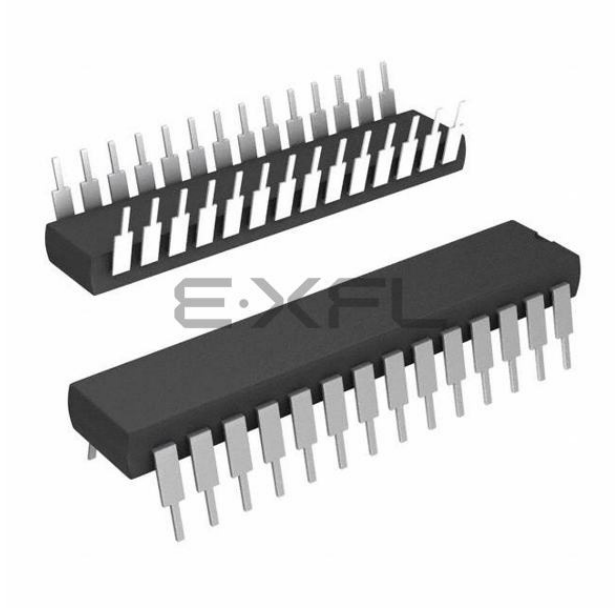


Welcome to E-XFL.COM

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	24
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)	2.3V ~ 5.5V
Data Converters	A/D 20x12b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Through Hole
Package / Case	28-DIP (0.300", 7.62mm)
Supplier Device Package	28-SPDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f19155-i-sp

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 19											
CPU CORE REGISTERS; see Table 4-3 for specifics											
98Ch	—	Unimplemented								---- ----	---- ----
98Dh	—	Unimplemented								---- ----	---- ----
98Eh	—	Unimplemented								---- ----	---- ----
98Fh	CMOUT	—	—	—	—	—	—	MC2OUT	MC1OUT	0000 0000	0000 0000
990h	CM1CON0	ON	OUT	—	POL	—	—	HYS	SYNC	00-0 --00	00-0 --00
991h	CM1CON1	—	—	—	—	—	—	INTP	INTN	0000 0000	0000 0000
992h	CM1NSEL	—	—	—	—	—	NCH<2:0>			0000 0000	0000 0000
992h		—	—	—	—	—	NCH2	NCH1	NCH0	0000 0000	0000 0000
993h	CM1PSEL	—	—	—	—	PCH<3:0>				0000 0000	0000 0000
993h		—	—	—	—	—	PCH2	PCH1	PCH0	0000 0000	0000 0000
994h	CM2CON0	ON	OUT	—	POL	—	—	HYS	SYNC	00-0 --00	00-0 --00
995h	CM2CON1	—	—	—	—	—	—	INTP	INTN	0000 0000	0000 0000
996h	CM2NSEL	—	—	—	—	—	NCH<2:0>			0000 0000	0000 0000
996h		—	—	—	—	—	NCH2	NCH1	NCH0	0000 0000	0000 0000
997h	CM2PSEL	—	—	—	—	—	PCH<2:0>			0000 0000	0000 0000
997h		—	—	—	—	—	PCH2	PCH1	PCH0	0000 0000	0000 0000
998h	—	Unimplemented								---- ----	---- ----
999h	—	Unimplemented								---- ----	---- ----
99Ah	—	Unimplemented								---- ----	---- ----
99Bh	—	Unimplemented								---- ----	---- ----
99Ch	—	Unimplemented								---- ----	---- ----
99Dh	—	Unimplemented								---- ----	---- ----
99Eh	—	Unimplemented								---- ----	---- ----
99Fh	—	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
Banks 30-57											
CPU CORE REGISTERS; see Table 4-3 for specifics											
F0Ch — 1C9Fh	—	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'.

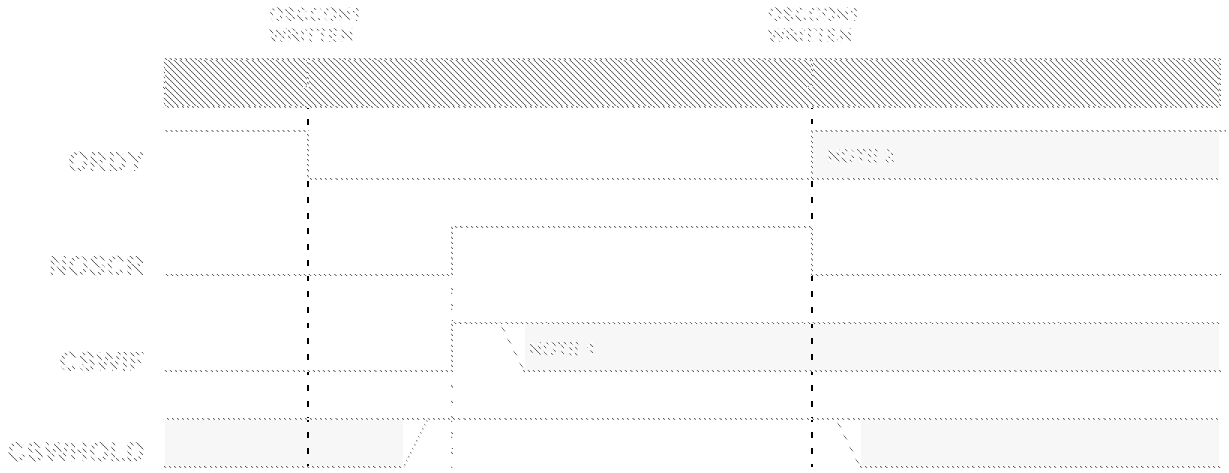
TABLE 4-12: SPECIAL FUNCTION REGISTER SUMMARY BANKS 0-63 PIC16(L)F19155/56/75/76/85/86 (CONTINUED)

