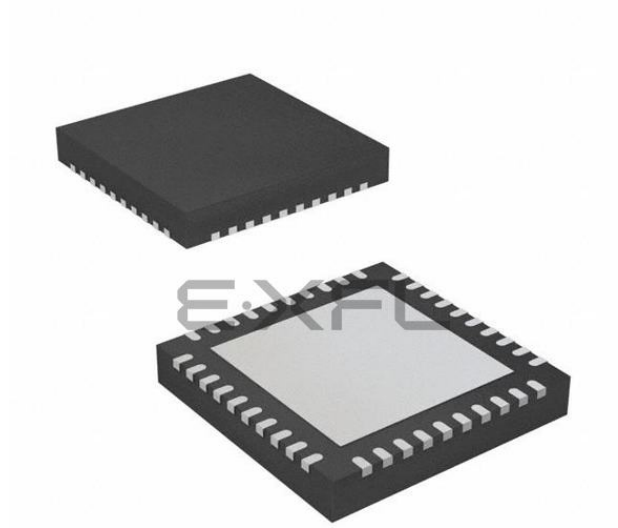


Welcome to E-XFL.COM

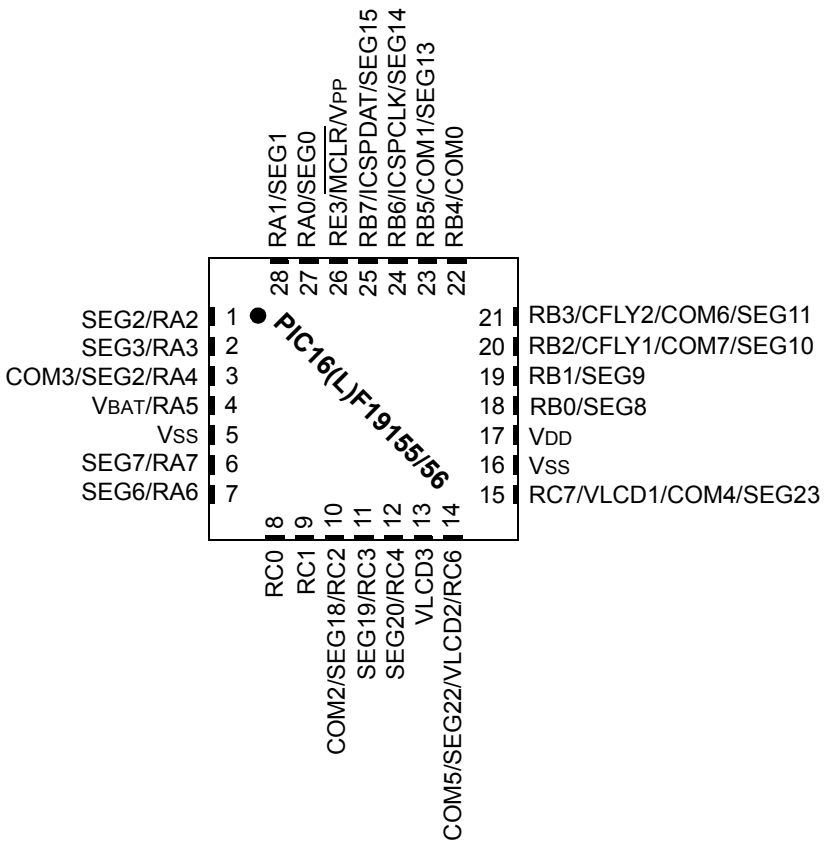
What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	32MHz
Connectivity	I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LCD, POR, PWM, WDT
Number of I/O	35
Program Memory Size	14KB (8K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 31x12b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	40-UFQFN Exposed Pad
Supplier Device Package	40-UQFN (5x5)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf19175-i-mv

FIGURE 2: 28-PIN UQFN PIN DIAGRAM FOR PIC16(L)F19155/56



Note 1: See Table 3 for location of all peripheral functions.

2: All VDD and all VSS pins must be connected at the circuit board level. Allowing one or more VSS or VDD pins to float may result in degraded electrical performance or non-functionality.

3: The bottom pad of the QFN/UQFN package should be connected to VSS at the circuit board level.

TABLE 4-12: SPECIAL FUNCTION REGISTER SUMMARY BANKS 0-63 PIC16(L)F19155/56/75/76/85/86 (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on: MCLR
Bank 9											
CPU CORE REGISTERS; see Table 4-3 for specifics											
48Ch	SMT1TMRL	SMT1TMR								0000 0000	0000 0000
48Dh	SMT1TMRH	SMT1TMR								0000 0000	0000 0000
48Eh	SMT1TMRU	SMT1TMR								0000 0000	0000 0000
48Fh	SMT1CPRL	CPR								xxxx xxxx	xxxx xxxx
490h	SMT1CPRH	CPR								xxxx xxxx	xxxx xxxx
491h	SMT1CPRU	CPR								xxxx xxxx	xxxx xxxx
492h	SMT1CPWL	CPW								xxxx xxxx	xxxx xxxx
493h	SMT1CPWH	CPW								xxxx xxxx	xxxx xxxx
494h	SMT1CPWU	CPW								xxxx xxxx	xxxx xxxx
495h	SMT1PRL	SMT1PR								1111 1111	1111 1111
496h	SMT1PRH	SMT1PR								1111 1111	1111 1111
497h	SMT1PRU	SMT1PR								1111 1111	1111 1111
498h	SMT1CON0	EN	—	STP	WPOL	SPOL	CPOL	SMT1PS<1:0>		0-00 0000	0-00 0000
499h	SMT1CON1	SMT1GO	REPEAT	—	—	MODE<3:0>				00-- 0000	00-- 0000
49Ah	SMT1STAT	CPRUP	CPWUP	RST	—	—	TS	WS	AS	000- -000	000- -000
49Bh	SMT1CLK	—	—	—	—	—	CSEL<2:0>			0000 0000	0000 0000
49Ch	SMT1SIG	—	—	—	SSEL<4:0>					0000 0000	0000 0000
49Dh	SMT1WIN	—	—	—	WSEL<4:0>					0000 0000	0000 0000
49Eh	—	Unimplemented								---- ----	---- ----
49Fh	—	Unimplemented								---- ----	---- ----

Legend: x = unknown, u = unchanged, q = depends on condition, - = unimplemented, read as '0', r = reserved. Shaded locations unimplemented, read as '0'.

