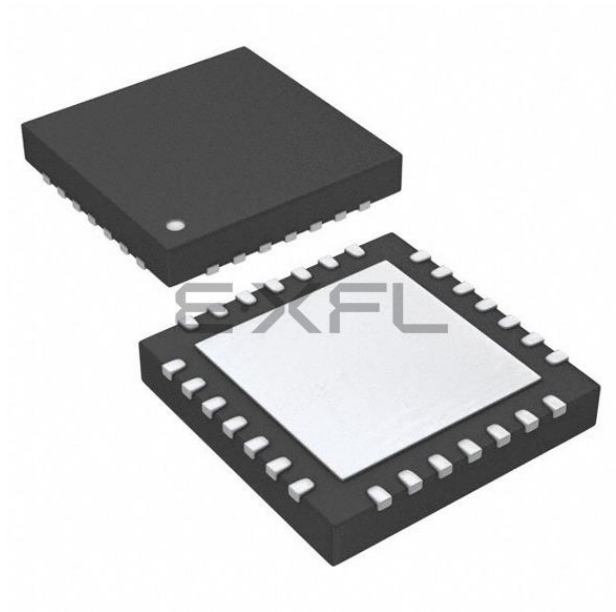


Welcome to E-XFL.COM

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, DMA, HLVD, 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)	1.8V ~ 3.6V
Data Converters	A/D 24x12b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-VQFN Exposed Pad
Supplier Device Package	28-QFN (6x6)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18lf26k42-i-ml

TABLE 4-11: SPECIAL FUNCTION REGISTER MAP FOR PIC18(L)F26/27/45/46/47/55/56/57K42 DEVICES BANK 56

38FFh	—	38DFh	—	38BFh	—	389Fh	IVTADU	387Fh	—	385Fh	—	383Fh	—	381Fh	—
38FEh	—	38DEh	—	38BEh	—	389Eh	IVTADH	387Eh	—	385Eh	—	383Eh	—	381Eh	—
38FDh	—	38DDh	—	38BDh	—	389Dh	IVTADL	387Dh	—	385Dh	—	383Dh	—	381Dh	—
38FCh	—	38DCh	—	38BCh	—	389Ch	—	387Ch	—	385Ch	—	383Ch	—	381Ch	—
38FBh	—	38DBh	—	38BBh	—	389Bh	—	387Bh	—	385Bh	—	383Bh	—	381Bh	—
38FAh	—	38DAh	—	38BAh	—	389Ah	—	387Ah	—	385Ah	—	383Ah	—	381Ah	—
38F9h	—	38D9h	—	38B9h	—	3899h	—	3879h	—	3859h	—	3839h	—	3819h	—
38F8h	—	38D8h	—	38B8h	—	3898h	—	3878h	—	3858h	—	3838h	—	3818h	—
38F7h	—	38D7h	—	38B7h	—	3897h	—	3877h	—	3857h	—	3837h	—	3817h	—
38F6h	—	38D6h	—	38B6h	—	3896h	—	3876h	—	3856h	—	3836h	—	3816h	—
38F5h	—	38D5h	—	38B5h	—	3895h	—	3875h	—	3855h	—	3835h	—	3815h	—
38F4h	—	38D4h	—	38B4h	—	3894h	—	3874h	—	3854h	—	3834h	—	3814h	—
38F3h	—	38D3h	—	38B3h	—	3893h	—	3873h	—	3853h	—	3833h	—	3813h	—
38F2h	—	38D2h	—	38B2h	—	3892h	—	3872h	—	3852h	—	3832h	—	3812h	—
38F1h	—	38D1h	—	38B1h	—	3891h	—	3871h	—	3851h	—	3831h	—	3811h	—
38F0h	—	38D0h	—	38B0h	—	3890h	PRODH_SHAD	3870h	—	3850h	—	3830h	—	3810h	—
38EFh	—	38CFh	—	38AFh	—	388Fh	PRODL_SHAD	386Fh	—	384Fh	—	382Fh	—	380Fh	—
38EEh	—	38CEh	—	38AEh	—	388Eh	FSR2H_SHAD	386Eh	—	384Eh	—	382Eh	—	380Eh	—
38EDh	—	38CDh	—	38ADh	—	388Dh	FSR2L_SHAD	386Dh	—	384Dh	—	382Dh	—	380Dh	—
38ECh	—	38CCh	—	38ACh	—	388Ch	FSR1H_SHAD	386Ch	—	384Ch	—	382Ch	—	380Ch	—
38EBh	—	38CBh	—	38ABh	—	388Bh	FSR1L_SHAD	386Bh	—	384Bh	—	382Bh	—	380Bh	—
38EAh	—	38CAh	—	38AAh	—	388Ah	FSR0H_SHAD	386Ah	—	384Ah	—	382Ah	—	380Ah	—
38E9h	—	38C9h	—	38A9h	—	3889h	FSR0L_SHAD	3869h	—	3849h	—	3829h	—	3809h	—
38E8h	—	38C8h	—	38A8h	—	3888h	PCLATU_SHAD	3868h	—	3848h	—	3828h	—	3808h	—
38E7h	—	38C7h	—	38A7h	—	3887h	PCLATH_SHAD	3867h	—	3847h	—	3827h	—	3807h	—
38E6h	—	38C6h	—	38A6h	—	3886h	BSR_SHAD	3866h	—	3846h	—	3826h	—	3806h	—
38E5h	—	38C5h	—	38A5h	—	3885h	WREG_SHAD	3865h	—	3845h	—	3825h	—	3805h	—
38E4h	—	38C4h	—	38A4h	—	3884h	STATUS_SHAD	3864h	—	3844h	—	3824h	—	3804h	—
38E3h	—	38C3h	—	38A3h	—	3883h	SHADCON	3863h	—	3843h	—	3823h	—	3803h	—
38E2h	—	38C2h	—	38A2h	—	3882h	BSR_CSHAD	3862h	—	3842h	—	3822h	—	3802h	—
38E1h	—	38C1h	—	38A1h	—	3881h	WREG_CSHAD	3861h	—	3841h	—	3821h	—	3801h	—
38E0h	—	38C0h	—	38A0h	—	3880h	STATUS_CSHAD	3860h	—	3840h	—	3820h	—	3800h	—

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.