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 61 (Continued)											
1EE5h	—	Unimplemented								---- ----	---- ----
1EE6h	—	Unimplemented								---- ----	---- ----
1EE7h	—	Unimplemented								---- ----	---- ----
1EE8h	—	Unimplemented								---- ----	---- ----
1EE9h	—	Unimplemented								---- ----	---- ----
1EEAh	—	Unimplemented								---- ----	---- ----
1EEBh	—	Unimplemented								---- ----	---- ----
1EECh	—	Unimplemented								---- ----	---- ----
1EEDh	—	Unimplemented								---- ----	---- ----
1EEEh	—	Unimplemented								---- ----	---- ----
1EEFh	—	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'.

FIGURE 9-6: CLOCK SWITCH ABANDONED



Note 1: CSWIF may be cleared before or after rewriting OSCCON1; CSWIF is not automatically cleared.
2: ORDY = 0 if OSCCON1 does not match OSCCON2; a new switch will begin.

REGISTER 10-11: PIR0: PERIPHERAL INTERRUPT STATUS REGISTER 0

U-0	U-0	R/W/HS-0/0	R-0	U-0	U-0	U-0	R/W/HS-0/0
—	—	TMR0IF	IOCIF	—	—	—	INTF(1)
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= Hardware Set

bit 7-6 **Unimplemented:** Read as '0'

bit 5 **TMR0IF:** Timer0 Overflow Interrupt Flag bit
- 1 = Timer0 register has overflowed (must be cleared in software)
- 0 = Timer0 register did not overflow

bit 4 **IOCIF:** Interrupt-on-Change Interrupt Flag bit (read-only)(2)
- 1 = One or more of the IOCAF-IOCEF register bits are currently set, indicating an enabled edge was detected by the IOC module.
- 0 = None of the IOCAF-IOCEF register bits are currently set

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

bit 0 **INTF:** INT External Interrupt Flag bit(1)
- 1 = The INT external interrupt occurred (must be cleared in software)
- 0 = The INT external interrupt did not occur

Note 1: The External Interrupt GPIO pin is selected by INTPPS (Register 15-1).

2: The IOCIF bit is the logical OR of all the IOCAF-IOCEF flags. Therefore, to clear the IOCIF flag, application firmware must clear all of the lower level IOCAF-IOCEF register bits.

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

To minimize current consumption, the following conditions should be considered:

- I/O pins should not be floating
- External circuitry sinking current from I/O pins
- Internal circuitry sourcing current from I/O pins
- Current draw from pins with internal weak pull-ups
- Modules using any oscillator

I/O pins that are high-impedance inputs should be pulled to VDD or VSS externally to avoid switching currents caused by floating inputs.

Any module with a clock source that is not FOSC can be enabled. Examples of internal circuitry that might be sourcing current include modules such as the DAC and FVR modules. See **Section 21.0 "5-Bit Digital-to-Analog Converter (DAC1) Module"**, **Section 18.0 "Fixed Voltage Reference (FVR)"** for more information on these modules.

11.2.1 WAKE-UP FROM SLEEP

The device can wake-up from Sleep through one of the following events:

1. External Reset input on $\overline{\text{MCLR}}$ pin, if enabled.
2. BOR Reset, if enabled.
3. POR Reset.
4. Watchdog Timer, if enabled.
5. Any external interrupt.
6. Interrupts by peripherals capable of running during Sleep (see individual peripheral for more information).

The first three events will cause a device Reset. The last three events are considered a continuation of program execution. To determine whether a device Reset or wake-up event occurred, refer to **Section 8.14 "Determining the Cause of a Reset"**.

When the SLEEP instruction is being executed, the next instruction (PC + 1) is prefetched. For the device to wake-up through an interrupt event, the corresponding interrupt enable bit must be enabled. Wake-up will occur regardless of the state of the GIE bit. If the GIE bit is disabled, the device continues execution at the instruction after the SLEEP instruction. If the GIE bit is enabled, the device executes the instruction after the SLEEP instruction, the device will then call the Interrupt Service Routine. In cases where the execution of the instruction following SLEEP is not desirable, the user should have a NOP after the SLEEP instruction.

The WWDT is cleared when the device wakes-up from Sleep, regardless of the source of wake-up.

