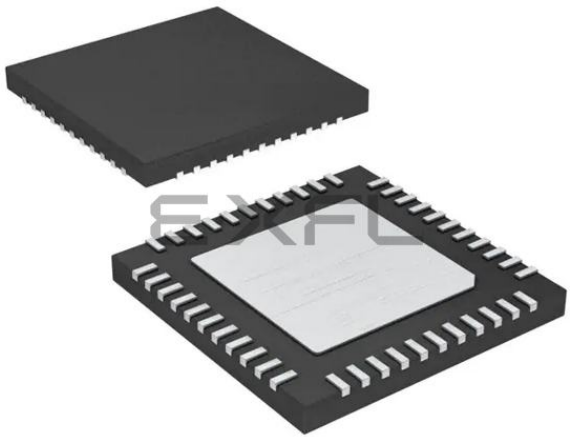


**Welcome to E-XFL.COM**

## 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 | 36 |
| Program Memory Size | 14KB (8K x 14) |
| Program Memory Type | FLASH |
| EEPROM Size | 256 x 8 |
| RAM Size | 512 x 8 |
| Voltage - Supply (Vcc/Vdd) | 1.8V ~ 3.6V |
| Data Converters | A/D 14x10b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 44-VQFN Exposed Pad |
| Supplier Device Package | 44-QFN (8x8) |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/pic16lf1937-i-ml |

**TABLE 2: 40/44-PIN SUMMARY(PIC16(L)F1934/7)**

| I/O | 40-Pin PDIP | 40-Pin UQFN | 44-Pin TQFP | 44-Pin QFN | ANSEL | A/D | Cap Sense | Comparator | SR Latch | Timers | CCP | EUSART | MSSP | LCD | Interrupt | Pull-up | Basic |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RA0 | 2 | 17 | 19 | 19 | Y | AN0 | — | C12IN0-/ C2OUT(1) | SRNQ(1) | — | — | — | $\overline{SS}$(1) | SEG12 | — | — | VCAP |
| RA1 | 3 | 18 | 20 | 20 | Y | AN1 | — | C12IN1- | — | — | — | — | — | SEG7 | — | — | — |
| RA2 | 4 | 19 | 21 | 21 | Y | AN2/ VREF- | — | C2IN+/ DACOUT | — | — | — | — | — | COM2 | — | — | — |
| RA3 | 5 | 20 | 22 | 22 | Y | AN3/ VREF+ | — | C1IN+ | — | — | — | — | — | SEG15 | — | — | — |
| RA4 | 6 | 21 | 23 | 23 | Y | — | CPS6 | C1OUT | SRQ | T0CKI | | — | — | SEG4 | — | — | — |
| RA5 | 7 | 22 | 24 | 24 | Y | AN4 | CPS7 | C2OUT(1) | SRNQ(1) | — | — | — | $\overline{SS}$(1) | SEG5 | — | — | VCAP |
| RA6 | 14 | 29 | 31 | 33 | — | — | — | — | — | — | — | — | — | SEG1 | — | — | OSC2/ CLKOUT VCAP |
| RA7 | 13 | 28 | 30 | 32 | — | — | — | — | — | — | — | — | — | SEG2 | — | — | OSC1/ CLKIN |
| RB0 | 33 | 8 | 8 | 9 | Y | AN12 | CPS0 | — | SRI | — | — | — | — | SEG0 | INT/ IOC | Y | — |
| RB1 | 34 | 9 | 9 | 10 | Y | AN10 | CPS1 | C12IN3- | — | — | — | — | — | VLCD1 | IOC | Y | — |
| RB2 | 35 | 10 | 10 | 11 | Y | AN8 | CPS2 | — | — | — | — | — | — | VLCD2 | IOC | Y | — |
| RB3 | 36 | 11 | 11 | 12 | Y | AN9 | CPS3 | C12IN2- | — | — | CCP2(1)/ P2A(1) | — | — | VLCD3 | IOC | Y | — |
| RB4 | 37 | 12 | 14 | 14 | Y | AN11 | CPS4 | — | — | — | — | — | — | COM0 | IOC | Y | — |
| RB5 | 38 | 13 | 15 | 15 | Y | AN13 | CPS5 | — | — | T1G(1) | CCP3(1)/ P3A(1) | — | — | COM1 | IOC | Y | — |
| RB6 | 39 | 14 | 16 | 16 | — | — | — | — | — | — | — | — | — | SEG14 | IOC | Y | ICSPCLK/ ICDCLK |
| RB7 | 40 | 15 | 17 | 17 | — | — | — | — | — | — | — | — | — | SEG13 | IOC | Y | ICSPDAT/ ICDDAT |
| RC0 | 15 | 30 | 32 | 34 | — | — | — | — | — | T1OSO/ T1CKI | P2B(1) | — | — | — | — | — | — |
| RC1 | 16 | 31 | 35 | 35 | — | — | — | — | — | T1OSI | CCP2(1)/ P2A(1) | — | — | — | — | — | — |
| RC2 | 17 | 32 | 36 | 36 | — | — | — | — | — | — | CCP1/ P1A | — | — | SEG3 | — | — | — |
| RC3 | 18 | 33 | 37 | 37 | — | — | — | — | — | — | — | — | SCK/SCL | SEG6 | — | — | — |
| RC4 | 23 | 38 | 42 | 42 | — | — | — | — | — | T1G(1) | — | — | SDI/SDA | SEG11 | — | — | — |
| RC5 | 24 | 39 | 43 | 43 | — | — | — | — | — | — | — | — | SDO | SEG10 | — | — | — |
| RC6 | 25 | 40 | 44 | 44 | — | — | — | — | — | — | — | TX/CK | — | SEG9 | — | — | — |
| RC7 | 26 | 1 | 1 | 1 | — | — | — | — | — | — | — | RX/DT | — | SEG8 | — | — | — |
| RD0 | 19 | 34 | 38 | 38 | Y | — | CPS8 | — | — | — | — | — | — | COM3 | — | — | — |
| RD1 | 20 | 35 | 39 | 39 | Y | — | CPS9 | — | — | — | CCP4 | — | — | — | — | — | — |
| RD2 | 21 | 36 | 40 | 40 | Y | — | CPS10 | — | — | — | P2B(1) | — | — | — | — | — | — |
| RD3 | 22 | 37 | 41 | 41 | Y | — | CPS11 | — | — | — | P2C | — | — | SEG16 | — | — | — |
| RD4 | 27 | 2 | 2 | 2 | Y | — | CPS12 | — | — | — | P2D | — | — | SEG17 | — | — | — |
| RD5 | 28 | 3 | 3 | 3 | Y | — | CPS13 | — | — | — | P1B | — | — | SEG18 | — | — | — |
| RD6 | 29 | 4 | 4 | 4 | Y | — | CPS14 | — | — | — | P1C | — | — | SEG19 | — | — | — |
| RD7 | 30 | 5 | 5 | 5 | Y | — | CPS15 | — | — | — | P1D | — | — | SEG20 | — | — | — |
| RE0 | 8 | 23 | 25 | 25 | Y | AN5 | — | — | — | — | CCP3(1) P3A(1) | — | — | SEG21 | — | — | — |
| RE1 | 9 | 24 | 26 | 26 | Y | AN6 | — | — | — | — | P3B | — | — | SEG22 | — | — | — |
| RE2 | 10 | 25 | 27 | 27 | Y | AN7 | — | — | — | — | CCP5 | — | — | SEG23 | — | — | — |
| RE3 | 1 | 16 | 18 | 18 | — | — | — | — | — | — | — | — | — | — | — | Y | $\overline{MCLR}$/VPP |
| VDD | 11, 32 | 7, 26 | 7, 28 | 7,8, 28 | — | — | — | — | — | — | — | — | — | — | — | — | VDD |
| Vss | 12, 31 | 6, 27 | 6, 29 | 6,30, 31 | — | — | — | — | — | — | — | — | — | — | — | — | VSS |