6.1 Power-on Reset (POR)

The POR circuit holds the device in Reset until VDD has reached an acceptable level for minimum operation. Slow rising VDD, fast operating speeds or analog performance may require greater than minimum VDD. The PWRT, BOR or MCLR features can be used to extend the start-up period until all device operation conditions have been met.

6.2 Brown-out Reset (BOR)

The BOR circuit holds the device in Reset when VDD reaches a selectable minimum level. Between the POR and BOR, complete voltage range coverage for execution protection can be implemented.

The Brown-out Reset module has four operating modes controlled by the BOREN<1:0> bits in Configuration Words. The four operating modes are:

- BOR is always on
- BOR is off when in Sleep
- BOR is controlled by software
- BOR is always off

Refer to Table 6-1 for more information.

The Brown-out Reset voltage level is selectable by configuring the BORV<1:0> bits in Configuration Words.

A VDD noise rejection filter prevents the BOR from triggering on small events. If VDD falls below VBOR for a duration greater than parameter TBORDC, the device will reset. See Table 44-13 for more information.

6.2.1 BOR IS ALWAYS ON

When the BOREN bits of Configuration Words are programmed to '11', the BOR is always on. The device start-up will be delayed until the BOR is ready and VDD is higher than the BOR threshold.

BOR protection is active during Sleep. The BOR does not delay wake-up from Sleep.

6.2.2 BOR IS OFF IN SLEEP

When the BOREN bits of Configuration Words are programmed to '10', the BOR is on, except in Sleep. The device start-up will be delayed until the BOR is ready and VDD is higher than the BOR threshold.

BOR protection is not active during Sleep. The device wake-up will be delayed until the BOR is ready.

6.2.3 BOR CONTROLLED BY SOFTWARE

When the BOREN bits of Configuration Words are programmed to '01', the BOR is controlled by the SBOREN bit of the BORCON register. The device start-up is not delayed by the BOR ready condition or the VDD level.

BOR protection begins as soon as the BOR circuit is ready. The status of the BOR circuit is reflected in the BORRDY bit of the BORCON register.

BOR protection is unchanged by Sleep.

6.2.4 BOR AND BULK ERASE

BOR is forced ON during PFM Bulk Erase operations to make sure that a safe erase voltage is maintained for a successful erase cycle.

During Bulk Erase, the BOR is enabled at 2.45V for F and LF devices, even if it is configured to some other value. If VDD falls, the erase cycle will be aborted, but the device will not be reset.

REGISTER 9-2: INTCON1: INTERRUPT CONTROL REGISTER 1

R-0/0	R-0/0	U-0	U-0	U-0	U-0	U-0	U-0
STAT<1:0>		—	—	—	—	—	—
bit 7							bit 0

Legend:		
HC = Bit is cleared by hardware		
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 7-6 **STAT<1:0>:** Interrupt State Status bits

11 = High priority ISR executing, high priority interrupt was received while a low priority ISR was executing

10 = High priority ISR executing, high priority interrupt was received in main routine

01 = Low priority ISR executing, low priority interrupt was received in main routine

00 = Main routine executing

bit 5-0 **Unimplemented**: Read as ‘0’

REGISTER 9-20: 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
TMR3GIE	TMR3IE	U2IE	U2EIE	U2TXIE	U2RXIE	I2C2EIE	I2C2IE
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 **TMR3GIE:** TMR3 Gate Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 6 **TMR3IE:** TMR3 Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 5 **U2IE:** UART2 Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 4 **U2EIE:** UART2 Framing Error Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 3 **U2TXIE:** UART2 Transmit Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 2 **U2RXIE:** UART2 Receive Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 1 **I2C2EIE:** I^2C2 Error Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 0 **I2C2IE:** I^2C2 Interrupt Enable bit
1 = Enabled
0 = Disabled

11.7 Register Definitions: Windowed Watchdog Timer Control

REGISTER 11-1: WDTCON0: WATCHDOG TIMER CONTROL REGISTER 0

U-0	U-0	R/W(3)-q/q(2)	R/W(3)-q/q(2)	R/W(3)-q/q(2)	R/W(3)-q/q(2)	R/W(3)-q/q(2)	R/W-0/0
—	—	PS<4:0>					SEN
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 7-6 **Unimplemented:** Read as '0'

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

Bit Value = Prescale Rate

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

•

•

•

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

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

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

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

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

2: When WDTCPS <4:0> in CONFIG3L = 11111, the Reset value of PS<4:0> is 01011. Otherwise, the Reset value of PS<4:0> is equal to WDTCPS<4:0> in CONFIG3L.

3: When WDTCPS <4:0> in CONFIG3L ≠ 11111, these bits are read-only.

4: When the WWDT is configured to run using the SOSC as a clock source and the device is allowed to undergo a Reset, as triggered by a WDT time-out, the SOSC would also undergo a Reset. That means the SOSC will execute its start-up sequence which requires 1024 SOSC clock counts before it is made available for peripherals to use. So for example, if the WDT is set for a 1 ms time-out and the device is allowed to undergo a WDT Reset, then the actual WDT Reset period will be: WDT_PERIOD = (1/(SOSC_FREQUENCY) * 1024) + 1 ms.