Note 1: Unimplemented data memory locations, read as '0'.

TABLE 4-12: SPECIAL FUNCTION REGISTER SUMMARY BANKS 0-63 PIC16(L)F19155/56/75/76/85/86 (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on: MCLR
Bank 60 (Continued)											
1E37h	CLC4GLS3	LC4G4D4T	LC4G4D4N	LC4G4D3T	LC4G4D3N	LC4G4D2T	LC4G4D2N	LC4G4D1T	LC4G4D1N	xxxx xxxx	uuuu uuuu
1E38h	RF0PPS	—	—	—	RF0PPS4	RF0PPS3	RF0PPS2	RF0PPS1	RF0PPS0	---x xxxx	--uu uuuu
1E39h	RF1PPS	—	—	—	RF1PPS4	RF1PPS3	RF1PPS2	RF1PPS1	RF1PPS0	--xx xxxx	--uu uuuu
1E3Ah	RF2PPS	—	—	—	RF2PPS4	RF2PPS3	RF2PPS2	RF2PPS1	RF2PPS0	--xx xxxx	--uu uuuu
1E3Bh	RF3PPS	—	—	—	RF3PPS4	RF3PPS3	RF3PPS2	RF3PPS1	RF3PPS0	--xx xxxx	--uu uuuu
1E3Ch	RF4PPS	—	—	—	RF4PPS4	RF4PPS3	RF4PPS2	RF4PPS1	RF4PPS0	--xx xxxx	--uu uuuu
1E3Dh	RF5PPS	—	—	—	RF5PPS4	RF5PPS3	RF5PPS2	RF5PPS1	RF5PPS0	--xx xxxx	--uu uuuu
1E3Eh	RF6PPS	—	—	—	RF6PPS4	RF6PPS3	RF6PPS2	RF6PPS1	RF6PPS0	--xx xxxx	--uu uuuu
1E3Fh	RF7PPS	—	—	—	RF7PPS4	RF7PPS3	RF7PPS2	RF7PPS1	RF7PPS0	--xx xxxx	--uu uuuu
1E40h	—	Unimplemented								---- ----	---- ----
1E41h	—	Unimplemented								---- ----	---- ----
1E42h	—	Unimplemented								---- ----	---- ----
1E43h	—	Unimplemented								---- ----	---- ----
1E44h	—	Unimplemented								---- ----	---- ----
1E45h	—	Unimplemented								---- ----	---- ----
1E46h	—	Unimplemented								---- ----	---- ----
1E47h	—	Unimplemented								---- ----	---- ----
1E48h	—	Unimplemented								---- ----	---- ----
1E49h	—	Unimplemented								---- ----	---- ----
1E4Ah	—	Unimplemented								---- ----	---- ----
1E4Bh	—	Unimplemented								---- ----	---- ----
1E4Ch	—	Unimplemented								---- ----	---- ----
1E4Dh	—	Unimplemented								---- ----	---- ----
1E4Eh	—	Unimplemented								---- ----	---- ----
1E4Fh	—	Unimplemented								---- ----	---- ----

Legend: x = unknown, u = unchanged, q = depends on condition, - = unimplemented, read as '0', r = reserved. Shaded locations unimplemented, read as '0'.

Note 1: Unimplemented data memory locations, read as '0'.

5.2 Register Definitions: Configuration Words

REGISTER 5-1: CONFIGURATION WORD 1: OSCILLATORS

R/P-1	U-1	R/P-1	R/P-1	R/P-1	R/P-1
FCMEN	—	CSWEN	LCDPEN	$\overline{\text{VBATEN}}$	$\overline{\text{CLKOUTEN}}$
bit 13					bit 8

U-1	R/P-1	R/P-1	R/P-1	U-1	R/P-1	R/P-1	R/P-1
—	RSTOSC2	RSTOSC1	RSTOSC0	—	FEXTOSC2	FEXTOSC1	FEXTOSC0
bit 7							bit 0

Legend:			
R = Readable bit	P = Programmable bit	x = Bit is unknown	U = Unimplemented bit, read as ‘1’
‘0’ = Bit is cleared	‘1’ = Bit is set	W = Writable bit	n = Value when blank or after Bulk Erase

bit 13 **FCMEN**: Fail-Safe Clock Monitor Enable bit
1 = FSCM timer enabled
0 = FSCM timer disabled

bit 12 **Unimplemented**: Read as ‘1’

bit 11 **CSWEN**: Clock Switch Enable bit
1 = Writing to NOSC and NDIV is allowed
0 = The NOSC and NDIV bits cannot be changed by user software

bit 10 **LCDPEN:** LCD Charge Pump Mode Enable bit
1 = User intends to enable LCD Charge Pump during LCD operation
0 = LCD Charge Pump forced off

bit 9 **$\overline{\text{VBATEN}}$:** VBAT Pin Enable bit
1 = VBAT functionality is disabled; VBAT pin becomes GPIO
0 = VBAT functionality is enabled; VBAT pin has a battery connected to it

bit 8 **$\overline{\text{CLKOUTEN}}$**: Clock Out Enable bit
If FEXTOSC = EC (high, mid or low) or Not Enabled:
1 = CLKOUT function is disabled; I/O or oscillator function on OSC2
0 = CLKOUT function is enabled; FOSC/4 clock appears at OSC2
Otherwise:
This bit is ignored.