**Note 1:** Pin functions can be moved using the APFCON register.

**TABLE 3-9: PIC16(L)F1936 MEMORY MAP, BANK 15**

| Bank 15 | |
|---|---|
| 791h | LCDCON |
| 792h | LCDPS |
| 793h | LCDREF |
| 794h | LCDCST |
| 795h | LCDRL |
| 796h | — |
| 797h | — |
| 798h | LCDSE0 |
| 799h | LCDSE1 |
| 79Ah | — |
| 79Bh | — |
| 79Ch | — |
| 79Dh | — |
| 79Eh | — |
| 79Fh | — |
| 7A0h | LCDDATA0 |
| 7A1h | LCDDATA1 |
| 7A2h | — |
| 7A3h | LCDDATA3 |
| 7A4h | LCDDATA4 |
| 7A5h | — |
| 7A6h | LCDDATA6 |
| 7A7h | LCDDATA7 |
| 7A8h | — |
| 7A9h | LCDDATA9 |
| 7AAh | LCDDATA10 |
| 7ABh | — |
| 7ACh | — |
| 7ADh | — |
| 7AEh | — |
| 7AFh | — |
| 7B0h | — |
| 7B1h | — |
| 7B2h | — |
| 7B3h | — |
| 7B4h | — |
| 7B5h | — |
| 7B6h | — |
| 7B7h | — |
| 7B8h | Unimplemented Read as '0' |
| 7EFh | |

**Legend:** = Unimplemented data memory locations, read as '0'.

**TABLE 3-10: PIC16(L)F1934/7 MEMORY MAP, BANK 15**

| Bank 15 | |
|---|---|
| 791h | LCDCON |
| 792h | LCDPS |
| 793h | LCDREF |
| 794h | LCDCST |
| 795h | LCDRL |
| 796h | — |
| 797h | — |
| 798h | LCDSE0 |
| 799h | LCDSE1 |
| 79Ah | LCDSE2 |
| 79Bh | — |
| 79Ch | — |
| 79Dh | — |
| 79Eh | — |
| 79Fh | — |
| 7A0h | LCDDATA0 |
| 7A1h | LCDDATA1 |
| 7A2h | LCDDATA2 |
| 7A3h | LCDDATA3 |
| 7A4h | LCDDATA4 |
| 7A5h | LCDDATA5 |
| 7A6h | LCDDATA6 |
| 7A7h | LCDDATA7 |
| 7A8h | LCDDATA8 |
| 7A9h | LCDDATA9 |
| 7AAh | LCDDATA10 |
| 7ABh | LCDDATA11 |
| 7ACh | — |
| 7ADh | — |
| 7AEh | — |
| 7AFh | — |
| 7B0h | — |
| 7B1h | — |
| 7B2h | — |
| 7B3h | — |
| 7B4h | — |
| 7B5h | — |
| 7B6h | — |
| 7B7h | — |
| 7B8h | Unimplemented Read as '0' |
| 7EFh | |

**Legend:** = Unimplemented data memory locations, read as '0'.

**TABLE 3-11: PIC16(L)F1934/6/7 MEMORY MAP, BANK 31**

| Bank 31 | |
|---|---|
| F8Ch | Unimplemented Read as '0' |
| | |
| FE3h | |
| FE4h | STATUS_SHAD |
| FE5h | WREG_SHAD |
| FE6h | BSR_SHAD |
| FE7h | PCLATH_SHAD |
| FE8h | FSR0L_SHAD |
| FE9h | FSR0H_SHAD |
| FEAh | FSR1L_SHAD |
| FEBh | FSR1H_SHAD |
| FECh | — |
| FEDh | STKPTR |
| FEEh | TOSL |
| FEFh | TOSH |

**Legend:** = Unimplemented data memory locations, read as '0'.

### 3.2.6 SPECIAL FUNCTION REGISTERS SUMMARY

The Special Function Register Summary for the device family are as follows:

| Device | Bank(s) | Page No. |
|---|---|---|
| PIC16(L)F1934/6/7 | 0 | 39 |
| | 1 | 40 |
| | 2 | 41 |
| | 3 | 42 |
| | 4 | 43 |
| | 5 | 44 |
| | 6 | 45 |
| | 7 | 46 |
| | 8 | 47 |
| | 9-14 | 48 |
| | 15 | 49 |
| | 16-30 | 51 |
| | 31 | 52 |

#### 5.2.2.6 32 MHz Internal Oscillator Frequency Selection

The Internal Oscillator Block can be used with the 4X PLL associated with the External Oscillator Block to produce a 32 MHz internal system clock source. The following settings are required to use the 32 MHz internal clock source:

- The FOSC bits in Configuration Word 1 must be set to use the INTOSC source as the device system clock (FOSC<2:0> = 100).
- The SCS bits in the OSCCON register must be cleared to use the clock determined by FOSC<2:0> in Configuration Word 1 (SCS<1:0> = 00).
- The IRCF bits in the OSCCON register must be set to the 8 MHz HFINTOSC set to use (IRCF<3:0> = 1110).
- The SPLLEN bit in the OSCCON register must be set to enable the 4xPLL, or the PLLEN bit of the Configuration Word 2 must be programmed to a '1'.

| | |
|---|---|
| **Note:** | When using the PLLEN bit of the Configuration Word 2, the 4xPLL cannot be disabled by software and the 8 MHz HFINTOSC option will no longer be available. |

The 4xPLL is not available for use with the internal oscillator when the SCS bits of the OSCCON register are set to '1x'. The SCS bits must be set to '00' to use the 4xPLL with the internal oscillator.

#### 5.2.2.7 Internal Oscillator Clock Switch Timing

When switching between the HFINTOSC, MFINTOSC and the LFINTOSC, the new oscillator may already be shut down to save power (see Figure 5-7). If this is the case, there is a delay after the IRCF<3:0> bits of the OSCCON register are modified before the frequency selection takes place. The OSCSTAT register will reflect the current active status of the HFINTOSC, MFINTOSC and LFINTOSC oscillators. The sequence of a frequency selection is as follows:

1. IRCF<3:0> bits of the OSCCON register are modified.
2. If the new clock is shut down, a clock start-up delay is started.
3. Clock switch circuitry waits for a falling edge of the current clock.
4. The current clock is held low and the clock switch circuitry waits for a rising edge in the new clock.
5. The new clock is now active.
6. The OSCSTAT register is updated as required.
7. Clock switch is complete.

See Figure 5-7 for more details.

If the internal oscillator speed is switched between two clocks of the same source, there is no start-up delay before the new frequency is selected. Clock switching time delays are shown in Table 5-1.

Start-up delay specifications are located in the oscillator tables in the applicable Electrical Specifications Chapter.

## 5.4 Two-Speed Clock Start-up Mode

Two-Speed Start-up mode provides additional power savings by minimizing the latency between external oscillator start-up and code execution. In applications that make heavy use of the Sleep mode, Two-Speed Start-up will remove the external oscillator start-up time from the time spent awake and can reduce the overall power consumption of the device. This mode allows the application to wake-up from Sleep, perform a few instructions using the INTOSC internal oscillator block as the clock source and go back to Sleep without waiting for the external oscillator to become stable.

Two-Speed Start-up provides benefits when the oscillator module is configured for LP, XT or HS modes. The Oscillator Start-up Timer (OST) is enabled for these modes and must count 1024 oscillations before the oscillator can be used as the system clock source.

If the oscillator module is configured for any mode other than LP, XT or HS mode, then Two-Speed Start-up is disabled. This is because the external clock oscillator does not require any stabilization time after POR or an exit from Sleep.

If the OST count reaches 1024 before the device enters Sleep mode, the OSTS bit of the OSCSTAT register is set and program execution switches to the external oscillator. However, the system may never operate from the external oscillator if the time spent awake is very short.

| **Note:** | Executing a SLEEP instruction will abort the oscillator start-up time and will cause the OSTS bit of the OSCSTAT register to remain clear. |
|---|---|

### 5.4.1 TWO-SPEED START-UP MODE CONFIGURATION

Two-Speed Start-up mode is configured by the following settings:

- IESO (of the Configuration Word 1) = 1; Internal/External Switchover bit (Two-Speed Start-up mode enabled).
- SCS (of the OSCCON register) = 00.
- FOSC<2:0> bits in the Configuration Word 1 configured for LP, XT or HS mode.