TABLE 11-3: SUMMARY OF REGISTERS ASSOCIATED WITH WINDOWED WATCHDOG TIMER

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
WDTCON0	—	—	PS<4:0>					SEN	182
WDTCON1	—	CS<2:0>			—	WINDOW<2:0>			183
WDTPSL	PSCNT<7:0>								184
WDTPSH	PSCNT<15:8>								184
WDTTMR	WDTTMR<4:0>					STATE	PSCNT<17:16>		185

Legend: x = unknown, u = unchanged, – = unimplemented locations read as '0'. Shaded cells are not used by Windowed Watchdog Timer.

16.2.6 INPUT THRESHOLD CONTROL

The INLVLx register (Register 16-8) controls the input voltage threshold for each of the available PORTx input pins. A selection between the Schmitt Trigger CMOS or the TTL compatible thresholds is available. The input threshold is important in determining the value of a read of the PORTx register and also the level at which an interrupt-on-change occurs, if that feature is enabled. See Table 44-6 for more information on threshold levels.

Note:	Changing the input threshold selection should be performed while all peripheral modules are disabled. Changing the threshold level during the time a module is active may inadvertently generate a transition associated with an input pin, regardless of the actual voltage level on that pin.

16.2.7 WEAK PULL-UP CONTROL

The WPUx register (Register 16-5) controls the individual weak pull-ups for each port pin.

16.2.8 EDGE SELECTABLE INTERRUPT-ON-CHANGE

An interrupt can be generated by detecting a signal at the port pin that has either a rising edge or a falling edge. Any individual pin can be configured to generate an interrupt. The interrupt-on-change module is present on all the pins. For further details about the IOC module refer to **Section 18.0 "Interrupt-on-Change"**.

16.2.9 I^2C PAD CONTROL

For the PIC18(L)F26/27/45/46/47/55/56/57K42 devices, the I^2C specific pads are available on RB1, RB2, RC3, RC4, RD0**(1)** and RD1**(1)** pins. The I^2C characteristics of each of these pins is controlled by the RxyI2C registers (see Register 16-9). These characteristics include enabling I^2C specific slew rate (over standard GPIO slew rate), selecting internal pull-ups for I^2C pins, and selecting appropriate input threshold as per SMBus specifications.

Note 1:	RD0 and RD1 I^2C pads are not available in PIC18(L)F26K42 parts.
2:	Any peripheral using the I^2C pins read the I^2C ST inputs when enabled via RxyI2C.

.

16.3 PORTE Registers

Depending on the device, PORTE is implemented in two different ways.

16.3.1 PORTE ON 40/44/48-PIN DEVICES

For PIC18(L)F45/46/47/55/56/57K42 devices, PORTE is a 4-bit wide port. Three pins (RE0, RE1 and RE2) are individually configurable as inputs or outputs. These pins have Schmitt Trigger input buffers. When selected as an analog input, these pins will read as '0's. The corresponding data direction register is TRISE. Setting a TRISE bit (= 1) will make the corresponding PORTE pin an input (i.e., disable the output driver).
Clearing a TRISE bit (= 0) will make the corresponding PORTE pin an output (i.e., enable the output driver and put the contents of the output latch on the selected pin). TRISE controls the direction of the REx pins, even when they are being used as analog pins. The user must make sure to keep the pins configured as inputs when using them as analog inputs. RE<2:0> bits have other registers associated with them (i.e., ANSELE, WPUE, INLVLE, SLRCONE and ODCONE). The functionality is similar to the other ports. The Data Latch register (LATE) is also memory-mapped. Read-modify-write operations on the LATE register read and write the latched output value for PORTE.

Note:	On a Power-on Reset, RE<2:0> are configured as analog inputs.

The fourth pin of PORTE (MCLR/VPP/RE3) is an input-only pin. Its operation is controlled by the MCLRE Configuration bit. When selected as a port pin, (MCLRE = 0), it functions as a digital input-only pin; as such, it does not have TRIS or LAT bits associated with its operation. Otherwise, it functions as the device's Master Clear input. In either configuration, RE3 also functions as the programming voltage input during programming. RE3 in PORTE register is a read-only bit and will read '1' when MCLRE = 1 (i.e., Master Clear enabled).

Note:	On a Power-on Reset, RE3 is enabled as a digital input only if Master Clear functionality is disabled.

REGISTER 16-7: SLRCONx: SLEW RATE CONTROL REGISTER

R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
SLRx7	SLRx6	SLRx5	SLRx4	SLRx3	SLRx2	SLRx1	SLRx0
bit 7							bit 0

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

bit 7-0 **SLRx<7:0>**: Slew Rate Control on Pins Rx<7:0>, respectively

1 = Port pin slew rate is limited
0 = Port pin slews at maximum rate

TABLE 16-8: SLEW RATE CONTROL REGISTERS

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SLRCONA	SLRA7	SLRA6	SLRA5	SLRA4	SLRA3	SLRA2	SLRA1	SLRA0
SLRCONB	SLRB7	SLRB6	SLRB5	SLRB4	SLRB3	SLRB2	SLRB1	SLRB0
SLRCONC	SLRC7	SLRC6	SLRC5	SLRC4	SLRC3	SLRC2	SLRC1	SLRC0
SLRCOND(1)	SLRD7	SLRD6	SLRD5	SLRD4	SLRD3	SLRD2	SLRD1	SLRD0
SLRCONE(1)	—	—	—	—	—	SLRE2	SLRE1	SLRE0
SLRCONF(2)	SLRF7	SLRF6	SLRF5	SLRF4	SLRF3	SLRF2	SLRF1	SLRF0

Note 1: Unimplemented in PIC18(L)F26/27K42.
2: Unimplemented in PIC18(L)F26/27/45/46/47K42.