bit 7 **Unimplemented**: Read as ‘1’

bit 6-4 **RSTOSC<2:0>**: Power-up Default Value for COSC bits
This value is the Reset-default value for COSC and selects the oscillator first used by user software.
111 = EXTOSC operating per FEXTOSC bits
110 = HFINTOSC with HFFRQ = 4’b0000
101 = LFINTOSC
100 = SOSC
011 = Reserved
010 = EXTOSC with 4x PLL, with EXTOSC operating per FEXTOSC bits
001 = HFINTOSC with 2x PLL = 32 MHz, with HFFRQ = ‘101’ and CDIV = ‘0000’
000 = HFINTOSC with OSCFRQ = 32 MHz and CDIV = 1:1

bit 3 **Unimplemented**: Read as ‘1’

bit 2-0 **FEXTOSC<2:0>**:FEXTOSC External Oscillator Mode Selection bits
111 = EC (External Clock) above 8 MHz
110 = EC (External Clock) for 100 kHz to 8 MHz
101 = EC (External Clock) below 100 kHz
100 = Oscillator not enabled
011 = Oscillator not enabled
010 = Oscillator not enabled
001 = Oscillator not enabled
000 = Oscillator not enabled

REGISTER 10-5: PIE3: PERIPHERAL INTERRUPT ENABLE REGISTER 3

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	U-0	R/W-0/0	R/W-0/0
RC2IE	TX2IE	RC1IE	TX1IE	—	—	BCL1IE	SSP1IE
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 **RC2IE:** USART Receive Interrupt Enable bit
1 = Enables the USART receive interrupt
0 = Enables the USART receive interrupt

bit 6 **TX2IE:** USART Transmit Interrupt Enable bit
1 = Enables the USART transmit interrupt
0 = Disables the USART transmit interrupt

bit 5 **RC1IE:** USART Receive Interrupt Enable bit
1 = Enables the USART receive interrupt
0 = Enables the USART receive interrupt

bit 4 **TX1IE:** USART Transmit Interrupt Enable bit
1 = Enables the USART transmit interrupt
0 = Disables the USART transmit interrupt

bit 3-2 **Unimplemented:** Read as '0'

bit 1 **BCL1IE:** MSSP1 Bus Collision Interrupt Enable bit
1 = MSSP bus collision interrupt enabled
0 = MSSP bus collision interrupt disabled

bit 0 **SSP1IE:** Synchronous Serial Port (MSSP1) Interrupt Enable bit
1 = Enables the MSSP interrupt
0 = Disables the MSSP interrupt

Note: Bit PEIE of the INTCON register must be set to enable any peripheral interrupt controlled by PIE1-PIE8.

TABLE 15-3: PPS OUTPUT SIGNAL ROUTING OPTIONS

Output Signal Name	Output Register Value	Remappable to Pins of PORTx													
		PIC16(L)F19155/56			PIC16(L)F19175/76					PIC16(L)F19185/86					
		PORTA	PORTB	PORTC	PORTA	PORTB	PORTC	PORTD	PORTE	PORTA	PORTB	PORTC	PORTD	PORTE	PORTF
RTCC	0x18	•		•	•		•			•		•			
ADGRDB	0x17	•		•	•		•			•					•
ADGRDA	0x16	•		•	•		•			•					•
TMR0	0x15		•	•		•	•					•			•
SDO1/SDA1	0x14		•	•		•	•				•	•			
SCK1/SCL1	0x13		•	•		•	•				•	•			
C2OUT	0x12	•		•	•				•	•				•	
C1OUT	0x11	•		•	•			•		•			•		
DT2	0x10		•	•		•		•			•		•		
TX2/CK2	0x0F		•	•		•		•			•		•		
DT1	0x0E		•	•		•	•					•			•
TX1/CK1	0x0D		•	•		•	•					•			•
PWM4OUT	0x0C		•	•		•		•			•		•		
PWM3OUT	0x0B		•	•		•		•			•		•		
CCP2	0x0A		•	•		•	•					•			•
CCP1	0x09		•	•		•	•					•			•
CWG1D	0x08		•	•		•		•			•		•		
CWG1C	0x07		•	•		•		•			•		•		
CWG1B	0x06		•	•		•		•			•		•		
CWG1A	0x05		•	•		•	•				•	•			
CLC4OUT	0x04		•	•		•		•			•		•		
CLC3OUT	0x03		•	•		•		•			•		•		
CLC2OUT	0x02	•		•	•		•			•					•
CLC1OUT	0x01	•		•	•		•			•					•

REGISTER 16-5: PMD4: PMD CONTROL REGISTER 4

R/W-0/0	R/W-0/0	U-0	R/W-0/0	U-0	U-0	U-0	R/W-0/0
UART2MD	UART1MD	—	MSSP1MD	—	—	—	CWG1MD
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 **UART2MD:** Disable EUSART2 bit
1 = EUSART2 module disabled
0 = EUSART2 module enabled

bit 6 **UART1MD:** Disable EUSART1 bit
1 = EUSART1 module disabled
0 = EUSART1 module enabled

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

bit 4 **MSSP1MD:** Disable MSSP1 bit
1 = MSSP1 module disabled
0 = MSSP1 module enabled

bit 3-1 **Unimplemented:** Read as '0'

bit 0 **CWG1MD:** Disable CWG1 bit
1 = CWG1 module disabled
0 = CWG1 module enabled

17.6 Register Definitions: Interrupt-on-Change Control

REGISTER 17-1: IOCAP: INTERRUPT-ON-CHANGE PORTA POSITIVE EDGE 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
IOCAP7	IOCAP6	IOCAP5	IOCAP4	IOCAP3	IOCAP2	IOCAP1	IOCAP0
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 **IOCAP<7:0>:** Interrupt-on-Change PORTA Positive Edge Enable bits
1 = Interrupt-on-Change enabled on the pin for a positive-going edge. IOCAFx bit and IOCIF flag will be set upon detecting an edge.
0 = Interrupt-on-Change disabled for the associated pin.

REGISTER 17-2: IOCAN: INTERRUPT-ON-CHANGE PORTA NEGATIVE EDGE 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
IOCAN7	IOCAN6	IOCAN5	IOCAN4	IOCAN3	IOCAN2	IOCAN1	IOCAN0
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 **IOCAN<7:0>:** Interrupt-on-Change PORTA Negative Edge Enable bits
1 = Interrupt-on-Change enabled on the pin for a negative-going edge. IOCAFx bit and IOCIF flag will be set upon detecting an edge.
0 = Interrupt-on-Change disabled for the associated pin.