Two-Speed Start-up mode is entered after:

- Power-on Reset (POR) and, if enabled, after Power-up Timer (PWRT) has expired, or
- Wake-up from Sleep.

**TABLE 5-1: OSCILLATOR SWITCHING DELAYS**

| Switch From | Switch To | Frequency | Oscillator Delay |
|---|---|---|---|
| Sleep/POR | LFINTOSC(1)<br>MFINTOSC(1)<br>HFINTOSC(1) | 31 kHz<br>31.25 kHz-500 kHz<br>31.25 kHz-16 MHz | Oscillator Warm-up Delay (TWARM) |
| Sleep/POR | EC, RC(1) | DC – 32 MHz | 2 cycles |
| LFINTOSC | EC, RC(1) | DC – 32 MHz | 1 cycle of each |
| Sleep/POR | Timer1 Oscillator<br>LP, XT, HS(1) | 32 kHz-20 MHz | 1024 Clock Cycles (OST) |
| Any clock source | MFINTOSC(1)<br>HFINTOSC(1) | 31.25 kHz-500 kHz<br>31.25 kHz-16 MHz | 2 μs (approx.) |
| Any clock source | LFINTOSC(1) | 31 kHz | 1 cycle of each |
| Any clock source | Timer1 Oscillator | 32 kHz | 1024 Clock Cycles (OST) |
| PLL inactive | PLL active | 16-32 MHz | 2 ms (approx.) |

**Note 1:** PLL inactive.

**EXAMPLE 11-4: ERASING ONE ROW OF PROGRAM MEMORY -**

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


        BCF         INTCON,GIE      ; Disable ints so required sequences will execute properly
        BANKSEL     EEADRL
        MOVF        ADDRL,W         ; Load lower 8 bits of erase address boundary
        MOVWF       EEADRL
        MOVF        ADDRH,W         ; Load upper 6 bits of erase address boundary
        MOVWF       EEADRH
        BSF         EECON1,EEPGD    ; Point to program memory
        BCF         EECON1,CFGS     ; Not configuration space
        BSF         EECON1,FREE     ; Specify an erase operation
        BSF         EECON1,WREN     ; Enable writes

Required
Sequence
        MOVLW       55h             ; Start of required sequence to initiate erase
        MOVWF       EECON2          ; Write 55h
        MOVLW       0AAh            ;
        MOVWF       EECON2          ; Write AAh
        BSF         EECON1,WR       ; Set WR bit to begin erase
        NOP                         ; Any instructions here are ignored as processor
                                    ; halts to begin erase sequence
        NOP                         ; Processor will stop here and wait for erase complete.