REGISTER 19-6: PMD5: PMD CONTROL REGISTER 5

U-0	U-0	R/W-0/0	R/W-0/0	U-0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	U2MD	U1MD	—	SPI1MD	I2C2MD	I2C1MD
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 7-6 **Unimplemented:** Read as '0'

bit 5 **U2MD:** Disable UART2 bit
1 = UART2 module disabled
0 = UART2 module enabled

bit 4 **U1MD:** Disable UART1 bit
1 = UART1 module disabled
0 = UART1 module enabled

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

bit 2 **SPI1MD:** Disable SPI1 Module bit
1 = SPI1 module disabled
0 = SPI1 module enabled

bit 1 **I2C2MD:** Disable I^2C2 Module bit
1 = I^2C2 module disabled
0 = I^2C2 module enabled

bit 0 **I2C1MD:** Disable I^2C1 Module bit
1 = I^2C1 module disabled
0 = I^2C1 module enabled

21.3 Timer1/3/5 Prescaler

Timer1/3/5 has four prescaler options allowing 1, 2, 4 or 8 divisions of the clock input. The CKPS bits of the TxCON register control the prescale counter. The prescale counter is not directly readable or writable; however, the prescaler counter is cleared upon a write to TMRxH or TMRxL.

21.4 Timer1/3/5 Operation in Asynchronous Counter Mode

If control bit $\overline{\text{SYNC}}$ of the TxCON register is set, the external clock input is not synchronized. The timer increments asynchronously to the internal phase clocks. If external clock source is selected then the timer will continue to run during Sleep and can generate an interrupt on overflow, which will wake up the processor. However, special precautions in software are needed to read/write the timer (see **Section 21.4.1 "Reading and Writing Timer1/3/5 in Asynchronous Counter Mode"**).

Note:	When switching from synchronous to asynchronous operation, it is possible to skip an increment. When switching from asynchronous to synchronous operation, it is possible to produce an additional increment.

21.4.1 READING AND WRITING TIMER1/3/5 IN ASYNCHRONOUS COUNTER MODE

Reading TMRxH or TMRxL while the timer is running from an external asynchronous clock will ensure a valid read (taken care of in hardware). However, the user should keep in mind that reading the 16-bit timer in two 8-bit values itself, poses certain problems, since the timer may overflow between the reads. For writes, it is recommended that the user simply stop the timer and write the desired values. A write contention may occur by writing to the timer registers, while the register is incrementing. This may produce an unpredictable value in the TMRxH:TMRxL register pair.

21.5 Timer1/3/5 16-Bit Read/Write Mode

Timer1/3/5 can be configured to read and write all 16 bits of data, to and from, the 8-bit TMRxL and TMRxH registers, simultaneously. The 16-bit read and write operations are enabled by setting the RD16 bit of the TxCON register.

To accomplish this function, the TMRxH register value is mapped to a buffer register called the TMRxH buffer register. While in 16-Bit mode, the TMRxH register is not directly readable or writable and all read and write operations take place through the use of this TMRxH buffer register.

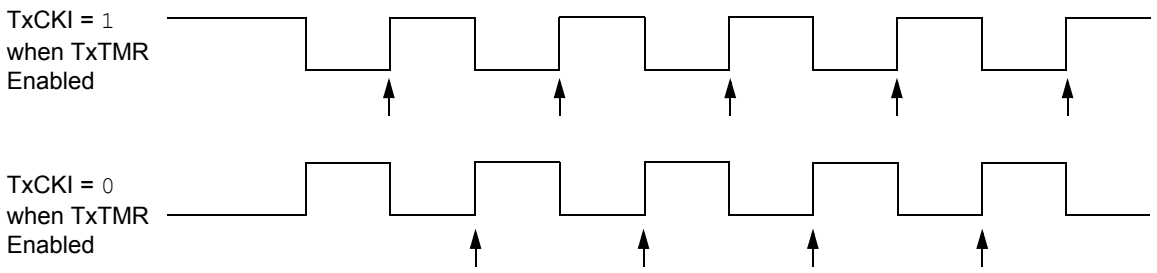
When a read from the TMRxL register is requested, the value of the TMRxH register is simultaneously loaded into the TMRxH buffer register. When a read from the TMRxH register is requested, the value is provided from the TMRxH buffer register instead. This provides the user with the ability to accurately read all 16 bits of the Timer1/3/5 value from a single instance in time. Reference the block diagram in Figure 21-2 for more details.

In contrast, when not in 16-Bit mode, the user must read each register separately and determine if the values have become invalid due to a rollover that may have occurred between the read operations.

When a write request of the TMRxL register is requested, the TMRxH buffer register is simultaneously updated with the contents of the TMRxH register. The value of TMRxH must be preloaded into the TMRxH buffer register prior to the write request for the TMRxL register. This provides the user with the ability to write all 16 bits to the TMRxL:TMRxH register pair at the same time.

Any requests to write to the TMRxH directly does not clear the Timer1/3/5 prescaler value. The prescaler value is only cleared through write requests to the TMRxL register.

FIGURE 21-3: TIMER1/3/5 INCREMENTING EDGE



Note 1: Arrows indicate counter increments.

2: In Counter mode, a falling edge must be registered by the counter prior to the first incrementing rising edge of the clock.