EXAMPLE 13-1: PFM PROGRAM MEMORY READ

```
* This code block will read 1 word of program
* memory at the memory address:
    PROG_ADDR_HI : PROG_ADDR_LO
*   data will be returned in the variables;
*   PROG_DATA_HI, PROG_DATA_LO

    BANKSEL  NVMADRL          ; Select Bank for NVMCON registers
    MOVLW    PROG_ADDR_LO     ;
    MOVWF    NVMADRL          ; Store LSB of address
    MOVLW    PROG_ADDR_HI     ;
    MOVWF    NVMADRH          ; Store MSB of address

    BCF      NVMCON1,NVMREGS  ; Do not select Configuration Space
    BSF      NVMCON1,RD       ; Initiate read

    MOVF     NVMDATL,W        ; Get LSB of word
    MOVWF    PROG_DATA_LO     ; Store in user location
    MOVF     NVMDATH,W        ; Get MSB of word
    MOVWF    PROG_DATA_HI     ; Store in user location
```

EXAMPLE 13-3: ERASING ONE ROW OF PROGRAM FLASH MEMORY (PFM)

```
; This sample row erase routine assumes the following:
; 1.A valid address within the erase row is loaded in variables ADDRH:ADDRL
; 2.ADDRH and ADDRL are located in common RAM (locations 0x70 - 0x7F)

BANKSEL         NVMADRL
MOVF            ADDRL,W
MOVWF           NVMADRL                 ; Load lower 8 bits of erase address boundary
MOVF            ADDRH,W
MOVWF           NVMADRH                 ; Load upper 6 bits of erase address boundary
BCF             NVMCON1,NVMREGS         ; Choose PFM memory area
BSF             NVMCON1,FREE            ; Specify an erase operation
BSF             NVMCON1,WREN            ; Enable writes
BCF             INTCON,GIE              ; Disable interrupts during unlock sequence

; ------------------------------REQUIRED UNLOCK SEQUENCE:------------------------------

MOVLW           55h                     ; Load 55h to get ready for unlock sequence
MOVWF           NVMCON2                 ; First step is to load 55h into NVMCON2
MOVLW           AAh                     ; Second step is to load AAh into W
MOVWF           NVMCON2                 ; Third step is to load AAh into NVMCON2
BSF             NVMCON1,WR              ; Final step is to set WR bit

; -------------------------------------------------------------------------------------

BSF             INTCON,GIE              ; Re-enable interrupts, erase is complete
BCF             NVMCON1,WREN            ; Disable writes
```

EXAMPLE 13-4: WRITING TO PROGRAM FLASH MEMORY (PFM)

```
; This write routine assumes the following:
; 1. 32 bytes of data are loaded, starting at the address in DATA_ADDR
; 2. Each word of data to be written is made up of two adjacent bytes in DATA_ADDR,
;    stored in little endian format
; 3. A valid starting address (the least significant bits = 00000) is loaded in ADDRH:ADDRL
; 4. ADDRH and ADDRL are located in common RAM (locations 0x70 - 0x7F)
; 5. NVM interrupts are not taken into account

        BANKSEL     NVMADRH
        MOVF        ADDRH,W
        MOVWF       NVMADRH             ; Load initial address
        MOVF        ADDRL,W
        MOVWF       NVMADRL
        MOVLW       LOW DATA_ADDR       ; Load initial data address
        MOVWF       FSR0L
        MOVLW       HIGH DATA_ADDR
        MOVWF       FSR0H
        BCF         NVMCON1,NVMREGS     ; Set Program Flash Memory as write location
        BSF         NVMCON1,WREN        ; Enable writes
        BSF         NVMCON1,LWLO        ; Load only write latches

LOOP
        MOVIW       FSR0++
        MOVWF       NVMDATL             ; Load first data byte
        MOVIW       FSR0++
        MOVWF       NVMDATH             ; Load second data byte

        MOVF        NVMADRL,W
        XORLW       0x1F                ; Check if lower bits of address are 00000
        ANDLW       0x1F                ; and if on last of 32 addresses
        BTFSC       STATUS,Z            ; Last of 32 words?
        GOTO        START_WRITE         ; If so, go write latches into memory

        CALL        UNLOCK_SEQ          ; If not, go load latch
        INCF        NVMADRL,F           ; Increment address
        GOTO        LOOP

START_WRITE
        BCF         NVMCON1,LWLO        ; Latch writes complete, now write memory
        CALL        UNLOCK_SEQ          ; Perform required unlock sequence
        BCF         NVMCON1,WREN        ; Disable writes

UNLOCK_SEQ
        MOVLW       55h
        BCF         INTCON,GIE          ; Disable interrupts
        MOVWF       NVMCON2             ; Begin unlock sequence
        MOVLW       AAh
        MOVWF       NVMCON2
        BSF         NVMCON1,WR
        BSF         INTCON,GIE          ; Unlock sequence complete, re-enable interrupts
        return
```

REGISTER 17-9: IOCCF: INTERRUPT-ON-CHANGE PORTC FLAG REGISTER

R/W/HS-0/0	R/W/HS-0/0	U-0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0
IOCCF7	IOCCF6	—	IOCCF4	IOCCF3	IOCCF2	IOCCF1	IOCCF0
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 7-6 **IOCCF<7:6>:** Interrupt-on-Change PORTC Flag bits
1 = An enabled change was detected on the associated pin
Set when IOCCPx = 1 and a rising edge was detected on RCx, or when IOCCNx = 1 and a falling edge was detected on RCx.
0 = No change was detected, or the user cleared the detected change

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

bit 4-0 **IOCCF<4:0>:** Interrupt-on-Change PORTC Flag bits
1 = An enabled change was detected on the associated pin
Set when IOCCPx = 1 and a rising edge was detected on RCx, or when IOCCNx = 1 and a falling edge was detected on RCx.
0 = No change was detected, or the user cleared the detected change