REGISTER 17-3: IOCAF: INTERRUPT-ON-CHANGE PORTA FLAG REGISTER

R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0
IOCAF7	IOCAF6	IOCAF5	IOCAF4	IOCAF3	IOCAF2	IOCAF1	IOCAF0
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 7-0 **IOCAF<7:0>:** Interrupt-on-Change PORTA Flag bits
1 = An enabled change was detected on the associated pin.
Set when IOCAPx = 1 and a rising edge was detected on RAx, or when IOCANx = 1 and a falling edge was detected on RAx.
0 = No change was detected, or the user cleared the detected change.

19.5.8 CONTINUOUS SAMPLING MODE

Setting the CONT bit in the ADCON0 register automatically retriggers a new conversion cycle after updating the ADACC register. The GO bit remains set and re-triggering occurs automatically.

If ADSOI = 1, a threshold interrupt condition will clear GO and the conversions will stop.

19.5.9 DOUBLE SAMPLE OR CVD CONVERSION MODE

Double sampling is enabled by setting the ADDSEN bit of the ADCON1 register. When this bit is set, two conversions are required before the module will calculate threshold error (each conversion must still be triggered separately). The first conversion will set the ADMATH bit of the ADSTAT register and update ADACC, but will not calculate ERR or trigger ADTIF. When the second conversion completes, the first value is transferred to PREV (depending on the setting of ADPSIS) and the value of the second conversion is placed into ADRES. Only upon the completion of the second conversion is ERR calculated and ADTIF triggered (depending on the value of ADCALC). The ADACC registers will contain the difference of the two samples taken.

26.7 Timer1 Interrupts

The Timer1 register pair (TMRxH:TMRxL) increments to FFFFh and rolls over to 0000h. When Timer1 rolls over, the Timer1 interrupt flag bit of the respective PIR register is set. To enable the interrupt-on-rollover, you must set these bits:

- ON bit of the T1CON register
- TMR1IE bit of the PIE4 register
- PEIE bit of the INTCON register
- GIE bit of the INTCON register

The interrupt is cleared by clearing the TMR1IF bit in the Interrupt Service Routine.

Note:	To avoid immediate interrupt vectoring, the TMR1H:TMR1L register pair should be preloaded with a value that is not imminently about to rollover, and the TMR1IF flag should be cleared prior to enabling the timer interrupts.

26.8 Timer1 Operation During Sleep

Timer1 can only operate during Sleep when setup in Asynchronous Counter mode. In this mode, an external crystal or clock source can be used to increment the counter. To set up the timer to wake the device:

- ON bit of the T1CON register must be set
- TMR1IE bit of the PIE4 register must be set
- PEIE bit of the INTCON register must be set
- $\overline{\text{SYNC}}$ bit of the T1CON register must be set

The device will wake-up on an overflow and execute the next instructions. If the GIE bit of the INTCON register is set, the device will call the Interrupt Service Routine.

26.9 CCP Capture/Compare Time Base

The CCP modules use the TMR1H:TMR1L register pair as the time base when operating in Capture or Compare mode.

In Capture mode, the value in the TMR1H:TMR1L register pair is copied into the CCPRxH:CCPRxL register pair on a configured event.

In Compare mode, an event is triggered when the value CCPRxH:CCPRxL register pair matches the value in the TMR1H:TMR1L register pair. This event can be an Auto-conversion Trigger.

For more information, see **Section 29.0 "Capture/Compare/PWM Modules"**.

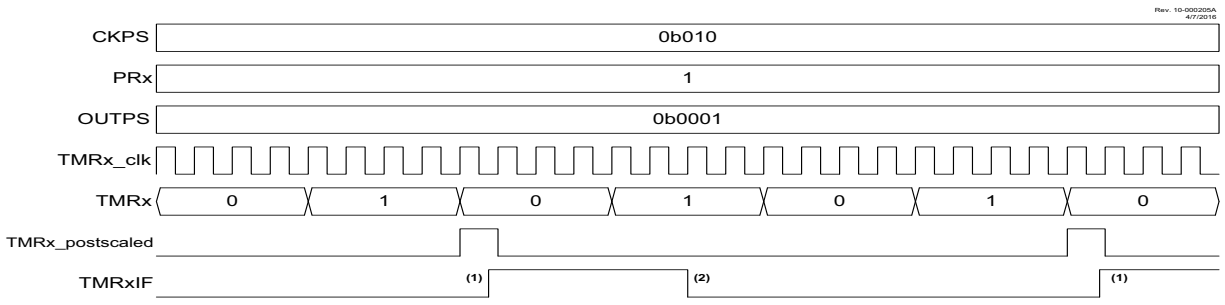
26.10 CCP Special Event Trigger

When any of the CCPs are configured to trigger a special event, the trigger will clear the TMRxH:TMRxL register pair. This special event does not cause a Timer1 interrupt. The CCP module may still be configured to generate a CCP interrupt. In this mode of operation, the CCPRxH:CCPRxL register pair becomes the period register for Timer1. Timer1 should be synchronized and FOSC/4 should be selected as the clock source in order to utilize the Special Event Trigger. Asynchronous operation of Timer1 can cause a Special Event Trigger to be missed. In the event that a write to TMRxH or TMRxL coincides with a Special Event Trigger from the CCP, the write will take precedence.

27.4 Timer2/4 Interrupt

Timer2/4 can also generate a device interrupt. The interrupt is generated when the postscaler counter matches one of 16 postscale options (from 1:1 through 1:16), which are selected with the postscaler control bits, OUTPS<3:0> of the T2CON register. The interrupt is enabled by setting the TMR2IE interrupt enable bit of the PIE4 register. Interrupt timing is illustrated in Figure 27-3.