FIGURE 21-4: TIMER1/3/5 GATE ENABLE MODE

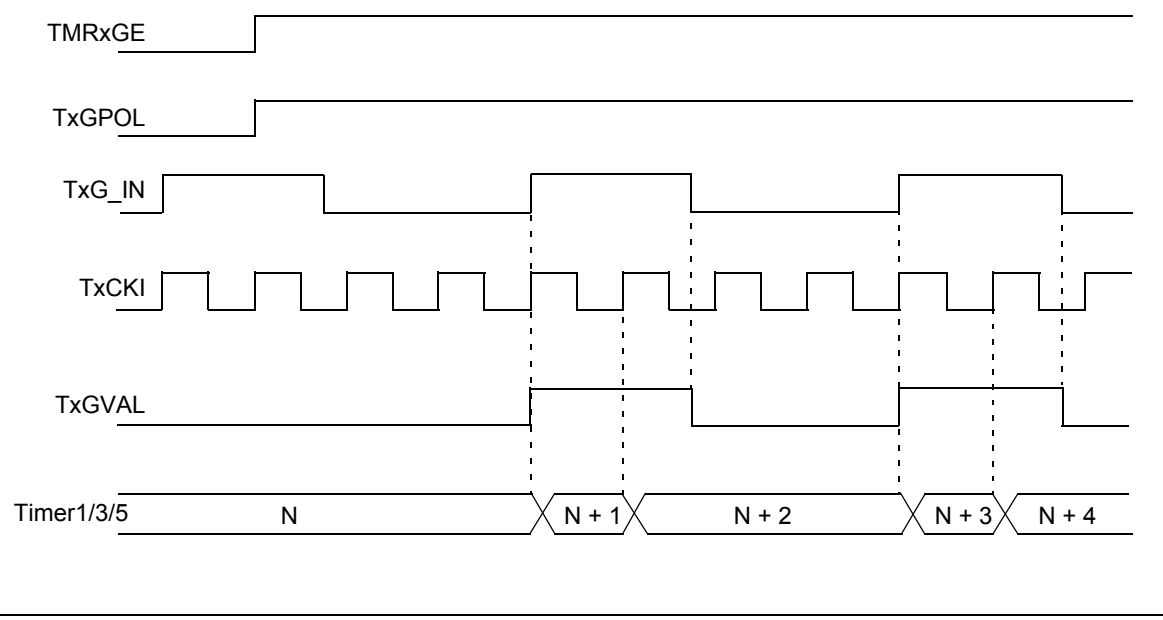
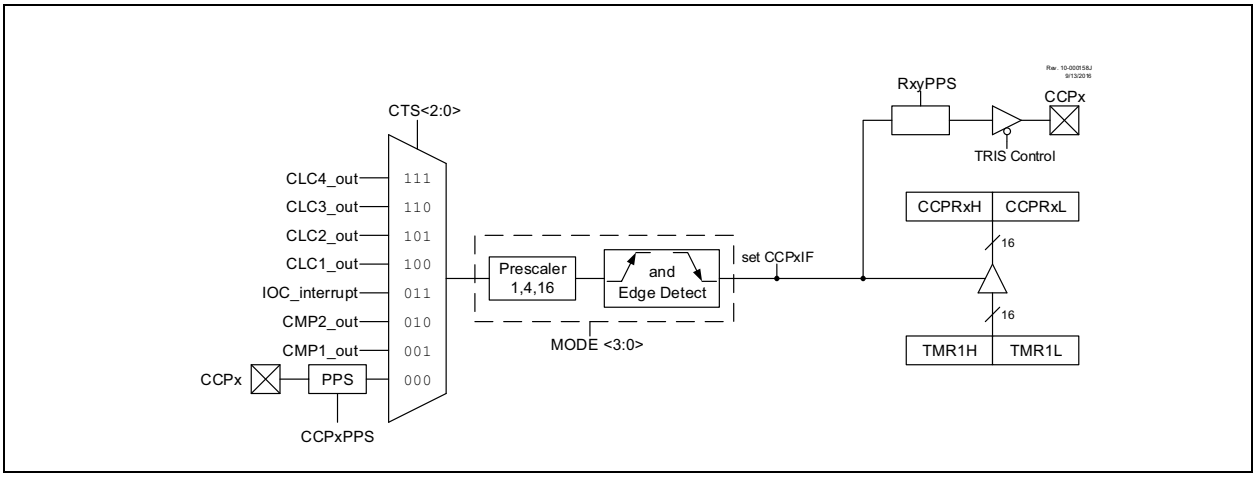


FIGURE 23-1: CAPTURE MODE OPERATION BLOCK DIAGRAM



REGISTER 24-3: PWMxDCH: PWM DUTY CYCLE HIGH BITS

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
DC<9:2>							
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 **DC<9:2>:** PWM Duty Cycle Most Significant bits
These bits are the MSbs of the PWM duty cycle. The two LSbs are found in PWMxDCL Register.

REGISTER 24-4: PWMxDCL: PWM DUTY CYCLE LOW BITS

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

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

bit 7-6 **DC<1:0>:** PWM Duty Cycle Least Significant bits
These bits are the LSbs of the PWM duty cycle. The MSbs are found in PWMxDCH Register.

bit 5-0 **Unimplemented:** Read as ‘0’

TABLE 24-3: SUMMARY OF REGISTERS ASSOCIATED WITH PWM

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
PWMxCON	EN	—	OUT	POL	—	—	—	—	358
PWMxDCH	DC<9:2>								360
PWMxDCL	DC<1:0>		—	—	—	—	—	—	360
CCPTMRS1	P8TSEL<1:0>		P7TSEL<1:0>		P6TSEL<1:0>		P5TSEL<1:0>		359

Legend: – = Unimplemented locations, read as ‘0’, u = unchanged, x = unknown. Shaded cells are not used by the PWM.

26.14 Register Definitions: CWG Control

Long bit name prefixes for the CWG peripheral is shown below. Refer to **Section 1.3.2.2 "Long Bit Names"** for more information.