                                    ; after erase processor continues with 3rd instruction

        BCF         EECON1,WREN     ; Disable writes
        BSF         INTCON,GIE      ; Enable interrupts
```

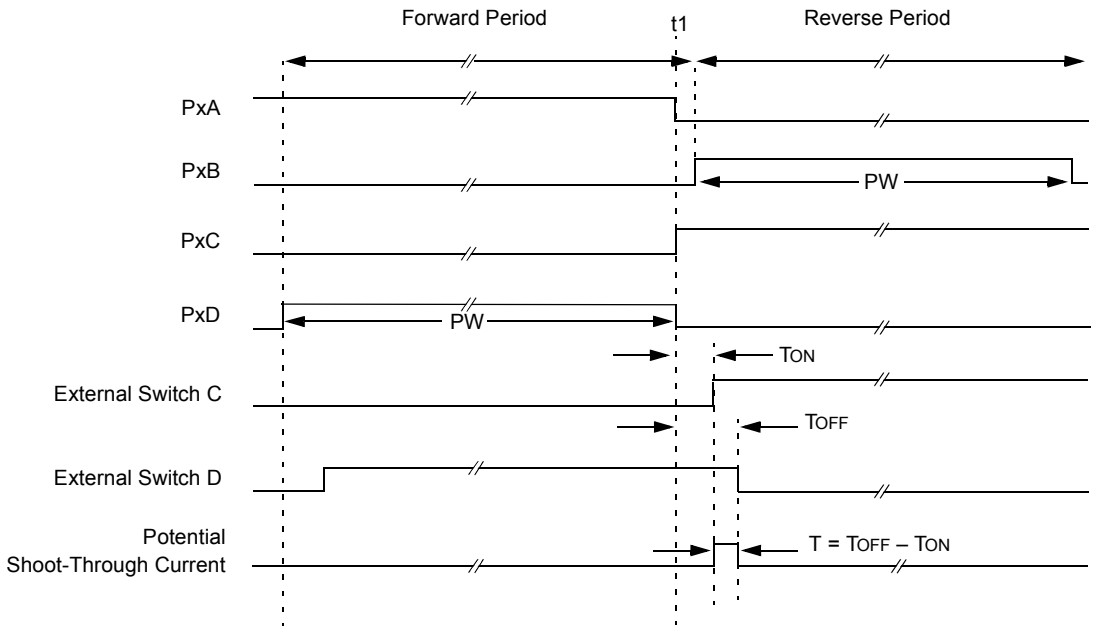
**TABLE 12-6: SUMMARY OF REGISTERS ASSOCIATED WITH PORTB**

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Register on Page |
|---|---|---|---|---|---|---|---|---|---|
| ADCON0 | — | CHS<4:0> | | | | | GO/$\overline{\text{DONE}}$ | ADON | 163 |
| ANSELB | — | — | ANSB5 | ANSB4 | ANSB3 | ANSB2 | ANSB1 | ANSB0 | 139 |
| APFCON | — | CCP3SEL | T1GSEL | P2BSEL | SRNQSEL | C2OUTSEL | SSSEL | CCP2SEL | 131 |
| CCPxCON | PxM<1:0> | | DCxB<1:0> | | CCPxM<3:0> | | | | 234 |
| CPSCON0 | CPSON | — | — | — | CPSRNG<1:0> | | CPSOUT | T0XCS | 323 |
| CPSCON1 | — | — | — | — | CPSCH<3:> | | | | 324 |
| INTCON | GIE | PEIE | TMR0IE | INTE | IOCIE | TMR0IF | INTF | IOCIF | 98 |
| IOCBP | IOCBP7 | IOCBP6 | IOCBP5 | IOCBP4 | IOCBP3 | IOCBP2 | IOCBP1 | IOCBP0 | 152 |
| IOCBN | IOCBN7 | IOCBN6 | IOCBN5 | IOCBN4 | IOCBN3 | IOCBN2 | IOCBN1 | IOCBN0 | 152 |
| IOCBF | IOCBF7 | IOCBF6 | IOCBF5 | IOCBF4 | IOCBF3 | IOCBF2 | IOCBF1 | IOCBF0 | 152 |
| LATB | LATB7 | LATB6 | LATB5 | LATB4 | LATB3 | LATB2 | LATB1 | LATB0 | 138 |
| LCDCON | LCDEN | SLPEN | WERR | — | CS<1:0> | | LMUX<1:0> | | 329 |
| LCDSE0 | SE7 | SE6 | SE5 | SE4 | SE3 | SE2 | SE1 | SE0 | 333 |
| LCDSE1 | SE15 | SE14 | SE13 | SE12 | SE11 | SE10 | SE9 | SE8 | 333 |
| OPTION_REG | $\overline{\text{WPUEN}}$ | INTEDG | TMR0CS | TMR0SE | PSA | PS<2:0> | | | 193 |
| PORTB | RB7 | RB6 | RB5 | RB4 | RB3 | RB2 | RB1 | RB0 | 138 |
| T1GCON | TMR1GE | T1GPOL | T1GTM | T1GSPM | T1GGO/$\overline{\text{DONE}}$ | T1GVAL | T1GSS<1:0> | | 204 |
| TRISB | TRISB7 | TRISB6 | TRISB5 | TRISB4 | TRISB3 | TRISB2 | TRISB1 | TRISB0 | 138 |
| WPUB | WPUB7 | WPUB6 | WPUB5 | WPUB4 | WPUB3 | WPUB2 | WPUB1 | WPUB0 | 139 |

**Legend:** x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by PORTB.

**NOTES:**

**FIGURE 23-13: EXAMPLE OF PWM DIRECTION CHANGE AT NEAR 100% DUTY CYCLE**




**Note 1:** All signals are shown as active-high.

**2:** TON is the turn on delay of power switch QC and its driver.

**3:** TOFF is the turn off delay of power switch QD and its driver.

### 23.4.4 AUTO-RESTART MODE

The Enhanced PWM can be configured to automatically restart the PWM signal once the auto-shutdown condition has been removed. Auto-restart is enabled by setting the PxRSEN bit in the PWMxCON register.

If auto-restart is enabled, the CCPxASE bit will remain set as long as the auto-shutdown condition is active. When the auto-shutdown condition is removed, the CCPxASE bit will be cleared via hardware and normal operation will resume.

**FIGURE 23-15: PWM AUTO-SHUTDOWN WITH AUTO-RESTART (PXRSEN = 1)**

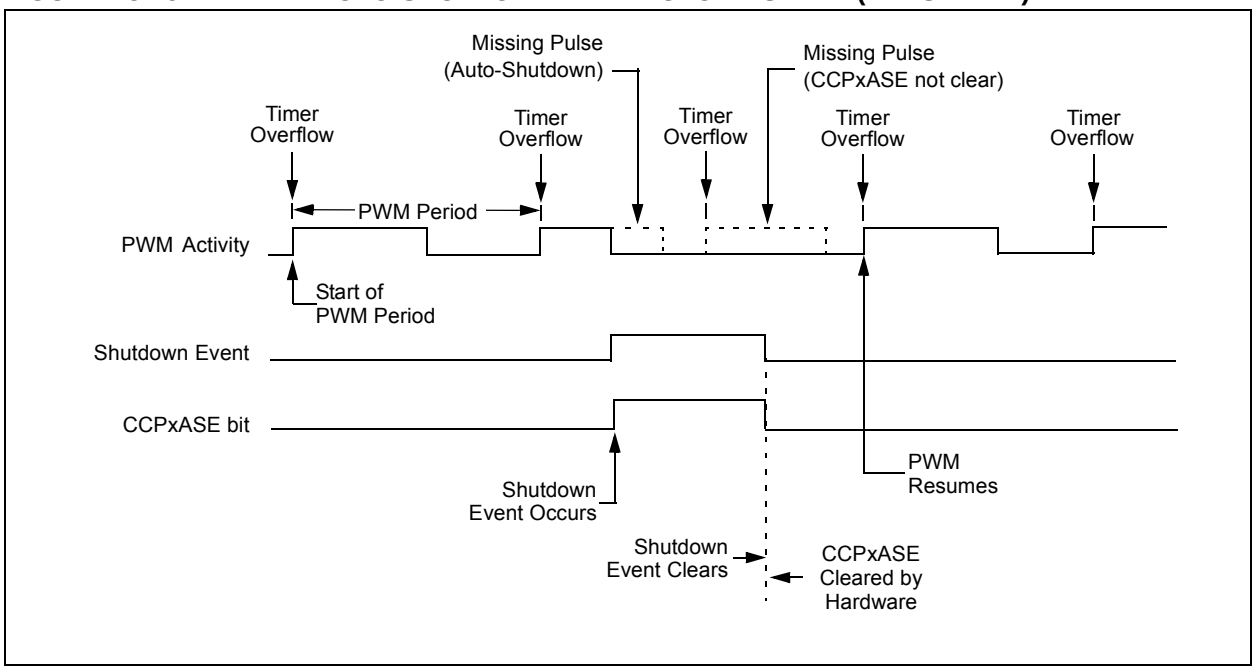
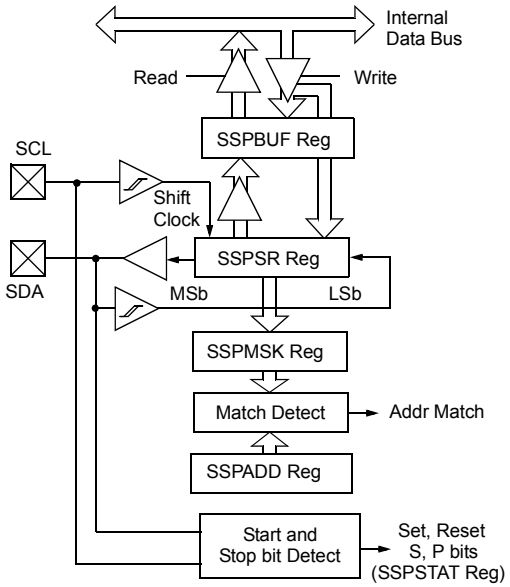


**FIGURE 24-3: MSSP BLOCK DIAGRAM ($I^2C$™ SLAVE MODE)**



## 24.2 SPI Mode Overview

The Serial Peripheral Interface (SPI) bus is a synchronous serial data communication bus that operates in Full-Duplex mode. Devices communicate in a master/slave environment where the master device initiates the communication. A slave device is controlled through a Chip Select known as Slave Select.

The SPI bus specifies four signal connections:

- Serial Clock (SCK)
- Serial Data Out (SDO)
- Serial Data In (SDI)
- Slave Select ($\overline{SS}$)

Figure 24-1 shows the block diagram of the MSSP module when operating in SPI Mode.

The SPI bus operates with a single master device and one or more slave devices. When multiple slave devices are used, an independent Slave Select connection is required from the master device to each slave device.

Figure 24-4 shows a typical connection between a master device and multiple slave devices.

The master selects only one slave at a time. Most slave devices have tri-state outputs so their output signal appears disconnected from the bus when they are not selected.