FIGURE 27-3: TIMER2 PRESCALER, POSTSCALER, AND INTERRUPT TIMING DIAGRAM



Note 1: Setting the interrupt flag is synchronized with the instruction clock. Synchronization may take as many as 2 instruction cycles
2: Cleared by software.

27.5.8 LEVEL RESET, EDGE-TRIGGERED HARDWARE LIMIT ONE-SHOT MODES

In Level -Triggered One-Shot mode the timer count is reset on the external signal level and starts counting on the rising/falling edge of the transition from Reset level to the active level while the ON bit is set. Reset levels are selected as follows:

- Low Reset level (MODE<4:0> = `01110`)
- High Reset level (MODE<4:0> = `01111`)

When the timer count matches the PRx period count, the timer is reset and the ON bit is cleared. When the ON bit is cleared by either a PRx match or by software control a new external signal edge is required after the ON bit is set to start the counter.

When Level-Triggered Reset One-Shot mode is used in conjunction with the CCP PWM operation the PWM drive goes active with the external signal edge that starts the timer. The PWM drive goes inactive when the timer count equals the CCPRx pulse width count. The PWM drive does not go active when the timer count clears at the PRx period count match.

28.6.3 GO STATUS

Timer run status is determined by the TS bit of the SMTxSTAT register, and will be delayed in time by synchronizer delays in non-Counter modes.

28.7 Modes of Operation

The modes of operation are summarized in Table 28-2. The following sections provide detailed descriptions, examples of how the modes can be used. Note that all waveforms assume WPOL/SPOL/CPOL = 0. When WPOL/SPOL/CPOL = 1, all SMTSIGx, SMTWINx and SMT clock signals will have a polarity opposite to that indicated. For all modes, the REPEAT bit controls whether the acquisition is repeated or single. When REPEAT = 0 (Single Acquisition mode), the timer will stop incrementing and the SMTxGO bit will be reset upon the completion of an acquisition. Otherwise, the timer will continue and allow for continued acquisitions to overwrite the previous ones until the timer is stopped in software.

28.7.1 TIMER MODE

Timer mode is the simplest mode of operation where the SMTxTMR is used as a 16/24-bit timer. No data acquisition takes place in this mode. The timer increments as long as the SMTxGO bit has been set by software. No SMT window or SMT signal events affect the SMTxGO bit. Everything is synchronized to the SMT clock source. When the timer experiences a period match (SMTxTMR = SMTxPR), SMTxTMR is reset and the period match interrupt trips. See Figure 28-3.

TABLE 28-2: MODES OF OPERATION

MODE	Mode of Operation	Synchronous Operation	Reference
0000	Timer	Yes	**Section 28.7.1 “Timer Mode”**
0001	Gated Timer	Yes	**Section 28.7.2 “Gated Timer Mode”**
0010	Period and Duty Cycle Acquisition	Yes	**Section 28.7.3 “Period and Duty-Cycle Mode”**
0011	High and Low Time Measurement	Yes	**Section 28.7.4 “High and Low-Measure Mode”**
0100	Windowed Measurement	Yes	**Section 28.7.5 “Windowed Measure Mode”**
0101	Gated Windowed Measurement	Yes	**Section 28.7.6 “Gated Window Measure Mode”**
0110	Time of Flight	Yes	**Section 28.7.7 “Time of Flight Measure Mode”**
0111	Capture	Yes	**Section 28.7.8 “Capture Mode”**
1000	Counter	No	**Section 28.7.9 “Counter Mode”**
1001	Gated Counter	No	**Section 28.7.10 “Gated Counter Mode”**
1010	Windowed Counter	No	**Section 28.7.11 “Windowed Counter Mode”**
1011-1111	Reserved	—	—

31.10 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 31-12.

31.10.1 SHUTDOWN

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

- Software generated
- External input

31.10.1.1 Software Generated Shutdown

Setting the SHUTDOWN bit of the CWG1AS0 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.

31.10.2 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 selected override levels without software delay. Several input sources can be selected to cause a shutdown condition. All input sources are active-low. The sources are:

- Comparator C1OUT_sync
- Comparator C2OUT_sync
- Timer2 – TMR2_postscaled
- CWG1IN input pin

Shutdown inputs are selected using the CWG1AS1 register (Register 31-6).

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.

31.11 Operation During Sleep

The CWG module operates independently from the system clock and will continue to run during Sleep, provided that the clock and input sources selected remain active.

The HFINTOSC remains active during Sleep when all the following conditions are met:

- CWG module is enabled
- Input source is active
- HFINTOSC is selected as the clock source, regardless of the system clock source selected.

In other words, if the HFINTOSC is simultaneously selected as the system clock and the CWG clock source, when the CWG is enabled and the input source is active, then the CPU will go idle during Sleep, but the HFINTOSC will remain active and the CWG will continue to operate. This will have a direct effect on the Sleep mode current.

32.2 CLCx Interrupts

An interrupt will be generated upon a change in the output value of the CLCx when the appropriate interrupt enables are set. A rising edge detector and a falling edge detector are present in each CLC for this purpose.

The CLCxIF bit of the associated PIR5 register will be set when either edge detector is triggered and its associated enable bit is set. The LCxINTP enables rising edge interrupts and the LCxINTN bit enables falling edge interrupts. Both are located in the CLCxCON register.

To fully enable the interrupt, set the following bits:

- CLCxIE bit of the PIE5 register
- LCxINTP bit of the CLCxCON register (for a rising edge detection)
- LCxINTN bit of the CLCxCON register (for a falling edge detection)
- PEIE and GIE bits of the INTCON register

The CLCxIF bit of the PIR5 register, must be cleared in software as part of the interrupt service. If another edge is detected while this flag is being cleared, the flag will still be set at the end of the sequence.

32.3 Output Mirror Copies

Mirror copies of all LCxCON output bits are contained in the CLCxDATA register. Reading this register reads the outputs of all CLCs simultaneously. This prevents any reading skew introduced by testing or reading the LCxOUT bits in the individual CLCxCON registers.