Peripheral	Bit Name Prefix
CWG1	CWG1
CWG2	CWG2
CWG3	CWG3

REGISTER 26-1: CWGxCON0: CWG CONTROL REGISTER 0

R/W-0/0	R/W/HC-0/0	U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0
EN	LD(1)	—	—	—	MODE<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	HC = Bit is cleared by hardware

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

bit 6 **LD:** CWGx Load Buffers bit(1)
- 1 = Dead-band count buffers to be loaded on CWG data rising edge, following first falling edge after this bit is set
- 0 = Buffers remain unchanged

bit 5-3 **Unimplemented**: Read as '0'

bit 2-0 **MODE<2:0>**: CWGx Mode bits
- 111 = Reserved
- 110 = Reserved
- 101 = CWG outputs operate in Push-Pull mode
- 100 = CWG outputs operate in Half-Bridge mode
- 011 = CWG outputs operate in Reverse Full-Bridge mode
- 010 = CWG outputs operate in Forward Full-Bridge mode
- 001 = CWG outputs operate in Synchronous Steering mode
- 000 = CWG outputs operate in Asynchronous Steering mode

Note 1: This bit can only be set after EN = 1; it cannot be set in the same cycle when EN is set.

TABLE 31-3: SUMMARY OF REGISTERS ASSOCIATED WITH THE UART

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
UxCON0	BRGS	ABDEN	TXEN	RXEN	MODE<3:0>				498
UxCON1	ON	—	—	WUE	RXBIMD	—	BRKOVR	SENDB	499
UxCON2	RUNOVF	RXPOL	STP<1:0>		C0EN	TXPOL	FLO<1:0>		500
UxERRIR	TXMTIF	PERIF	ABDOVF	CERIF	FERIF	RXBKIF	RXFOIF	TXCIF	501
UxERRIE	TXMTIE	PERIE	ABDOVE	CERIE	FERIE;	RXBKIE	RXFOIE	TXCIE	502
UxUIR	WUIF	ABDIF	—	—	—	ABDIE	—	—	503
UxFIFO	TXWRE	STPMD	TXBE	TXBF	RXIDL	XON	RXBE	RXBF	504
UxBRGL	BRG<7:0>								505
UxBRGH	BRG<15:8>								505
UxRXB	RXB<7:0>								506
UxTXB	TXB<7:0>								506
UxP1H	—	—	—	—	—	—	—	P1<8>	507
UxP1L	P1<7:0>								507
UxP2H	—	—	—	—	—	—	—	P2<8>	508
UxP2L	P2<7:0>								508
UxP3H	—	—	—	—	—	—	—	P3<8>	509
UxP3L	P3<7:0>								509
UxTXCHK	TXCHK<7:0>								510
UxRXCHK	RXCHK<7:0>								510

Legend: — = unimplemented, read as '0'. Shaded cells are unused by the UART module.

34.0 FIXED VOLTAGE REFERENCE (FVR)

The Fixed Voltage Reference, or FVR, is a stable voltage reference, independent of VDD, with 1.024V, 2.048V or 4.096V selectable output levels. The output of the FVR can be configured to supply a reference voltage to the following:

- ADC input channel
- ADC positive reference
- Comparator input
- Digital-to-Analog Converter (DAC)

The FVR can be enabled by setting the EN bit of the FVRCON register.

Note:	Fixed Voltage Reference output cannot exceed VDD.

34.1 Independent Gain Amplifiers

The output of the FVR, which is connected to the ADC, Comparators, and DAC, is routed through two independent programmable gain amplifiers. Each amplifier can be programmed for a gain of 1x, 2x or 4x, to produce the three possible voltage levels.

The ADFVR<1:0> bits of the FVRCON register are used to enable and configure the gain amplifier settings for the reference supplied to the ADC module. Reference **Section 36.0 "Analog-to-Digital Converter with Computation (ADC2) Module"** for additional information.

The CDAFVR<1:0> bits of the FVRCON register are used to enable and configure the gain amplifier settings for the reference supplied to the DAC and comparator module. Reference **Section 37.0 "5-Bit Digital-to-Analog Converter (DAC) Module"** and **Section 38.0 "Comparator Module"** for additional information.

34.2 FVR Stabilization Period

When the Fixed Voltage Reference module is enabled, it requires time for the reference and amplifier circuits to stabilize. Once the circuits stabilize and are ready for use, the RDY bit of the FVRCON register will be set.