Transmissions involve two shift registers, eight bits in size, one in the master and one in the slave. With either the master or the slave device, data is always shifted out one bit at a time, with the Most Significant bit (MSb) shifted out first. At the same time, a new Least Significant bit (LSb) is shifted into the same register.

Figure 24-5 shows a typical connection between two processors configured as master and slave devices.

Data is shifted out of both shift registers on the programmed clock edge and latched on the opposite edge of the clock.

The master device transmits information out on its SDO output pin which is connected to, and received by, the slave's SDI input pin. The slave device transmits information out on its SDO output pin, which is connected to, and received by, the master's SDI input pin.

To begin communication, the master device first sends out the clock signal. Both the master and the slave devices should be configured for the same clock polarity.

The master device starts a transmission by sending out the MSb from its shift register. The slave device reads this bit from that same line and saves it into the LSb position of its shift register.

During each SPI clock cycle, a full-duplex data transmission occurs. This means that while the master device is sending out the MSb from its shift register (on its SDO pin) and the slave device is reading this bit and saving it as the LSb of its shift register, that the slave device is also sending out the MSb from its shift register (on its SDO pin) and the master device is reading this bit and saving it as the LSb of its shift register.

After 8 bits have been shifted out, the master and slave have exchanged register values.

If there is more data to exchange, the shift registers are loaded with new data and the process repeats itself.

Whether the data is meaningful or not (dummy data), depends on the application software. This leads to three scenarios for data transmission:

- Master sends useful data and slave sends dummy data.
- Master sends useful data and slave sends useful data.
- Master sends dummy data and slave sends useful data.

Transmissions may involve any number of clock cycles. When there is no more data to be transmitted, the master stops sending the clock signal and it deselects the slave.

Every slave device connected to the bus that has not been selected through its slave select line must disregard the clock and transmission signals and must not transmit out any data of its own.

25.1.2.8 Asynchronous Reception Set-up:

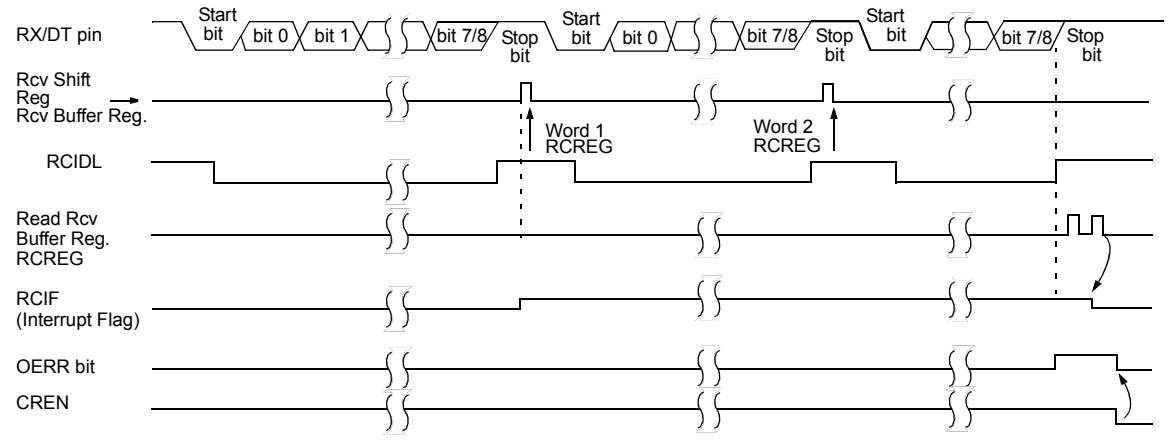
1. Initialize the SPBRGH, SPBRGL register pair and the BRGH and BRG16 bits to achieve the desired baud rate (see **Section 25.3 "EUSART Baud Rate Generator (BRG)"**).
2. Clear the ANSEL bit for the RX pin (if applicable).
3. Enable the serial port by setting the SPEN bit. The SYNC bit must be clear for asynchronous operation.
4. If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
5. If 9-bit reception is desired, set the RX9 bit.
6. Enable reception by setting the CREN bit.
7. The RCIF interrupt flag bit will be set when a character is transferred from the RSR to the receive buffer. An interrupt will be generated if the RCIE interrupt enable bit was also set.
8. Read the RCSTA register to get the error flags and, if 9-bit data reception is enabled, the ninth data bit.
9. Get the received 8 Least Significant data bits from the receive buffer by reading the RCREG register.
10. If an overrun occurred, clear the OERR flag by clearing the CREN receiver enable bit.

25.1.2.9 9-bit Address Detection Mode Set-up

This mode would typically be used in RS-485 systems. To set up an Asynchronous Reception with Address Detect Enable:

1. Initialize the SPBRGH, SPBRGL register pair and the BRGH and BRG16 bits to achieve the desired baud rate (see **Section 25.3 "EUSART Baud Rate Generator (BRG)"**).
2. Clear the ANSEL bit for the RX pin (if applicable).