REGISTER 17-10: IOCEP: INTERRUPT-ON-CHANGE PORTE POSITIVE EDGE REGISTER

U-0	U-0	U-0	U-0	R/W/HS-0/0	U-0	U-0	U-0
—	—	—	—	IOCEP3(1)	—	—	—
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

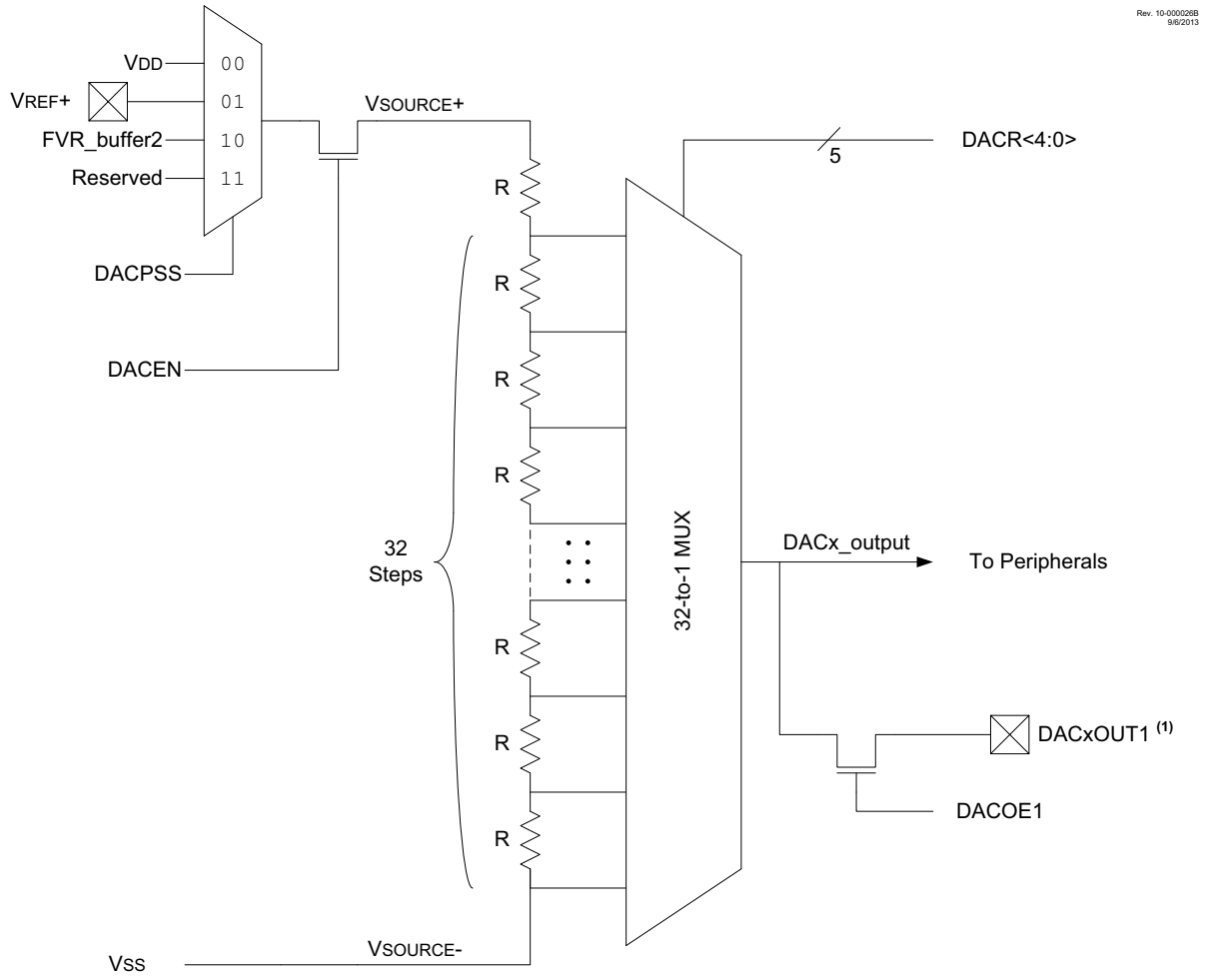
bit 7-4 **Unimplemented:** Read as '0'

bit 3 **IOCEP3:** Interrupt-on-Change PORTE Positive Edge Enable bit
1 = Interrupt-on-Change enabled on the pin for a positive-going edge. IOCEFx bit and IOCIF flag will be set upon detecting an edge.
0 = Interrupt-on-Change disabled for the associated pin

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

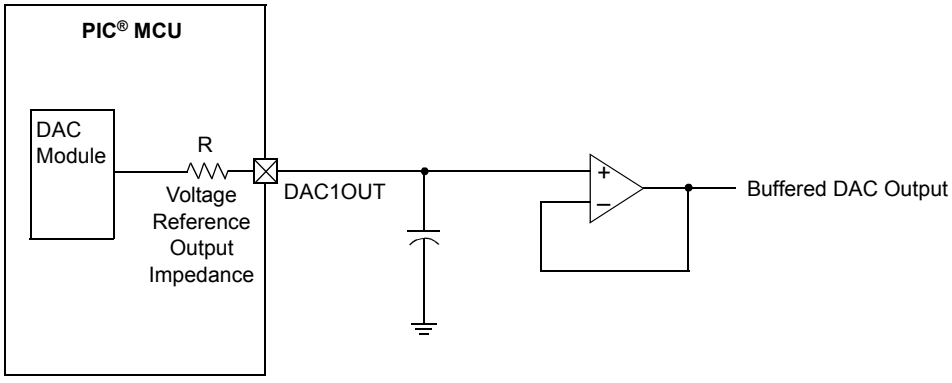
Note 1: If MCLRE = 1 or LVP = 1, RC port functionality is disabled and IOC is not available on RE3.