32.4 Effects of a Reset

The CLCxCON register is cleared to zero as the result of a Reset. All other selection and gating values remain unchanged.

32.5 Operation During Sleep

The CLC module operates independently from the system clock and will continue to run during Sleep, provided that the input sources selected remain active.

The HFINTOSC remains active during Sleep when the CLC module is enabled and the HFINTOSC is selected as an input source, regardless of the system clock source selected.

In other words, if the HFINTOSC is simultaneously selected as the system clock and as a CLC input source, when the CLC is enabled, the CPU will go idle during Sleep, but the CLC will continue to operate and the HFINTOSC will remain active.

This will have a direct effect on the Sleep mode current.

32.6 CLCx Setup Steps

The following steps should be followed when setting up the CLCx:

- Disable CLCx by clearing the LCxEN bit.
- Select desired inputs using CLCxSEL0 through CLCxSEL3 registers (See Table 32-2).
- Clear any associated ANSEL bits.
- Enable the chosen inputs through the four gates using CLCxGLS0, CLCxGLS1, CLCxGLS2, and CLCxGLS3 registers.
- Select the gate output polarities with the LCxGyPOL bits of the CLCxPOL register.
- Select the desired logic function with the LCxMODE<2:0> bits of the CLCxCON register.
- Select the desired polarity of the logic output with the LCxPOL bit of the CLCxPOL register. (This step may be combined with the previous gate output polarity step).
- If driving a device pin, set the desired pin PPS control register and also clear the TRIS bit corresponding to that output.
- If interrupts are desired, configure the following bits:
  - Set the LCxINTP bit in the CLCxCON register for rising event.
  - Set the LCxINTN bit in the CLCxCON register for falling event.
  - Set the CLCxIE bit of the PIE5 register.
  - Set the GIE and PEIE bits of the INTCON register.
- Enable the CLCx by setting the LCxEN bit of the CLCxCON register.

33.0 MASTER SYNCHRONOUS SERIAL PORT (MSSP) MODULES

33.1 MSSP Module Overview

The Master Synchronous Serial Port (MSSP) module is a serial interface useful for communicating with other peripheral or microcontroller devices. These peripheral devices may be serial EEPROMs, shift registers, display drivers, A/D converters, etc. The MSSP module can operate in one of two modes:

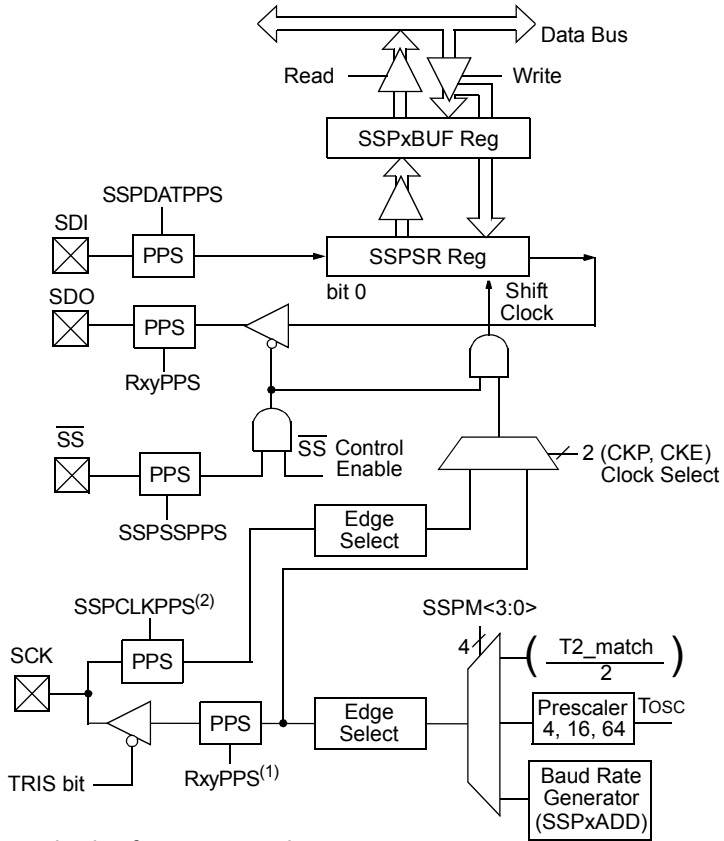
- Serial Peripheral Interface (SPI)
- Inter-Integrated Circuit (I^2C)

The SPI interface supports the following modes and features:

- Master mode
- Slave mode
- Clock Polarity
- Slave Select Synchronization (Slave mode only)
- Daisy-chain connection of slave devices

Figure 33-1 is a block diagram of the SPI interface module.

FIGURE 33-1: MSSP BLOCK DIAGRAM (SPI MODE)



Note 1: Output selection for master mode

2: Input selection for slave mode

33.6.10 SLEEP OPERATION

While in Sleep mode, the I^2C slave module can receive addresses or data and when an address match or complete byte transfer occurs, wake the processor from Sleep (if the MSSP interrupt is enabled).

33.6.11 EFFECTS OF A RESET

A Reset disables the MSSP module and terminates the current transfer.

33.6.12 MULTI-MASTER MODE

In Multi-Master mode, the interrupt generation on the detection of the Start and Stop conditions allows the determination of when the bus is free. The Stop (P) and Start (S) bits are cleared from a Reset or when the MSSP module is disabled. Control of the I^2C bus may be taken when the P bit of the SSPxSTAT register is set, or the bus is Idle, with both the S and P bits clear. When the bus is busy, enabling the SSP interrupt will generate the interrupt when the Stop condition occurs.

In multi-master operation, the SDA line must be monitored for arbitration to see if the signal level is the expected output level. This check is performed by hardware with the result placed in the BCL1IF bit.