3. Enable the serial port by setting the SPEN bit. The SYNC bit must be clear for asynchronous operation.
4. If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
5. Enable 9-bit reception by setting the RX9 bit.
6. Enable address detection by setting the ADDEN bit.
7. Enable reception by setting the CREN bit.
8. The RCIF interrupt flag bit will be set when a character with the ninth bit set is transferred from the RSR to the receive buffer. An interrupt will be generated if the RCIE interrupt enable bit was also set.
9. Read the RCSTA register to get the error flags. The ninth data bit will always be set.
10. Get the received 8 Least Significant data bits from the receive buffer by reading the RCREG register. Software determines if this is the device's address.
11. If an overrun occurred, clear the OERR flag by clearing the CREN receiver enable bit.
12. If the device has been addressed, clear the ADDEN bit to allow all received data into the receive buffer and generate interrupts.

**FIGURE 25-5: ASYNCHRONOUS RECEPTION**




**Note:** This timing diagram shows three words appearing on the RX input. The RCREG (receive buffer) is read after the third word, causing the OERR (overrun) bit to be set.

## 26.5 Timer Resources

To measure the change in frequency of the capacitive sensing oscillator, a fixed time base is required. For the period of the fixed time base, the capacitive sensing oscillator is used to clock either Timer0 or Timer1. The frequency of the capacitive sensing oscillator is equal to the number of counts in the timer divided by the period of the fixed time base.

## 26.6 Fixed Time Base

To measure the frequency of the capacitive sensing oscillator, a fixed time base is required. Any timer resource or software loop can be used to establish the fixed time base. It is up to the end user to determine the method in which the fixed time base is generated.

| | |
|---|---|
| **Note:** | The fixed time base can not be generated by the timer resource that the capacitive sensing oscillator is clocking. |

### 26.6.1 TIMER0

To select Timer0 as the timer resource for the CPS module:

- Set the T0XCS bit of the CPSCON0 register.
- Clear the TMR0CS bit of the OPTION_REG register.

When Timer0 is chosen as the timer resource, the capacitive sensing oscillator will be the clock source for Timer0. Refer to **Section 20.0 “Timer0 Module”** for additional information.

### 26.6.2 TIMER1

To select Timer1 as the timer resource for the CPS module, set the TMR1CS<1:0> of the T1CON register to ‘11’. When Timer1 is chosen as the timer resource, the capacitive sensing oscillator will be the clock source for Timer1. Because the Timer1 module has a gate control, developing a time base for the frequency measurement can be simplified by using the Timer0 overflow flag.

It is recommend that the Timer0 overflow flag, in conjunction with the Toggle mode of the Timer1 gate, be used to develop the fixed time base required by the software portion of the CPS module. Refer to **Section 21.12 “Timer1 Gate Control Register”** for additional information.

**TABLE 26-2: TIMER1 ENABLE FUNCTION**

| TMR1ON | TMR1GE | Timer1 Operation |
|---|---|---|
| 0 | 0 | Off |
| 0 | 1 | Off |
| 1 | 0 | On |
| 1 | 1 | Count Enabled by input |

## 26.7 Software Control

The software portion of the CPS module is required to determine the change in frequency of the capacitive sensing oscillator. This is accomplished by the following:

- Setting a fixed time base to acquire counts on Timer0 or Timer1.
- Establishing the nominal frequency for the capacitive sensing oscillator.
- Establishing the reduced frequency for the capacitive sensing oscillator due to an additional capacitive load.
- Set the frequency threshold.

### 26.7.1 NOMINAL FREQUENCY (NO CAPACITIVE LOAD)

To determine the nominal frequency of the capacitive sensing oscillator:

- Remove any extra capacitive load on the selected CPSx pin.
- At the start of the fixed time base, clear the timer resource.
- At the end of the fixed time base save the value in the timer resource.