FIGURE 21-1: DIGITAL-TO-ANALOG CONVERTER BLOCK DIAGRAM



Note 1: The unbuffered DACx_output is provided on the DACxOUT pin(s).

FIGURE 21-2: VOLTAGE REFERENCE OUTPUT BUFFER EXAMPLE



21.4 Operation During Sleep

The DAC continues to function during Sleep. When the device wakes up from Sleep through an interrupt or a Watchdog Timer time-out, the contents of the DAC1CON0 register are not affected.

21.5 Effects of a Reset

A device Reset affects the following:

- DAC is disabled.
- DAC output voltage is removed from the DAC1OUT1/2 pins.
- The DAC1R<4:0> range select bits are cleared.

REGISTER 27-3: TxHLT: TIMER2/4 HARDWARE LIMIT CONTROL 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
PSYNC(1, 2)	CKPOL(3)	CKSYNC(4, 5)	MODE<4:0>(6, 7)				
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 **PSYNC:** Timer2/4 Prescaler Synchronization Enable bit(1, 2)
- 1 = TMRx Prescaler Output is synchronized to Fosc/4
- 0 = TMRx Prescaler Output is not synchronized to Fosc/4

bit 6 **CKPOL:** Timer2/4 Clock Polarity Selection bit(3)
- 1 = Falling edge of input clock clocks timer/prescaler
- 0 = Rising edge of input clock clocks timer/prescaler

bit 5 **CKSYNC:** Timer2/4 Clock Synchronization Enable bit(4, 5)
- 1 = ON register bit is synchronized to TMR2_clk input
- 0 = ON register bit is not synchronized to TMR2_clk input

bit 4-0 **MODE<4:0>:** Timer2/4 Control Mode Selection bits(6, 7)
See Table 27-1.

Note 1: Setting this bit ensures that reading TMRx will return a valid value.

2: When this bit is '1', Timer2/4 cannot operate in Sleep mode.

3: CKPOL should not be changed while ON = 1.

4: Setting this bit ensures glitch-free operation when the ON is enabled or disabled.

5: When this bit is set then the timer operation will be delayed by two TMRx input clocks after the ON bit is set.

6: Unless otherwise indicated, all modes start upon ON = 1 and stop upon ON = 0 (stops occur without affecting the value of TMRx).

7: When TMRx = PRx, the next clock clears TMRx, regardless of the operating mode.

28.7.7 TIME OF FLIGHT MEASURE MODE

Time of Flight Measure mode measures the time interval between a rising edge on the SMTWINx input and a rising edge on the SMTx_signal input, beginning to increment the timer upon observing a rising edge on the SMTWINx input, while updating the SMTxCPR register and resetting the timer upon observing a rising edge on the SMTx_signal input. In the event of two SMTWINx rising edges without an SMTx_signal rising edge, it will update the SMTxCPW register with the current value of the timer and reset the timer value. See Figure 28-14 and Figure 28-15.

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.

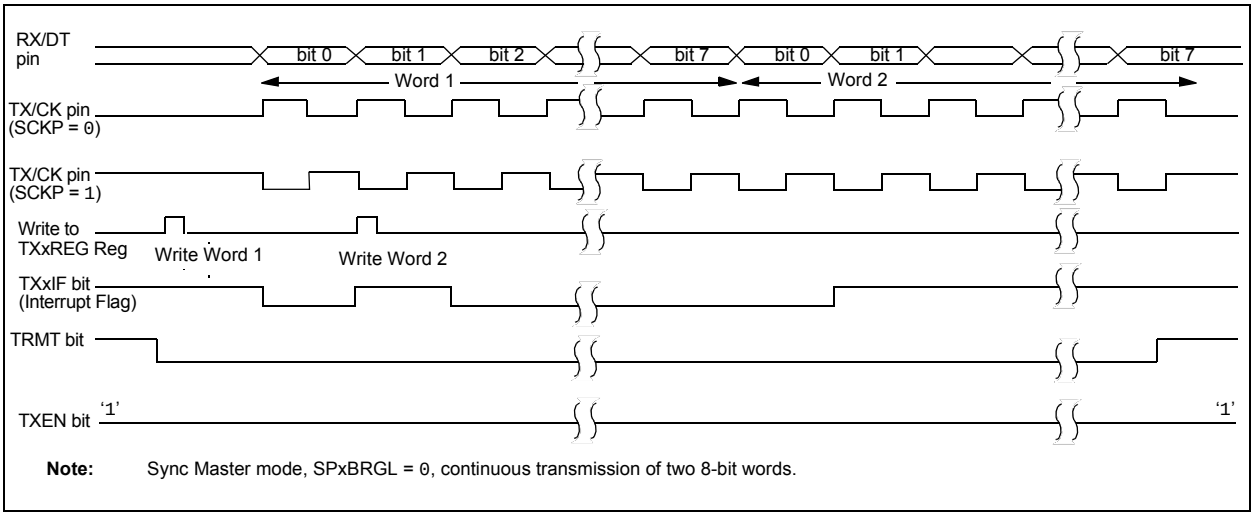
This will have a direct effect on the Sleep mode current.

