31-15: UxP2L: UART PARAMETER 2 LOW REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
P2<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 **P2<7:0>:** Least Significant Bits of Parameter 2
DMX mode:
Least Significant Byte of first address of receive block
LIN Slave mode:
Number of data bytes to transmit
DALI mode:
Least Significant Byte of number of half-bit periods of idle time in Forward Frame detection threshold
Asynchronous Address mode:
Receiver address
Other modes:
Not used

REGISTER 37-5: ADSTAT: ADC STATUS REGISTER

R-0/0	R-0/0	R-0/0	R/HS/HC-0/0	U-0	R-0/0	R-0/0	R-0/0
AOV	UTHR	LTHR	MATH	—	STAT<2:0>		
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	HS/HC = Bit is set/cleared by hardware

bit 7 **AOV**: ADC Accumulator Overflow bit
1 = ADC accumulator or ERR calculation have overflowed
0 = ADC accumulator and ERR calculation have not overflowed

bit 6 **UTHR**: ADC Module Greater-than Upper Threshold Flag bit
1 = ERR >UTH
0 = ERR ≤ UTH

bit 5 **LTHR**: ADC Module Less-than Lower Threshold Flag bit
1 = ERR < LTH
0 = ERR ≥ LTH

bit 4 **MATH**: ADC Module Computation Status bit
1 = Registers ACC, FLTR, UTH, LTH and the AOV bit are updating or have already updated
0 = Associated registers/bits have not changed since this bit was last cleared

bit 3 **Unimplemented:** Read as '0'

bit 2-0 **STAT<2:0>**: ADC Module Cycle Multistage Status bits(1)
111 = ADC module is in 2nd conversion stage
110 = ADC module is in 2nd acquisition stage
101 = ADC module is in 2nd precharge stage
100 = Not used
011 = ADC module is in 1st conversion stage
010 = ADC module is in 1st acquisition stage
001 = ADC module is in 1st precharge stage
000 = ADC module is not converting

Note 1: If CS = 1, and FOSC<FRC, these bits may be invalid.

MOVSF **Move Indexed to f**

Syntax: MOVSF [z_s], f_d

Operands: $0 \le z_s \le 127$
$0 \le f_d \le 4095$

Operation: ((FSR2) + z_s) → f_d

Status Affected: None

Encoding:

1st word (source)	1110	1011	0zzz	$zzzz_s$
2nd word (destin.)	1111	ffff	ffff	$ffff_d$

Description: The contents of the source register are moved to destination register 'f_d'. The actual address of the source register is determined by adding the 7-bit literal offset 'z_s' in the first word to the value of FSR2. The address of the destination register is specified by the 12-bit literal 'f_d' in the second word. Both addresses can be anywhere in the 4096-byte data space (000h to FFFh).
MOVSF has curtailed the destination range to the lower 4 Kbyte space in memory (Banks 1 through 15). For everything else, use MOVSFL.

Words: 2

Cycles: 2

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Determine source addr	Determine source addr	Read source reg
Decode	No operation No dummy read	No operation	Write register 'f' (dest)

Example: MOVSF [05h], REG2

Before Instruction
FSR2 = 80h
Contents of 85h = 33h
REG2 = 11h
After Instruction
FSR2 = 80h
Contents of 85h = 33h
REG2 = 33h

MOVSFL **Move Indexed to f (Long Range)**

Syntax: MOVSFL [z_s], f_d

Operands: $0 \le z_s \le 127$
$0 \le f_d \le 16383$

Operation: ((FSR2) + z_s) → f_d

Status Affected: None

Encoding:

1st word (opcode)	0000	0000	0110	0010
2nd word (source)	1111	xxxz	zzzz	zz_sff
3rd word (full destin.)	1111	ffff	ffff	$ffff_d$

Description: The contents of the source register are moved to destination register 'f_d'. The actual address of the source register is determined by adding the 7-bit literal offset 'z_s' in the first word to the value of FSR2 (14 bits). The address of the destination register is specified by the 14-bit literal 'f_d' in the second word. Both addresses can be anywhere in the 16 Kbyte data space (0000h to 3FFFh). The MOVSFL instruction cannot use the PCL, TOSU, TOSH or TOSL as the destination register. If the resultant source address points to an indirect addressing register, the value returned will be 00h.

Words: 3

Cycles: 3

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	No opera-tion	No operation	No operation
Decode	Read register "z" (src.)	Process data	No operation
Decode	No opera-tion No dummy read	No operation	Write register "f" (dest.)