The value of the timer resource is the number of oscillations of the capacitive sensing oscillator for the given time base. The frequency of the capacitive sensing oscillator is equal to the number of counts on in the timer divided by the period of the fixed time base.

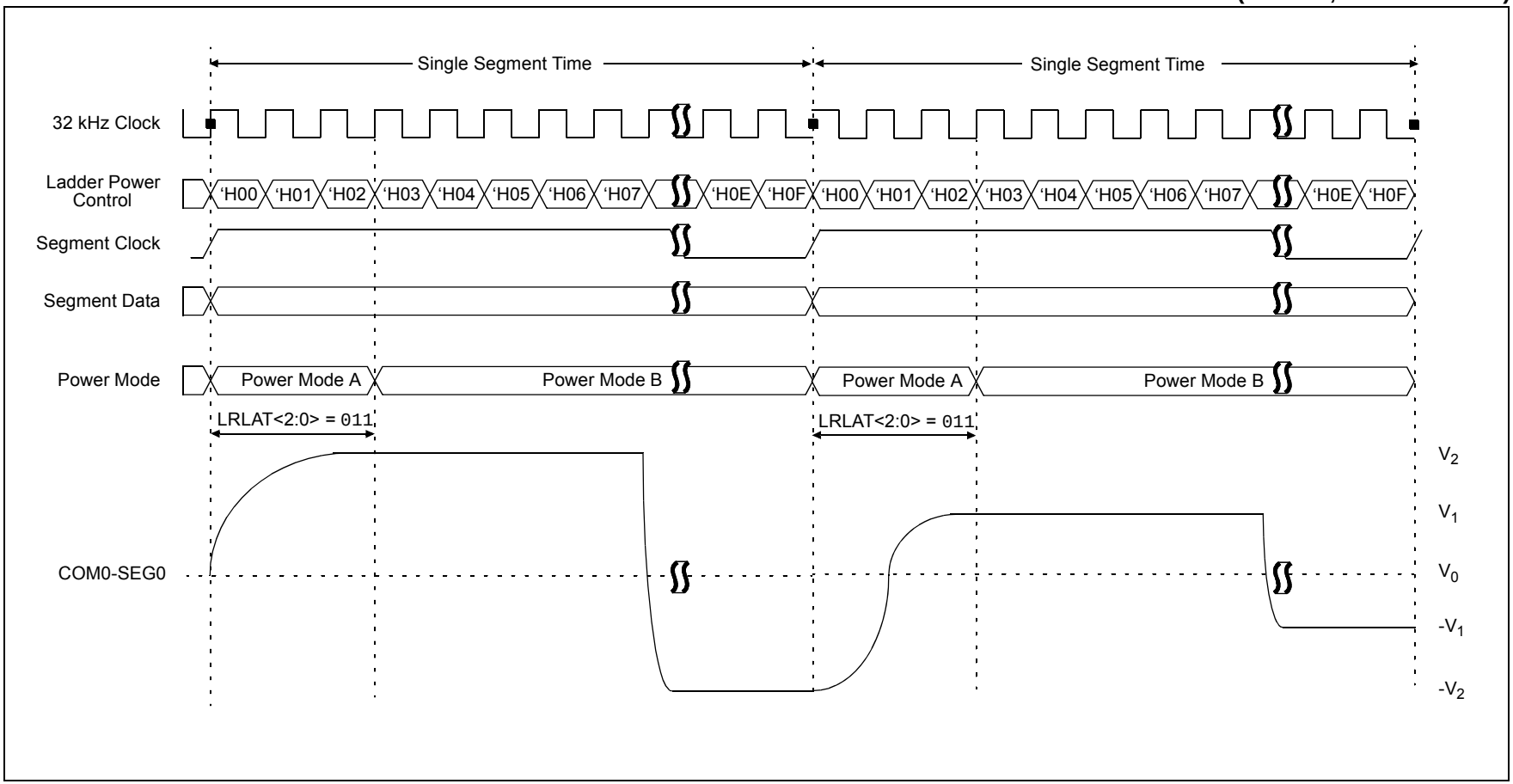
### 26.7.2 REDUCED FREQUENCY (ADDITIONAL CAPACITIVE LOAD)

The extra capacitive load will cause the frequency of the capacitive sensing oscillator to decrease. To determine the reduced frequency of the capacitive sensing oscillator:

- Add a typical capacitive load on the selected CPSx pin.
- Use the same fixed time base as the nominal frequency measurement.
- At the start of the fixed time base, clear the timer resource.
- At the end of the fixed time base save the value in the timer resource.

The value of the timer resource is the number of oscillations of the capacitive sensing oscillator with an additional capacitive load. The frequency of the capacitive sensing oscillator is equal to the number of counts on in the timer divided by the period of the fixed time base. This frequency should be less than the value obtained during the nominal frequency measurement.

**FIGURE 27-5: LCD INTERNAL REFERENCE LADDER POWER MODE SWITCHING DIAGRAM – TYPE A WAVEFORM (1/2 MUX, 1/2 BIAS DRIVE)**



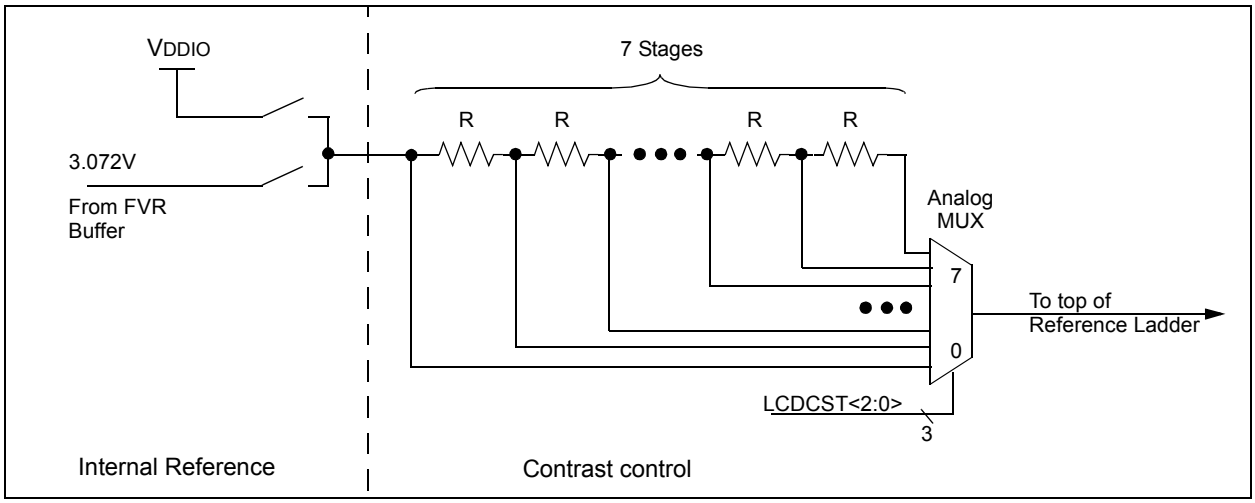
### 27.4.4 CONTRAST CONTROL

The LCD contrast control circuit consists of a seven-tap resistor ladder, controlled by the LCDCST bits. Refer to Figure 27-7.

The contrast control circuit is used to decrease the output voltage of the signal source by a total of approximately 10%, when LCDCST = 111.

Whenever the LCD module is inactive (LCDA = 0), the contrast control ladder will be turned off (open).

**FIGURE 27-7: INTERNAL REFERENCE AND CONTRAST CONTROL BLOCK DIAGRAM**




### 27.4.5 INTERNAL REFERENCE

Under firmware control, an internal reference for the LCD bias voltages can be enabled. When enabled, the source of this voltage can be either VDDIO or a voltage 3 times the main fixed voltage reference (3.072V). 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.

When the internal reference is enabled and the Fixed Voltage Reference is selected, the LCDIRI bit can be used to minimize power consumption by tieing into the LCD Reference Ladder Automatic Power mode switching. When LCDIRI = 1 and the LCD reference ladder is in Power mode 'B', the LCD internal FVR buffer is disabled.

.

| **Note:** | The LCD module automatically turns on the Fixed Voltage Reference when needed. |
|---|---|

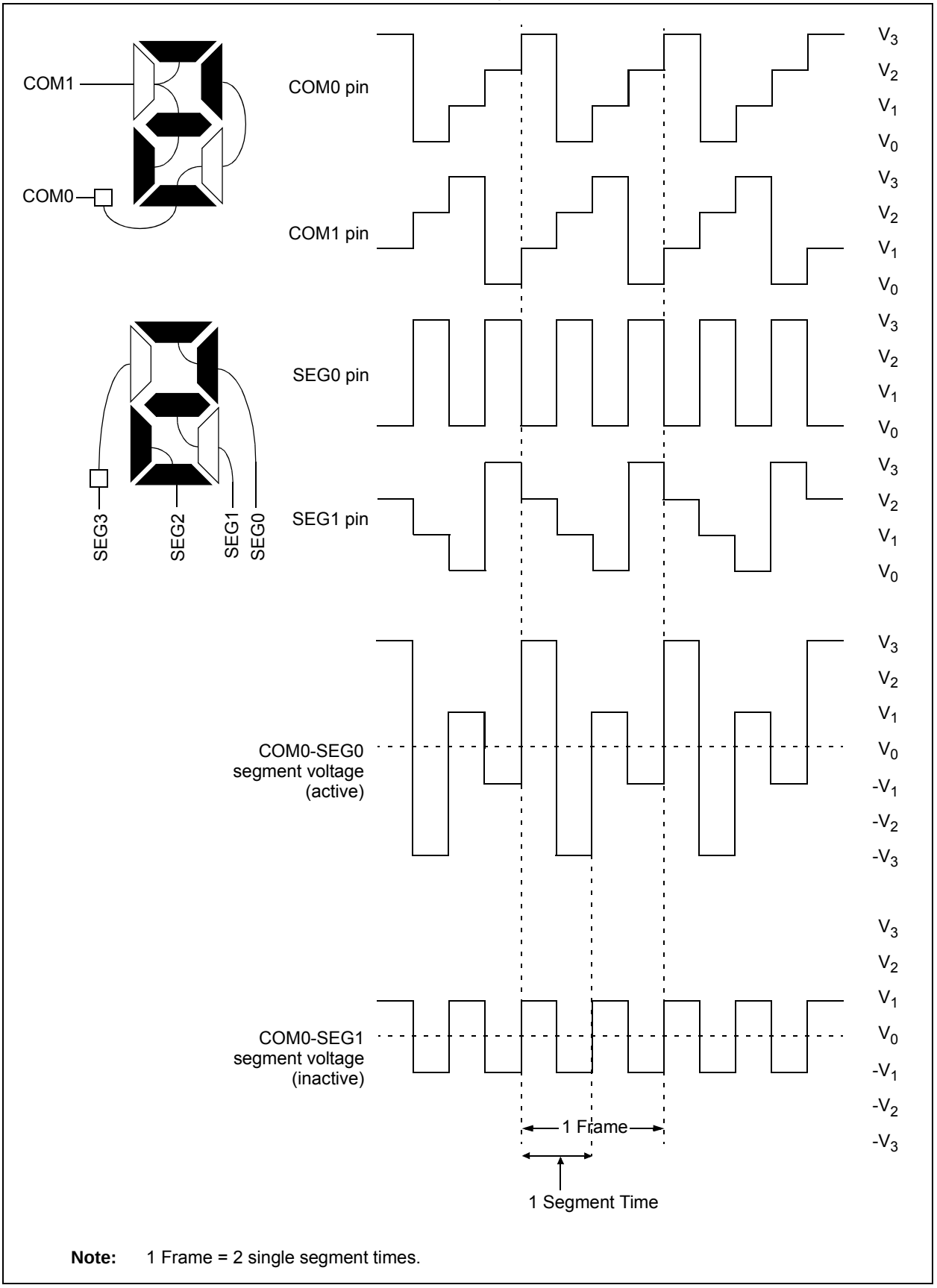
### 27.4.6 VLCD<3:1> PINS

The VLCD<3:1> pins provide the ability for an external LCD bias network to be used instead of the internal ladder. Use of the VLCD<3:1> pins does not prevent use of the internal ladder. Each VLCD pin has an independent control in the LCDREF register (Register 27-3), allowing access to any or all of the LCD Bias signals. This architecture allows for maximum flexibility in different applications

For example, the VLCD<3:1> pins may be used to add capacitors to the internal reference ladder, increasing the drive capacity.

For applications where the internal contrast control is insufficient, the firmware can choose to only enable the VLCD3 pin, allowing an external contrast control circuit to use the internal reference divider.

**FIGURE 27-11: TYPE-A WAVEFORMS IN 1/2 MUX, 1/3 BIAS DRIVE**




**Note:** 1 Frame = 2 single segment times.

## 28.0 IN-CIRCUIT SERIAL PROGRAMMING™ (ICSP™)

ICSP™ programming allows customers to manufacture circuit boards with unprogrammed devices. Programming can be done after the assembly process allowing the device to be programmed with the most recent firmware or a custom firmware. Five pins are needed for ICSP™ programming:

- ICSPCLK
- ICSPDAT
- $\overline{MCLR}/V_{PP}$
- $V_{DD}$
- $V_{SS}$

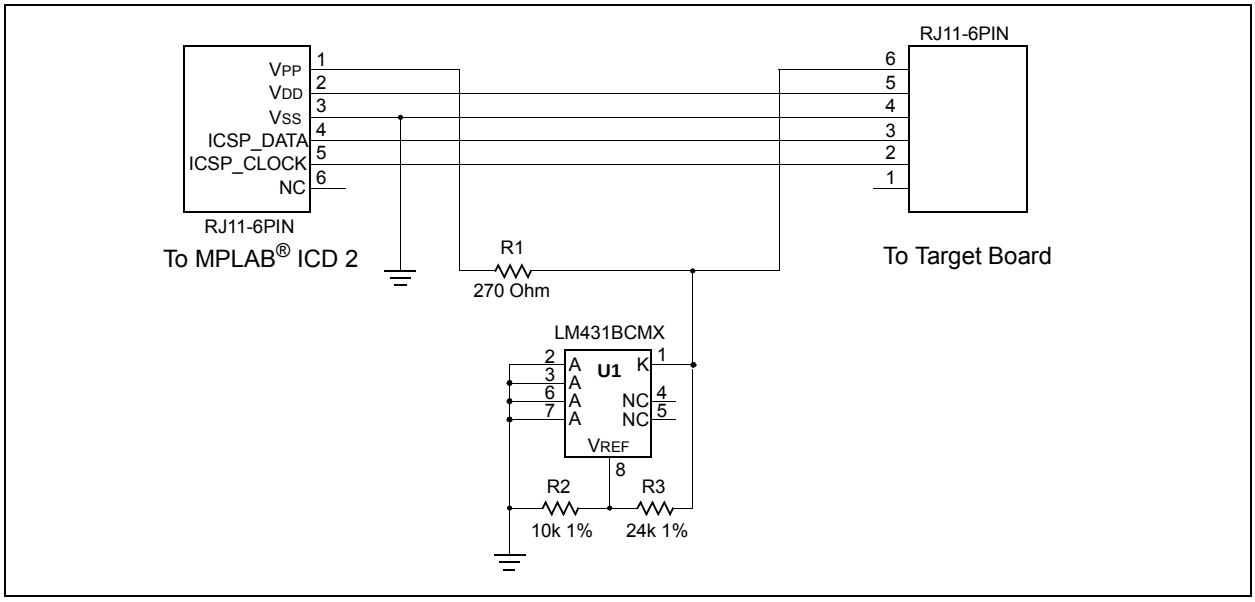