32.6 CLCx Setup Steps

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

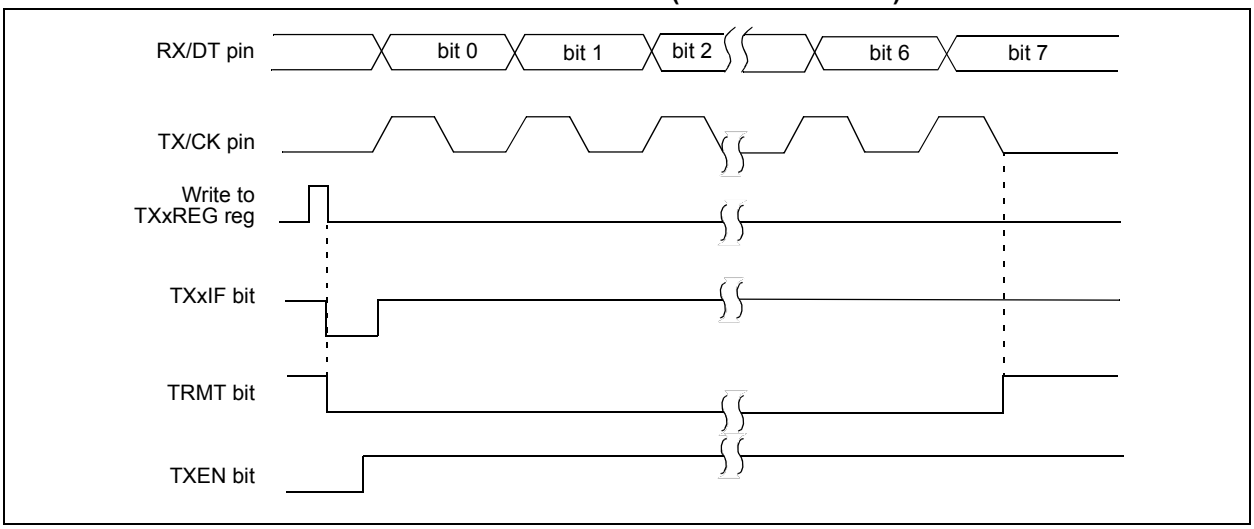
- Disable CLCx by clearing the LCxEN bit.
- Select desired inputs using CLCxSEL0 through CLCxSEL3 registers (See Table 32-2).
- Clear any associated ANSEL bits.
- Enable the chosen inputs through the four gates using CLCxGLS0, CLCxGLS1, CLCxGLS2, and CLCxGLS3 registers.
- Select the gate output polarities 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.

FIGURE 34-10: SYNCHRONOUS TRANSMISSION



Note: Sync Master mode, SPxBRGL = 0, continuous transmission of two 8-bit words.

FIGURE 34-11: SYNCHRONOUS TRANSMISSION (THROUGH TXEN)



34.4.1.5 Synchronous Master Reception

Data is received at the RX/DT pin. The RX/DT pin output driver is automatically disabled when the EUSART is configured for synchronous master receive operation.

In Synchronous mode, reception is enabled by setting either the Single Receive Enable bit (SREN of the RCxSTA register) or the Continuous Receive Enable bit (CREN of the RCxSTA register).

When SREN is set and CREN is clear, only as many clock cycles are generated as there are data bits in a single character. The SREN bit is automatically cleared at the completion of one character. When CREN is set, clocks are continuously generated until CREN is cleared. If CREN is cleared in the middle of a character the CK clock stops immediately and the partial character is discarded. If SREN and CREN are both set, then SREN is cleared at the completion of the first character and CREN takes precedence.

To initiate reception, set either SREN or CREN. Data is sampled at the RX/DT pin on the trailing edge of the TX/CK clock pin and is shifted into the Receive Shift Register (RSR). When a complete character is received into the RSR, the RXxIF bit is set and the character is automatically transferred to the two character receive FIFO. The Least Significant eight bits of the top character in the receive FIFO are available in RCxREG. The RXxIF bit remains set as long as there are unread characters in the receive FIFO.

Note:	If the RX/DT function is on an analog pin, the corresponding ANSEL bit must be cleared for the receiver to function.