Example: MOVSFL [05h], REG2

Before Instruction
FSR2 = 80h
Contents of 85h = 33h
REG2 = 11h
After Instruction
FSR2 = 80h
Contents of 85h = 33h

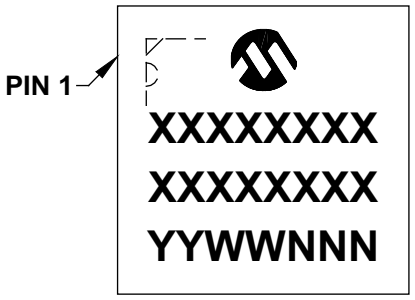
TABLE 43-1: REGISTER FILE SUMMARY FOR PIC18(L)F25/26K83 DEVICES (CONTINUED)

Addr	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
3A15h	RC5PPS	—	—	—	RC5PPS4	RC5PPS3	RC5PPS2	RC5PPS1	RC5PPS0	266
3A14h	RC4PPS	—	—	—	RC4PPS4	RC4PPS3	RC4PPS2	RC4PPS1	RC4PPS0	266
3A13h	RC3PPS	—	—	—	RC3PPS4	RC3PPS3	RC3PPS2	RC3PPS1	RC3PPS0	266
3A12h	RC2PPS	—	—	—	RC2PPS4	RC2PPS3	RC2PPS2	RC2PPS1	RC2PPS0	266
3A11h	RC1PPS	—	—	—	RC1PPS4	RC1PPS3	RC1PPS2	RC1PPS1	RC1PPS0	266
3A10h	RC0PPS	—	—	—	RC0PPS4	RC0PPS3	RC0PPS2	RC0PPS1	RC0PPS0	266
3A0Fh	RB7PPS	—	—	—	RB7PPS4	RB7PPS3	RB7PPS2	RB7PPS1	RB7PPS0	266
3A0Eh	RB6PPS	—	—	—	RB6PPS4	RB6PPS3	RB6PPS2	RB6PPS1	RB6PPS0	266
3A0Dh	RB5PPS	—	—	—	RB5PPS4	RB5PPS3	RB5PPS2	RB5PPS1	RB5PPS0	266
3A0Ch	RB4PPS	—	—	—	RB4PPS4	RB4PPS3	RB4PPS2	RB4PPS1	RB4PPS0	266
3A0Bh	RB3PPS	—	—	—	RB3PPS4	RB3PPS3	RB3PPS2	RB3PPS1	RB3PPS0	266
3A0Ah	RB2PPS	—	—	—	RB2PPS4	RB2PPS3	RB2PPS2	RB2PPS1	RB2PPS0	266
3A09h	RB1PPS	—	—	—	RB1PPS4	RB1PPS3	RB1PPS2	RB1PPS1	RB1PPS0	266
3A08h	RB0PPS	—	—	—	RB0PPS4	RB0PPS3	RB0PPS2	RB0PPS1	RB0PPS0	266
3A07h	RA7PPS	—	—	—	RA7PPS4	RA7PPS3	RA7PPS2	RA7PPS1	RA7PPS0	266
3A06h	RA6PPS	—	—	—	RA6PPS4	RA6PPS3	RA6PPS2	RA6PPS1	RA6PPS0	266
3A05h	RA5PPS	—	—	—	RA5PPS4	RA5PPS3	RA5PPS2	RA5PPS1	RA5PPS0	266
3A04h	RA4PPS	—	—	—	RA4PPS4	RA4PPS3	RA4PPS2	RA4PPS1	RA4PPS0	266
3A03h	RA3PPS	—	—	—	RA3PPS4	RA3PPS3	RA3PPS2	RA3PPS1	RA3PPS0	266
3A02h	RA2PPS	—	—	—	RA2PPS4	RA2PPS3	RA2PPS2	RA2PPS1	RA2PPS0	266
3A01h	RA1PPS	—	—	—	RA1PPS4	RA1PPS3	RA1PPS2	RA1PPS1	RA1PPS0	266
3A00h	RA0PPS	—	—	—	RA0PPS4	RA0PPS3	RA0PPS2	RA0PPS1	RA0PPS0	266
39FFh - 39F8h	—	Unimplemented								—
39F7h	SCANPR	—	—	—	—	—	PR			21
39F6h - 39F5h	—	Unimplemented								—
39F4h	DMA2PR	—	—	—	—	—	PR			21
39F3h	DMA1PR	—	—	—	—	—	PR			20
39F2h	MAINPR	—	—	—	—	—	PR			20
39F1h	ISRPR	—	—	—	—	—	PR			20
39F0h	—	Unimplemented								—
39EFh	PRLOCK	—	—	—	—	—	—	—	PRLOCKED	21
39EEh - 39E7h	—	Unimplemented								—
39E6h	NVMCON2	NVMCON2								201
39E5h	NVMCON1	REG		—	FREE	WRERR	WREN	WR	RD	200
39E4h	—	Unimplemented								—
39E3h	NVMDAT	DAT								202
39E2h	—	Unimplemented								—
39E1h	—	Unimplemented								—
39E0h	NVMADRL	ADR								201
39DFh	OSCFRQ	—	—	—	—	FRQ				97
39DEh	OSCTUNE	—	—	TUN						98
39DDh	OSCEN	EXTOEN	HFOEN	MFOEN	LFOEN	SOSCEN	ADOEN	—	—	99
39DCh	OSCSTAT	EXTOR	HFOR	MFOR	LFOR	SOR	ADOR	—	PLLR	96
39DBh	OSCCON3	CSWHOLD	SOSCPWR	—	ORDY	NOSCR	—	—	—	95
39DAh	OSCCON2	—	COSC			CDIV				95
39D9h	OSCCON1	—	NOSC			NDIV				94
39D8h	CPUDOZE	IDLEN	DOZEN	ROI	DOE	—	DOZE			167