In Program/Verify mode the Program Memory, User IDs and the Configuration Words are programmed through serial communications. The ICSPDAT pin is a bidirectional I/O used for transferring the serial data and the ICSPCLK pin is the clock input. For more information on ICSP™ refer to the "*PIC16193X/PIC16LF193X Memory Programming Specification*" (DS41360).

### 28.1 High-Voltage Programming Entry Mode

The device is placed into High-Voltage Programming Entry mode by holding the ICSPCLK and ICSPDAT pins low then raising the voltage on $\overline{MCLR}/V_{PP}$ to $V_{IHH}$.

Some programmers produce $V_{PP}$ greater than $V_{IHH}$ (9.0V), an external circuit is required to limit the $V_{PP}$ voltage. See Figure 28-1 for example circuit.

**FIGURE 28-1: VPP LIMITER EXAMPLE CIRCUIT**




| **Note:** | The MPLAB® ICD 2 produces a $V_{PP}$ voltage greater than the maximum $V_{PP}$ specification of the PIC16(L)F1934/6/7. |
|---|---|

## 30.4 DC Characteristics: PIC16(L)F1934/6/7-I/E

| DC CHARACTERISTICS | | | Standard Operating Conditions (unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for industrial<br>-40°C ≤ TA ≤ +125°C for extended | | | | |
|---|---|---|---|---|---|---|---|
| **Param No.** | **Sym.** | **Characteristic** | **Min.** | **Typ†** | **Max.** | **Units** | **Conditions** |
| | VIL | **Input Low Voltage** | | | | | |
| | | I/O PORT: | | | | | |
| D032 | | with TTL buffer | — | — | 0.8 | V | 4.5V ≤ VDD ≤ 5.5V |
| D032A | | | — | — | 0.15 VDD | V | 1.8V ≤ VDD ≤ 4.5V |
| D033 | | with Schmitt Trigger buffer | — | — | 0.2 VDD | V | 2.0V ≤ VDD ≤ 5.5V |
| | | with I²C™ levels | — | — | 0.3 VDD | V | |
| | | with SMBus levels | — | — | 0.8 | V | 2.7V ≤ VDD ≤ 5.5V |
| D034 | | $\overline{MCLR}$, OSC1 (RC mode)(1) | — | — | 0.2 VDD | V | |
| D034A | | OSC1 (HS mode) | — | — | 0.3 VDD | V | |
| | VIH | **Input High Voltage** | | | | | |
| | | I/O ports: | | | | | |
| D040 | | with TTL buffer | 2.0 | — | — | V | 4.5V ≤ VDD ≤ 5.5V |
| D040A | | | 0.25 VDD + 0.8 | — | — | V | 1.8V ≤ VDD ≤ 4.5V |
| D041 | | with Schmitt Trigger buffer | 0.8 VDD | — | — | V | 2.0V ≤ VDD ≤ 5.5V |
| | | with I²C™ levels | 0.7 VDD | — | — | V | |
| | | with SMBus levels | 2.1 | — | — | V | 2.7V ≤ VDD ≤ 5.5V |
| D042 | | $\overline{MCLR}$ | 0.8 VDD | — | — | V | |
| D043A | | OSC1 (HS mode) | 0.7 VDD | — | — | V | |
| D043B | | OSC1 (RC mode) | 0.9 VDD | — | — | V | VDD > 2.0V **(Note 1)** |
| | IIL | **Input Leakage Current(2)** | | | | | |
| D060 | | I/O ports | — | ± 5 | ± 125 | nA | VSS ≤ VPIN ≤ VDD, Pin at high-impedance @ 85°C |
| | | | | ± 5 | ± 1000 | nA | 125°C |
| D061 | | $\overline{MCLR}$(3) | — | ± 50 | ± 200 | nA | VSS ≤ VPIN ≤ VDD @ 85°C |
| | IPUR | **Weak Pull-up Current** | | | | | |
| D070* | | | 25 | 100 | 200 | μA | VDD = 3.3V, VPIN = VSS |
| | | | 25 | 140 | 300 | | VDD = 5.0V, VPIN = VSS |
| | VOL | **Output Low Voltage(4)** | | | | | |
| D080 | | I/O ports | — | — | 0.6 | V | IOL = 8mA, VDD = 5V<br>IOL = 6mA, VDD = 3.3V<br>IOL = 1.8mA, VDD = 1.8V |
| | VOH | **Output High Voltage(4)** | | | | | |
| D090 | | I/O ports | VDD - 0.7 | — | — | V | IOH = 3.5mA, VDD = 5V<br>IOH = 3mA, VDD = 3.3V<br>IOH = 1mA, VDD = 1.8V |
| | | **Capacitive Loading Specs on Output Pins** | | | | | |
| D101* | COSC2 | OSC2 pin | — | — | 15 | pF | In XT, HS and LP modes when external clock is used to drive OSC1 |
| D101A* | CIO | All I/O pins | — | — | 50 | pF | |

* These parameters are characterized but not tested.

† Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

**Note 1:** In RC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended to use an external clock in RC mode.

**2:** Negative current is defined as current sourced by the pin.

**3:** The leakage current on the $\overline{MCLR}$ pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.

**4:** Including OSC2 in CLKOUT mode.

**FIGURE 31-69: PIC16F1937 WDT**

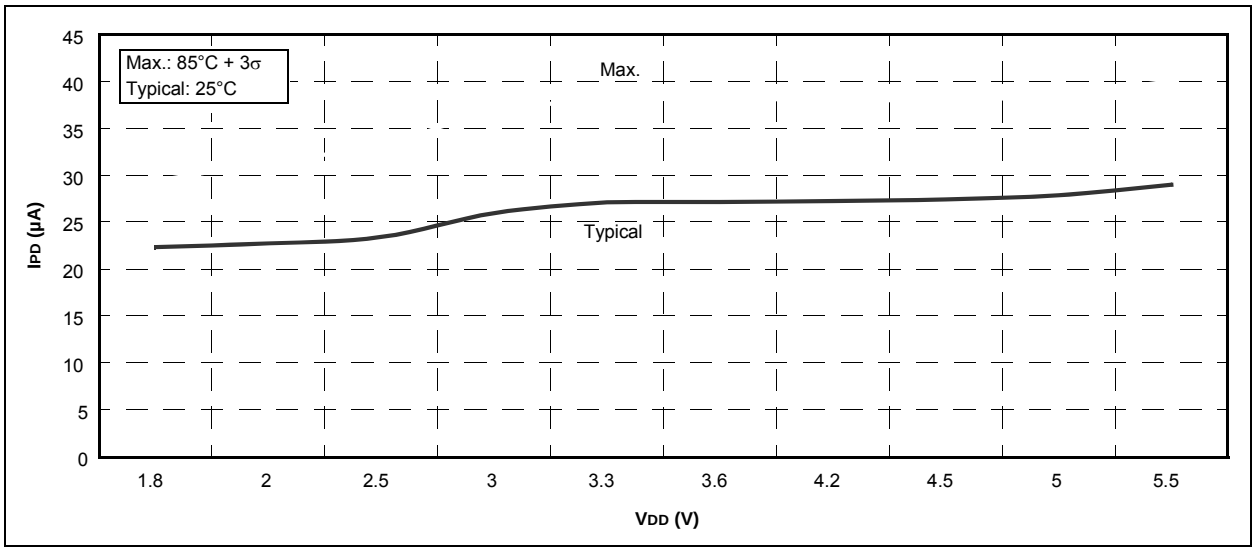



**FIGURE 31-70: PIC16LF1937 FVR_DAC**

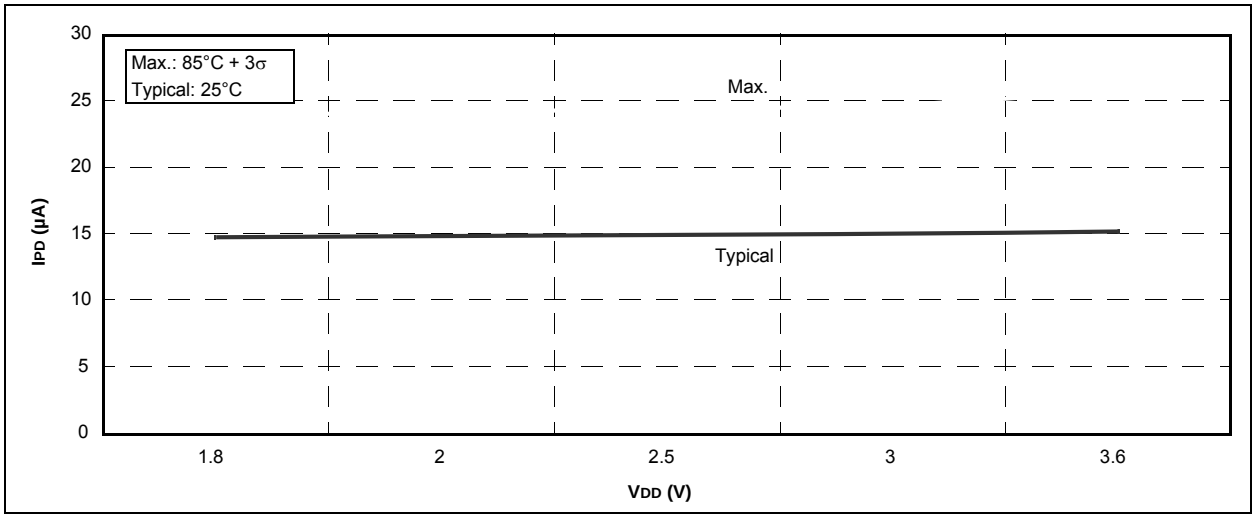



**FIGURE 31-71: PIC16F1937 FVR_DAC**