FIGURE 35-9: TYPE-A WAVEFORMS IN 1/2 MUX, 1/2 BIAS DRIVE

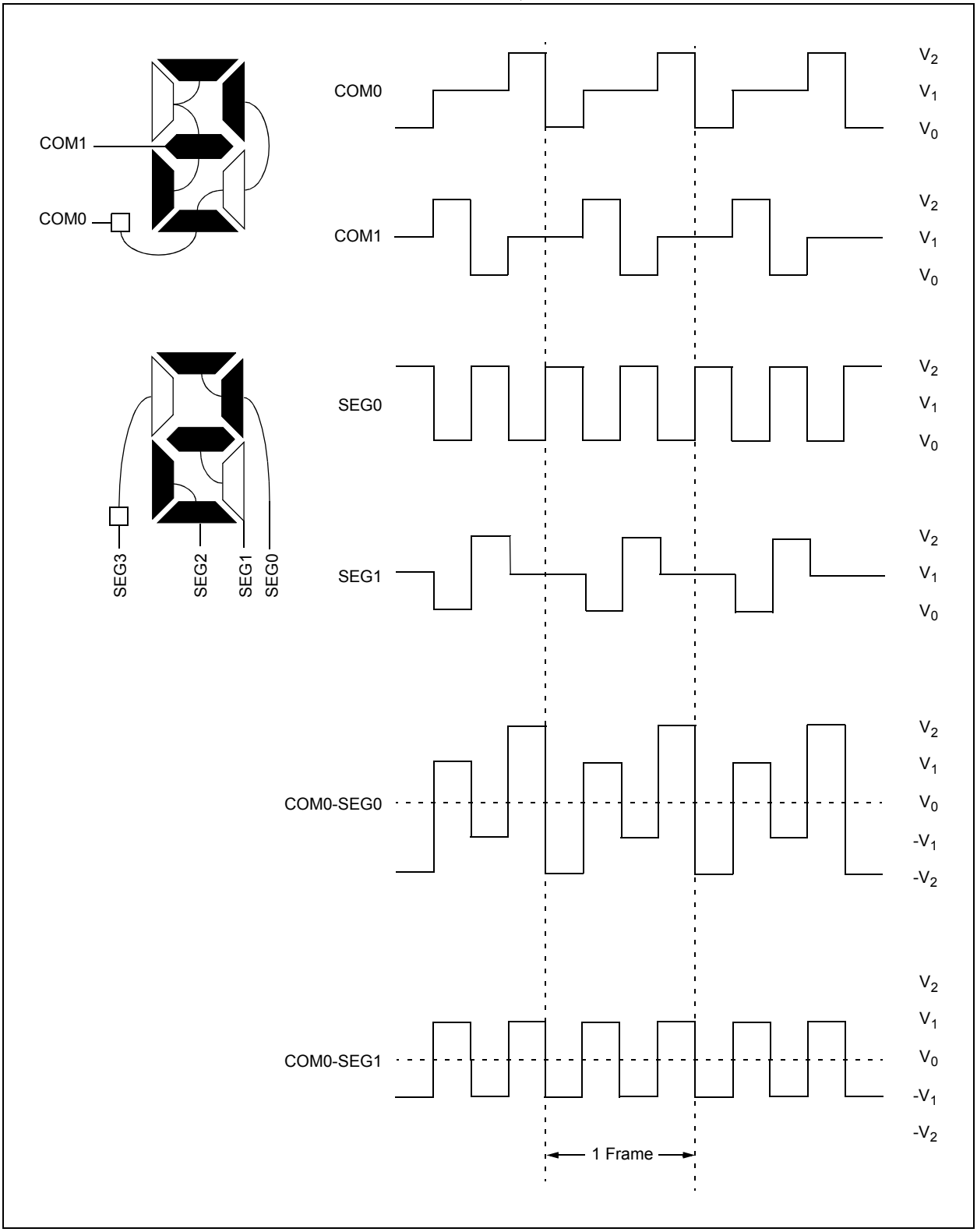
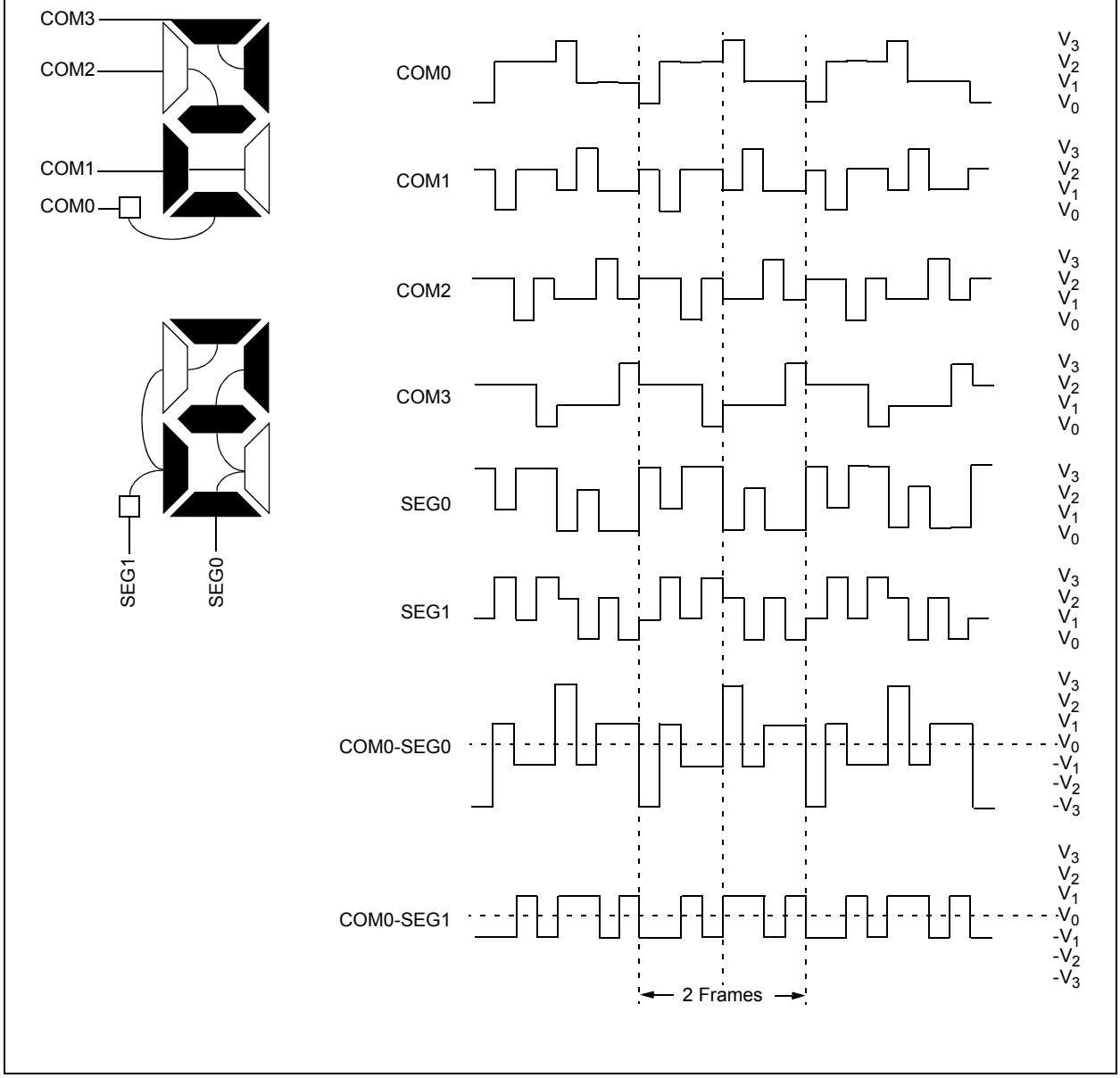