FIGURE 34-1: VOLTAGE REFERENCE BLOCK DIAGRAM

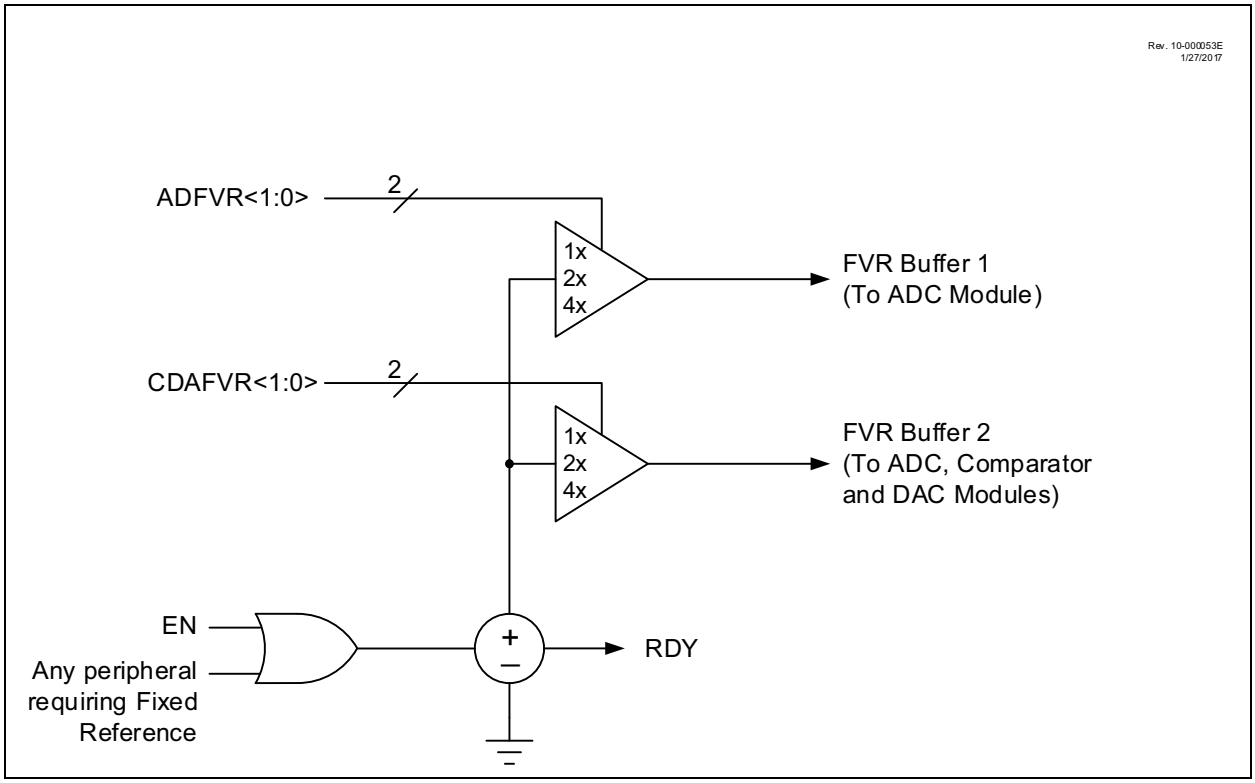
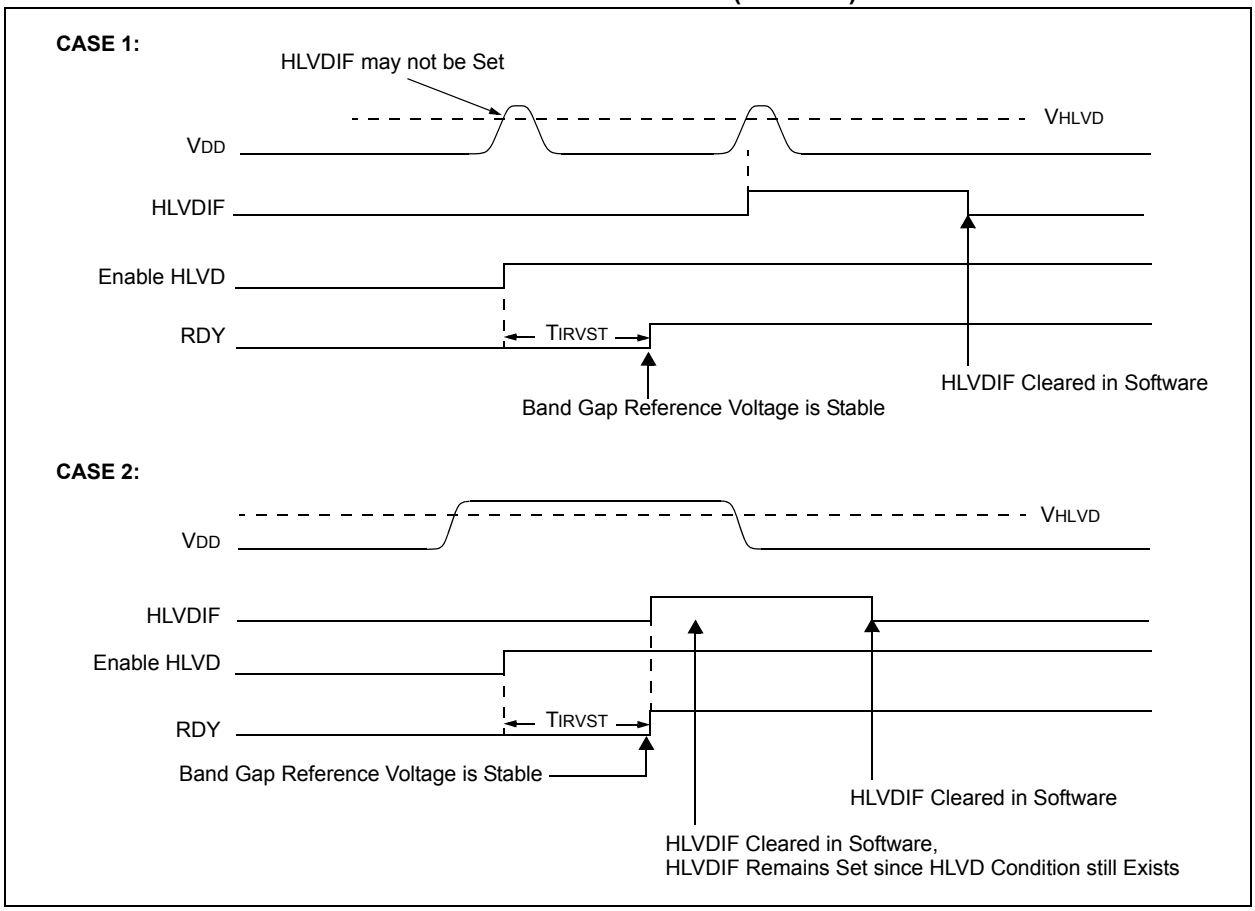


FIGURE 39-3: HIGH-VOLTAGE DETECT OPERATION (INTH = 1)