The states where arbitration can be lost are:

- Address Transfer
- Data Transfer
- A Start Condition
- A Repeated Start Condition
- An Acknowledge Condition

33.6.13 MULTI -MASTER COMMUNICATION, BUS COLLISION AND BUS ARBITRATION

Multi-Master mode support is achieved by bus arbitration. When the master outputs address/data bits onto the SDA pin, arbitration takes place when the master outputs a '1' on SDA, by letting SDA float high and another master asserts a '0'. When the SCL pin floats high, data should be stable. If the expected data on SDA is a '1' and the data sampled on the SDA pin is '0', then a bus collision has taken place. The master will set the Bus Collision Interrupt Flag, BCL1IF and reset the I^2C port to its Idle state (Figure 33-32).

If a transmit was in progress when the bus collision occurred, the transmission is halted, the BF flag is cleared, the SDA and SCL lines are deasserted and the SSPxBUF can be written to. When the user services the bus collision Interrupt Service Routine and if the I^2C bus is free, the user can resume communication by asserting a Start condition.

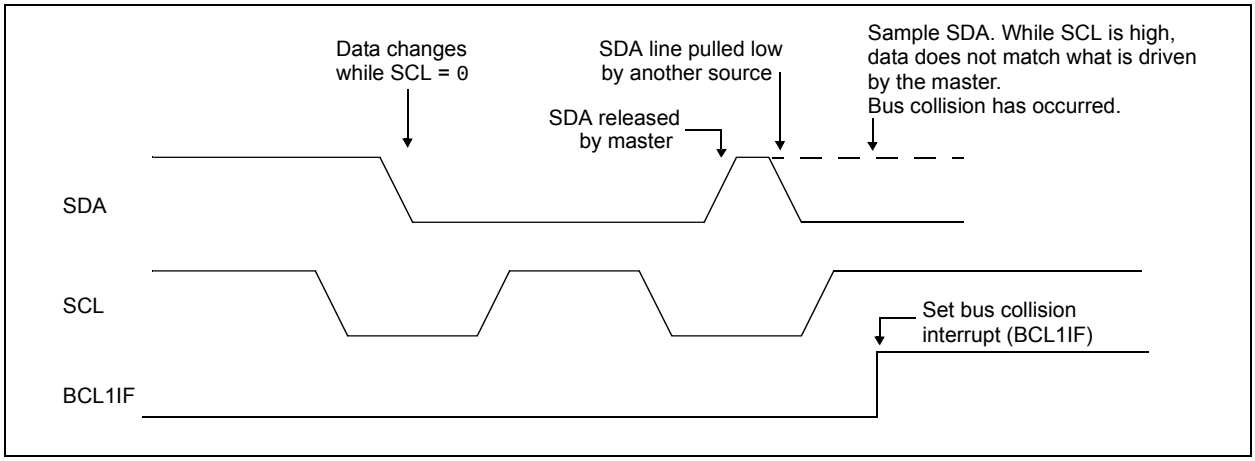
If a Start, Repeated Start, Stop or Acknowledge condition was in progress when the bus collision occurred, the condition is aborted, the SDA and SCL lines are deasserted and the respective control bits in the SSPxCON2 register are cleared. When the user services the bus collision Interrupt Service Routine and if the I^2C bus is free, the user can resume communication by asserting a Start condition.

The master will continue to monitor the SDA and SCL pins. If a Stop condition occurs, the SSPxIF bit will be set.

A write to the SSPxBUF will start the transmission of data at the first data bit, regardless of where the transmitter left off when the bus collision occurred.

In Multi-Master mode, the interrupt generation on the detection of Start and Stop conditions allows the determination of when the bus is free. Control of the I^2C bus can be taken when the P bit is set in the SSPxSTAT register, or the bus is Idle and the S and P bits are cleared.

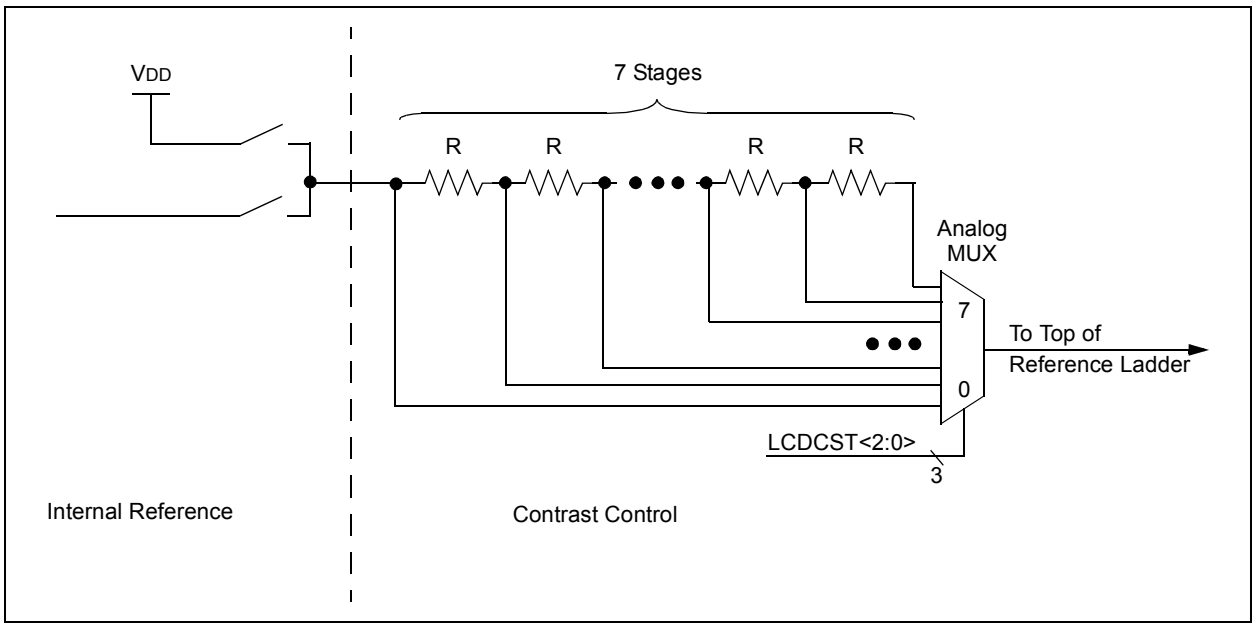
FIGURE 33-32: BUS COLLISION TIMING FOR TRANSMIT AND ACKNOWLEDGE



35.5.3 CONTRAST CONTROL

The LCD contrast control circuit consists of a 7-tap resistor ladder, controlled by the LCDCSTx bits (see Figure 35-4).