FIGURE 35-18: TYPE-B WAVEFORMS IN 1/4 MUX, 1/3 BIAS DRIVE



REGISTER 35-1: LCDCON: LCD CONTROL REGISTER

R/W-0	R/W-0	HS/C-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
LCDEN	SLPEN	WERR	CS	LMUX3	LMUX2	LMUX1	LMUX0
bit 7							bit 0

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

bit 7 **LCDEN**: LCD Enable bit
1 = LCD module is enabled
0 = LCD module is disabled

bit 6 **SLPEN**: LCD Display Sleep-Enabled bit
1 = Module will stop driving in Sleep
0 = Module will continue driving in Sleep

bit 5 **WERR**: LCD Write Failed Error bit(1)
1 = Write failure to LCDDATA register occurred (must be reset in software)
0 = No LCD write error

bit 4 **CS**: Clock Source Select bit
1 = SOSC Selected
0 = LFINTOSC Selected

bit 3-0 **LMUX<3:0>**: Common Selection bits. Specifies the number of commons(2)

LMUX<3:0>	Multiplex	Bias
0000	All COMs off	—
0001	Static (COM0)	Static
0010	1/2 MUX (COM<1:0>)	1/2
0011	1/3 MUX (COM<2:0>)	1/3
0100	1/4 MUX (COM<3:0>)	1/3
0101	1/5 MUX (COM<4:0>)	1/3
0110	1/6 MUX (COM<5:0>)	1/3
0111	1/7 MUX (COM<6:0>)	1/3
1000	1/8 MUX (COM<7:0>)	1/3

Note 1: Bit can only be set by hardware and only cleared in software by writing to zero.
2: Cannot be changed when LCDEN = 1.

TABLE 38-1: REGISTER FILE SUMMARY FOR PIC16(L)F19155/56/75/76/85/86 DEVICES

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
212h	—	Unimplemented								
213h	—	Unimplemented								
214h	—	Unimplemented								
215h	—	Unimplemented								
216h	—	Unimplemented								
217h	—	Unimplemented								
218h	—	Unimplemented								
219h	—	Unimplemented								
21Ah	—	Unimplemented								
21Bh	—	Unimplemented								
21Ch	—	Unimplemented								
21Dh	—	Unimplemented								
21Eh	CCPTMRS0	P4TSEL<1:0>		P3TSEL1:0>		C2TSEL<1:0>		C1TSEL<1:0>		461
21Fh	—	Unimplemented								
28Ch	T2TMR	TMR2								
28Dh	T2PR	PR2								
28Eh	T2CON	ON	CKPS<2:0>			OUTPS<3:0>				404
28Fh	T2HLT	PSYNC	CKPOL	CKSYNC	MODE<4:0>					405
290h	T2CLKCON	—	—	—	—	CS<3:0>				403
291h	T2RST	—	—	—	—	RSEL<3:0>				406
292h	T4TMR	TMR4								
293h	T4PR	PR4								
294h	T4CON	ON	CKPS<2:0>			OUTPS<3:0>				404
295h	T4HLT	PSYNC	CKPOL	CKSYNC	MODE<4:0>					405
296h	T4CLKCON	—	—	—	—	CS<3:0>				403
297h	T4RST	—	—	—	—	RSEL<3:0>				406
298h	—	Unimplemented								
299h	—	Unimplemented								
29Ah	—	Unimplemented								
29Bh	—	Unimplemented								
29Ch	—	Unimplemented								
29Dh	—	Unimplemented								
29Eh	—	Unimplemented								
29Fh	—	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'.