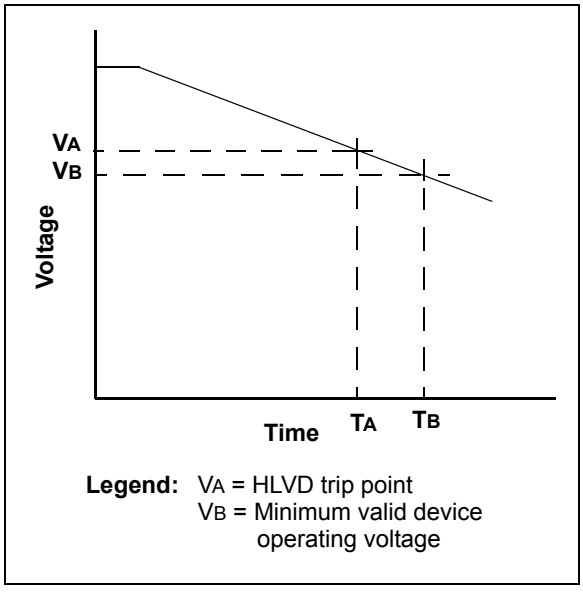


39.5 Applications

In many applications, it is desirable to detect a drop below, or rise above, a particular voltage threshold. For example, the HLVD module could be periodically enabled to detect Universal Serial Bus (USB) attach or detach. This assumes the device is powered by a lower voltage source than the USB when detached. An attach would indicate a High-Voltage Detect from, for example, 3.3V to 5V (the voltage on USB) and vice versa for a detach. This feature could save a design a few extra components and an attach signal (input pin).

For general battery applications, Figure 39-4 shows a possible voltage curve. Over time, the device voltage decreases. When the device voltage reaches voltage, VA, the HLVD logic generates an interrupt at time, TA. The interrupt could cause the execution of an Interrupt Service Routine (ISR), which would allow the application to perform “housekeeping tasks” and a controlled shutdown before the device voltage exits the valid operating range at TB. This would give the application a time window, represented by the difference between TA and TB, to safely exit.

FIGURE 39-4: TYPICAL LOW-VOLTAGE DETECT APPLICATION



REGISTER 39-2: HLVDCON1: LOW-VOLTAGE DETECT CONTROL REGISTER 1

U-0	U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—	—	SEL<3:0>			
bit 7							bit 0

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

bit 7-4 **Unimplemented:** Read as '0'

bit 3-0 **SEL<3:0>:** High/Low Voltage Detection Limit Selection bits
Refer to Table 44-14 for voltage detection limits.

TABLE 39-2: SUMMARY OF REGISTERS ASSOCIATED WITH HIGH/LOW-VOLTAGE DETECT MODULE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
HLVDCON0	EN	—	OUT	RDY	—	—	INTH	INTL	657
HLVDCON1	—	—	—	—	SEL<3:0>				658

Legend: — = unimplemented, read as '0'. Shaded cells are unused by the HLVD module.

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 operation	No operation	No operation
Decode	Read register "z" (src.)	Process data	No operation
Decode	No operation 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 44-26: I^2C BUS DATA REQUIREMENTS

Standard Operating Conditions (unless otherwise stated)							
Param. No.	**Symbol**	**Characteristic**		**Min.**	**Max.**	**Units**	**Conditions**
SP100*	THIGH	Clock high time	100 kHz mode	4000	—	ns	Device must operate at a minimum of 1.5 MHz
			400 kHz mode	600	—	ns	Device must operate at a minimum of 10 MHz
			1 MHz module	260	—	ns	Device must operate at a minimum of 10 MHz
SP101*	TLOW	Clock low time	100 kHz mode	4700	—	ns	Device must operate at a minimum of 1.5 MHz
			400 kHz mode	1300	—	ns	Device must operate at a minimum of 10 MHz
			1 MHz module	500	—	—	Device must operate at a minimum of 10 MHz
SP102*	TR	SDA and SCL rise time	100 kHz mode	—	1000	ns	
			400 kHz mode	20	300	ns	CB is specified to be from 10-400 pF
			1 MHz module	—	120	ns	
SP103*	TF	SDA and SCL fall time	100 kHz mode	—	250	ns	
			400 kHz mode	20 X (VDD/ 5.5V)	250	ns	CB is specified to be from 10-400 pF
			1 MHz module	20 X (VDD/ 5.5V)	120	ns	
SP106*	THD:DAT	Data input hold time	100 kHz mode	0	—	ns	
			400 kHz mode	0	—	ns	
			1 MHz module	0	—	ns	
SP107*	TSU:DAT	Data input setup time	100 kHz mode	250	—	ns	(2)
			400 kHz mode	100	—	ns	
			1 MHz module	50	—	ns	
SP109*	TAA	Output valid from clock	100 kHz mode	—	3450	ns	(1)
			400 kHz mode	—	900	ns	
			1 MHz module	—	450	ns	
SP110*	TBUF	Bus free time	100 kHz mode	4700	—	ns	Time the bus must be free before a new transmission can start
			400 kHz mode	1300	—	ns	
			1 MHz module	500	—	ns	
SP111	CB	Bus capacitive loading		—	400	pF	

* These parameters are characterized but not tested.

Note 1: As a transmitter, the device must provide this internal minimum delay time to bridge the undefined region (min. 300 ns) of the falling edge of SCL to avoid unintended generation of Start or Stop conditions.

2: A Fast mode (400 kHz) I^2C bus device can be used in a Standard mode (100 kHz) I^2C bus system, but the requirement $T_{SU:DAT} \geq 250$ ns must then be met. This will automatically be the case if the device does not stretch the low period of the SCL signal. If such a device does stretch the low period of the SCL signal, it must output the next data bit to the SDA line T_R max. + $T_{SU:DAT}$ = 1000 + 250 = 1250 ns (according to the Standard mode I^2C bus specification), before the SCL line is released.