FIGURE 35-4: INTERNAL REFERENCE AND CONTRAST CONTROL BLOCK DIAGRAM



35.6 Bias Generation

35.6.1 INTERNAL REFERENCE

An internal reference for the LCD bias voltage can be enabled under firmware control. When enabled, the source of this voltage can be VDD, LCD charge pump or 3x FVR.

When no internal reference is selected, the LCD contrast control circuit is disabled and LCD bias must be provided externally. Whenever the LCD module is inactive (LCDA = 0), the internal reference will be turned off.

35.6.2 VLCDx PINS AND EXTERNAL BIAS

The VLCD3, VLCD2 and VLCD1 pins provide the ability for an external LCD bias network to be used instead of the internal ladder. Use of the VLCDx pins does not prevent use of the internal ladder.

35.6.3 LCD BIAS GENERATION

The LCD driver module is capable of generating the required bias voltages for LCD operation with a minimum of external components. This includes the ability to generate the different voltage levels required by the different bias types that are required by the LCD. The driver module can also provide bias voltages, both above and below microcontroller VDD, through the use of an on-chip LCD charge pump.

35.6.4 LCD CHARGE PUMP

The purpose of the LCD charge pump is to provide proper bias voltage and good contrast for the LCD, regardless of VDD levels. This module contains a charge pump and internal voltage reference. The charge pump can be configured by using external components to boost bias voltage above VDD. It can also operate a display at a constant voltage below VDD. The charge pump can also be selectively disabled to allow bias voltages to be generated by an external resistor network.

The LCD charge pump is controlled through the LCDVCONx registers.

35.6.5 VLCD3 MONITORING

The ADC can be used to measure the VLCD3 voltage via a VLCD3 divided by 4 channel on the ADC. This feature is useful when active adjustment of the LCDCST<2:0> or BIAS<2:0> bits need to be made to account of contrast changes due to extreme temperatures and/or a high number of large active pixels. See **Section 19.0 "Analog-to-Digital Converter with Computation (ADC2) Module"** for additional details.

35.8 LCD Multiplex Types

The LCD driver module can be configured into four multiplex types:

- Static (only COM0 used)
- 1/2 multiplex (COM0 and COM1 are used)
- 1/3 multiplex (COM0, COM1 and COM2 are used)
- 1/4 multiplex (COM0, COM1, COM2 and COM3 are used)
- 1/5 multiplex (COM0, COM1, COM2, COM3 and COM4 are used)
- 1/6 multiplex (COM0, COM1, COM2, COM3, COM4 and COM5 are used)
- 1/7 multiplex (COM0, COM1, COM2, COM3, COM4, COM5 and COM6 are used)
- 1/8 multiplex (COM0, COM1, COM2, COM3, COM4, COM5, COM6 and COM7 are used)

The LMUX<3:0> setting (LCDCON<3:0>) decides the function of the COM pins. (For details, see Table 35-8).

If the pin is a digital I/O, the corresponding TRIS bit controls the data direction. If the pin is a COM drive, the TRIS setting of that pin is overridden.

Note: On a Power-on Reset, the LMUX<3:0> bits are '0000'.

TABLE 35-8: COM<7:0> PIN FUNCTIONS

LMUX<3:0>	COM7 Pin	COM6 Pin	COM5 Pin	COM4 Pin	COM3 Pin	COM2 Pin	COM1 Pin	COM0 Pin
1000	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
0111	I/O Pin	COM6	COM5	COM4	COM3	COM2	COM1	COM0
0110	I/O Pin	I/O Pin	COM5	COM4	COM3	COM2	COM1	COM0
0101	I/O Pin	I/O Pin	I/O Pin	COM4	COM3	COM2	COM1	COM0
0100	I/O Pin	I/O Pin	I/O Pin	I/O Pin	COM3	COM2	COM1	COM0
0011	I/O Pin	I/O Pin	I/O Pin	I/O Pin	I/O Pin	COM2	COM1	COM0
0010	I/O Pin	I/O Pin	I/O Pin	I/O Pin	I/O Pin	I/O Pin	COM1	COM0
0001	I/O Pin	I/O Pin	I/O Pin	I/O Pin	I/O Pin	I/O Pin	I/O Pin	COM0
0000	I/O Pin	I/O Pin	I/O Pin	I/O Pin	I/O Pin	I/O Pin	I/O Pin	I/O Pin

LSLF **Logical Left Shift**

Syntax: [*label*] LSLF f {,d}

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: (f<7>) → C
(f<6:0>) → dest<7:1>
0 → dest<0>

Status Affected: C, Z

Description: The contents of register 'f' are shifted one bit to the left through the Carry flag. A '0' is shifted into the LSb. If 'd' is '0', the result is placed in W. If 'd' is '1', the result is stored back in register 'f'.

LSRF **Logical Right Shift**

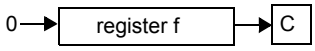
Syntax: [*label*] LSRF f {,d}

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: 0 → dest<7>
(f<7:1>) → dest<6:0>,
(f<0>) → C,

Status Affected: C, Z

Description: The contents of register 'f' are shifted one bit to the right through the Carry flag. A '0' is shifted into the MSb. If 'd' is '0', the result is placed in W. If 'd' is '1', the result is stored back in register 'f'.



MOVF **Move f**

Syntax: [*label*] MOVF f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: (f) → (dest)

Status Affected: Z

Description: The contents of register f is moved to a destination dependent upon the status of d. If d = 0, destination is W register. If d = 1, the destination is file register f itself. d = 1 is useful to test a file register since status flag Z is affected.

Words: 1

Cycles: 1

Example:

```
MOVF    FSR, 0
```

After Instruction
W = value in FSR register
Z = 1