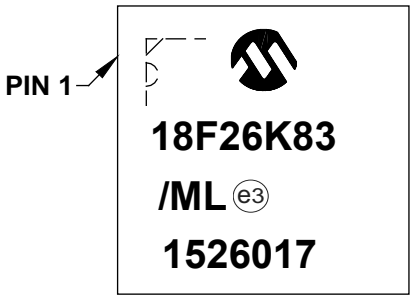
Legend: x = unknown, u = unchanged, — = unimplemented, q = value depends on condition
Note 1: Not present in LF devices.

Package Marking Information (Continued)

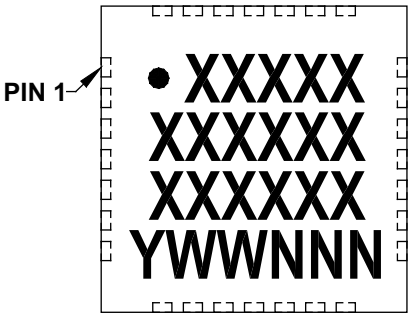
28-Lead QFN (6x6 mm)



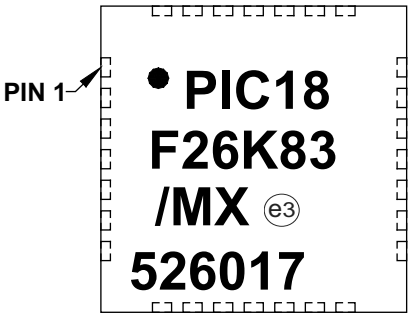
Example



28-Lead UQFN (6x6x0.5 mm)



Example

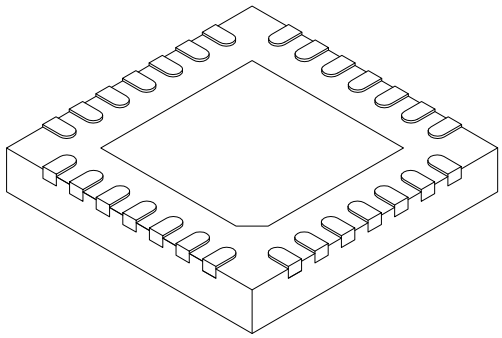


Legend:	XX...X	Customer-specific information or Microchip part number
	Y	Year code (last digit of calendar year)
	YY	Year code (last 2 digits of calendar year)
	WW	Week code (week of January 1 is week ‘01’)
	NNN	Alphanumeric traceability code
	(e3)	Pb-free JEDEC® designator for Matte Tin (Sn)
	*	This package is Pb-free. The Pb-free JEDEC designator ((e3)) can be found on the outer packaging for this package.

Note: In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line, thus limiting the number of available characters for customer-specific information.

28-Lead Plastic Quad Flat, No Lead Package (ML) - 6x6 mm Body [QFN]
With 0.55 mm Terminal Length

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



	Units	MILLIMETERS		
Dimension	Limits	MIN	NOM	MAX
Number of Pins	N		28	
Pitch	e		0.65 BSC	
Overall Height	A	0.80	0.90	1.00
Standoff	A1	0.00	0.02	0.05
Terminal Thickness	A3		0.20 REF	
Overall Width	E		6.00 BSC	
Exposed Pad Width	E2	3.65	3.70	4.20
Overall Length	D		6.00 BSC	
Exposed Pad Length	D2	3.65	3.70	4.20
Terminal Width	b	0.23	0.30	0.35
Terminal Length	L	0.50	0.55	0.70
Terminal-to-Exposed Pad	K	0.20	-	-

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated
3. Dimensioning and tolerancing per ASME Y14.5M.
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-105C Sheet 2 of 2